NTEC Department
(Nottingham Transportation Engineering Centre)



3D-SIMULATION OF PACKING OF AGGREGATES WITH REALISTIC MORPHOLOGY AND SIZE DISTRIBUTION, BY USING A REAL-TIME PHYSICS´ ENGINE

A Thesis

Presented to

University of Nottingham

In Partial Fulfillment

of the Requirements for the Degree of

Doctor of Philosophy

---

by

Sonia Elisa Michot Roberto

3D-SIMULATION OF PACKING OF AGGREGATES WITH REALISTIC MORPHOLOGY AND SIZE DISTRIBUTION, BY USING A REAL-TIME PHYSICS´ ENGINE

Supervised by:

Dr. Álvaro García Hernández, Associate Professor of Faculty of Engineering
University of Nottingham

Dr. Andrew Dawson, Ex-Associate Professor of Faculty of Engineering
University of Nottingham

Dr. Nick Thom, Assistant Professor of Faculty of Engineering
University of Nottingham

DEDICATION

Dedicada a mi maravillosa, guapa e increible hermana, “Patatita” y a mis padres, que los quiero con locura. A mi compañero del alma, por aguantarme y apoyarme siempre; y a mis buenos amigos, que siempre han estado pendientes de mi y de la evolución de la tesis.

Dedicated to my wonderful, beautiful, and incredible sister, “little potato” and my parents, who I love them SOOO MUCH. To my soul mate, for always being patient with me and supporting me; and to my good friends, who have always been asking for how I am doing and for the evolution of the thesis.

“I never let schooling interfere with my education”. Mark Twain

ACKNOWLEDGEMENTS

Thanks to Jorge and Santiago for introducing me to the world of the programming in an entertaining way and for opening my mind to overcome new challenges. Thanks to my supervisors to correct my thesis.

LIST OF PUBLICATIONS

- García-Hernández, A., Michot-Roberto, S., Dopazo-Hilario, S., Chiarelli, A. (2020) and Dawson, A., Creation of realistic virtual aggregate avatars. Powder Technology. 378, 760-771. Accepted.

- S. Michot-Roberto a, A. Garcia-Hernández, S. Dopazo-Hilario, A. Dawson. (2021). The spherical primitive and perlin noise method to recreate realistic aggregate shapes. Granular Matter. Accepted.

- Michot-Roberto, S., Garcia-Hernández, A., Dopazo-Hilario, S., Dawson, A. (2021). The use of a physic engine to simulate aggregate packing. Construction and Building Materials. Submitted.

- Michot-Roberto, S., Garcia-Hernández, A., Dopazo-Hilario, S., Dawson, A. (2021). Air void prediction model of porous asphalts. In process.

ABSTRACT

Granular materials are present in almost every field of science, from the pharmacy industry to civil engineering. Their study has been the objective of numerous research to understand better how the particle properties influence final material quality. Particularly in the engineering field, a more in-depth study of stone-based materials is important, as the effect of the grain's size and shape and their arrangement and orientation influence their workability, durability, and mechanical properties.

Nevertheless, the current models of granular materials have been unable to deal with some drawbacks, as the generation of realistic randomised 3D particles. This is due to the lack of mathematical models to represent the irregular shape of aggregates, being those oversimplified by primitive or convex polyhedrons; or the computational cost involved in computing the interaction between the numerous particles of a granular assemblage, because to most of the current numerical models require prohibitive consumer-grade hardware when entering complex shape; especially when those have concavities.

To overcome those barriers, a virtual physical model for aggregate's assemblies at the mesoscale level has been developed using an impulse-based discrete element numerical method, commonly used for video game and animation industries, which is implemented in a physics engine. Based on a simple analysis of aggregates through Image Analyse software, the model can generate many particles with realistic, complex shapes (including concavities) and size distribution quickly and automatically.

Another important contribution is the creation of a new simulations approach of aggregate's packing that can be used to simulate stone-based materials with specific

applications such as asphalt's aggregate structures, ballast or any other similar granular system, providing a starting point for future research.

Although the model is still at its very beginning, the results obtained from the simulations are quite close to the experimental ones, which indicates its ability to be used as a design tool for granular materials. However, there are still some limitations to overcome, as hardware and software capabilities, which will be overcome with further technological advances.

KEYWORDS: Granular materials, Physics Engine, Aggregates; Simulations, Compaction, 3D virtual modelling; Morphological properties

TABLE OF CONTENTS

DEDICATION iii

ACKNOWLEDGEMENTS iv

LIST OF PUBLICATIONS v

ABSTRACT vi

TABLE OF CONTENTS viii

DECLARATION xiv

LIST OF TABLES xv

LIST OF FIGURES xviii

CHAPTER 1. INTRODUCTION 1

1.1 A summary of granular material´s simulation 1

1.2 Scope and Outline 3

CHAPTER 2. LITERATURE REVIEW 7

2.1 Importance of size and shape of aggregates in stone-based materials 7

2.1.1 Gradation of Coarse Aggregate. Importance in civil engineering 7

2.1.2 The shape of Coarse Aggregate. Importance in civil engineering 10

2.1.3 Summary and conclusion 12

2.2 Different methods for the aggregates characterization 14

2.2.1 Conventional laboratory methods 15

2.2.2 Static Digital Image Analysis methods 19

2.2.3 Dynamic Digital Image Analysis methods 22

2.2.4 Particle reconstruction and their 3D index characterization 24

2.2.5 Conclusion 27

2.2.6 Particle´s shapes indices used in this research 27

2.3 Background of study of granular materials 32

2.3.1 Experimental works 33

2.3.2 Computer simulation works 34

2.3.3 Indices to quantify the granular material used in this study 49

2.4 Theory of the Physics Engines (PE). The Nvidia PhysX engine 53
2.4.1 Basics Structure 53
2.4.2 Collision detection 57
2.4.3 Collision resolution. Constrained Rigid Body Simulation 61
2.5 Techniques employed for modelling virtual aggregates. 69
2.5.1 3D random particle´s generation models 71
2.5.2 Conclusion 76
2.6 Conclusion from the literature review 76

CHAPTER 3. EXPERIMENTAL WORK 80

3.1 Particle´s samples characterization 81
3.1.1 Image-based method to quantify Coarse Aggregates samples 82
3.1.2 Results. Statistical parameters of the measured shape factors 85
3.2 Preparation and analysis of the real aggregates packing specimen 87
3.2.1 Specimens' preparation and compaction method 88
3.2.2 Internal Structure factors used for the aggregate assembly characterization 90
3.2.3 Results. 95
3.3 Preparation and Air Void measure of real asphalt specimens 104
3.3.1 Asphalt mixture specimens design 105
3.3.2 Mixing and compaction 107
3.3.3 Measurement of bulk density and Air Void content 109
3.3.4 Results 110
3.4 Summary of the experimental program 112
3.4.1 Particle characterization 112
3.4.2 Preparation and analysis of the real aggregates packing specimen 112
3.4.3 Preparation and Air Void measure of real asphalt specimens 114

CHAPTER 4. A 3D AGGREGATE´S GENERATOR MODEL WITH REALISTIC SHAPES 115

4.1 Introduction 116
4.2 Experimental measurement of aggregates' morphological properties 118
4.3 3D aggregate generation model 119
4.3.1 Creation of irregularly shaped particle with different level of angularity 121

4.3.2 Particle rescaling to adjust their minor Feret and Aspect Ratio to that of real aggregates. 126
4.3.3 Summary of the inputs parameters which define the size and shape of virtual aggregates 128
4.4 Results 128
4.4.1 Input parameter´s setting to generate different level of angularity 129
4.4.2 Comparison of the virtual data with the experimental data 131
4.5 Conclusion 136

CHAPTER 5. THE USE OF A PHYSIC ENGINE FOR THE SIMULATION OF A PARTICLE´s PACKING PROCESS 138

5.1 Introduction 139
5.2 Experimental analysis of the real aggregates packing specimens 140
5.3 Unity 3D game engine platform 141
5.4 Parameter´s optimization and configuration of Unity 3D 142
5.4.1 The Rigid Body parameters 144
5.4.2 The Physics Setting parameters 147
5.4.3 The Time Manager parameters 149
5.4.4 Summary of the Unity 3D Parameters configuration 151
5.5 Particle´s packing simulation process 151
5.6 Results and discussion 153
5.6.1 Visual comparison and virtual mixtures' number of particles 153
5.6.2 Difference between the internal structures of the real and virtual particles packing 156
5.6.3 Comparison between the real and virtual Air Void´s distribution curves 158
5.6.4 Comparison between the real and virtual particle´s segregation and orientation 160
5.7 Conclusion 161

CHAPTER 6. POROSITY PREDICTION MODEL FOR POROUS ASPHALT MIXTURES THROUGH A PHYSICS ENGINE 163

6.1 Introduction 164
6.2 Virtual aggregate generation model optimisation 166
6.2.1 Differential Evolution Description 168
6.2.2 Inputs parameters setting 171
6.3 Experimental analysis 174

6.4 Virtual production of asphalt solid skeletons 175
6.4.1 Input variables and porosity calculation 178
6.4.2 Determination of the maximum virtual force 180
6.5 Results 182
6.5.1 Validation of the DE optimisation for the aggregate generation model 182
6.5.2 Maximum virtual compaction force 184
6.5.3 Porosity prediction model validation. Comparison between experimental and virtual data 188
6.6 Conclusion 200

CHAPTER 7. CONCLUSION AND RECOMMENDATION FOR FUTURE RESEARCH 203

7.1 Introduction 204
7.2 Summary of findings and recommendations 205
7.2.1 3D aggregate generator model with realistic shapes 205
7.2.2 The use of the physics engine for the simulation of the particles packing process 207
7.2.3 Porosity prediction model for porous asphalt mixtures through a physics engine 208
7.3 Final conclusion 210

CHAPTER 8. POSSIBLE FUTURE APPLICATIONS 212

8.1 Aggregate´s generator model with realistic shapes 212
8.2 The use of a physic engine for the simulation of a particle´s packing process 212
8.3 Porosity prediction model for porous asphalt mixtures 213

REFERENCES 215

APPENDIX A. EXPERIMENTAL WORK 240

A.1 Particle´s samples characterization 240
A.1.1 ImageJ procedure 240
A.1.2 Anderson-Darling adjustment though the software Minitab 241
A.1.3 Iterative least squares technique to find the Weibull statistic parameters 246
A.1.4 Analysis of the Particle Size Distribution through different techniques 248

A.2 Analysis of the real aggregate´s packing specimens 258
A.2.1 Aggregate´s segregation and orientation of the various size aggregate´s packings specimens 258
A.3 Analysis of the real asphalt mixtures specimens 262
A.3.1 Aggregate´s gradation of the asphalt mixtures 262
A.3.2 Bulk density and Air Void content measurement procedure 262
A.3.3 Air void and standard deviation obtained from the sealed asphalt specimens and the SGC. 265

APPENDIX B. VIRTUAL PARTICLE GENERATION MODEL 270

B.1 Volume Reconstruction from implicit shape´s representation 270
B.1.1 The marching Cubes algorithm 271
B.2 Video of the creation of different type of particles 275
B.3 One sample T-test for the virtual aggregate generation model validation 276
B.4 Experimental and simulated cumulative frequency distribution curves 281

APPENDIX C. VIRTUAL PARTICLE PACKING 294

C.1 Results from different collision meshes 294
C.2 Video of a particle packing process 295
C.3 Virtual compaction parameter´s configuration 296
C.4 Slicing process of the particle´s assembly with Blender 297
C.5 Experimental and simulated Air void distribution curves 299
C.5.1 Visual comparison between real and simulated packed aggregates 299
C.5.2 Experimental and simulated Air Void distribution curves 301

APPENDIX D. ASPHALT SKELETON COMPRESSION SIMULATION 303

D.1 Video of the virtual compaction with rotational movement 303
D.2 DE optimisation algorithm 303
D.3 Real and virtual stone´s photos used for the mixtures manufacture 306
D.4 Validation of the DE optimization implemented on the aggregate generation model 307
D.4.1 Cumulative frequency distribution curves after the DE optimization 307
D.4.2 One sample T-test for the virtual aggregate generation model validation 313

D.5 Analysis of the virtual compaction force in the model 316
D.5.1 Virtual force versus Air void content 316
D.5.2 Virtual force versus particle's overlapping 318
D.5.3 Virtual force versus the number of contacts 319
D.6 Analysis of the of the Air Void prediction model 320
D.6.1 Multiple regression analysis with three variables 322
D.6.2 Multiple regression analysis with two variables 325
D.6.3 Pearson Correlation between all the variables analysed in the model 328

APPENDIX E. ATTACHED CPDES AND CODE SOURCE 330

E.1 Particles samples generation with complex shapes 330
E.2 Particle´s volume measurement and vibration of particles packing and 330
E.3 Compression of the aggregate´s skeleton of porous asphalt 331
E.4 Particles creation. Chapter 4 331
E.4.1 Particle creation with the Marching Cube Algorithm 332
E.4.2 Particles measurement with the Raycast function 343
E.5 Particles assembly. Chapter 5 349
E.6 Asphalt skeletons compression. Chapter 6 351
E.6.1 Fitness Function of the Differential Evolution algorithm 351
E.6.2 Calculation of the air voids content 355
E.6.3 Pistons movement 356
E.6.4 Discarded script of the pistons movement with rotation and angle of gyration 360

DECLARATION

The research reported in this thesis was conducted at the University of Nottingham, Department of Civil Engineering, Nottingham Transportation Engineering Centre (NTEC), between June 2017 and September 2021.

I declare that the work is my own and has not been submitted for a degree at another university.

Sonia Elisa Michot Roberto.

LIST OF TABLES

Table 3-1 – Morphological parameters for particle´s characterization 83

Table 3-2 – Experimentally measured shape factors that characterise the aggregates studied 85

Table 3-3 – Parameters to characterize the internal structure of the specimens 89

Table 3-4 – Particle´s density, number of particles in the mixtures and, height of the mixtures 94

Table 3-5 – Void and particle characteristics derived from CT-scans for the particle´s mixtures 95

Table 3-6 – Pearson's correlation coefficients between all the parameters studied 95

Table 3-7 – S_L Values for Aggregate's Vertical and Radial Segregations 101

Table 3-8 – Average Orientation Angle of Aggregates, ε^{o} 103

Table 3-9 – Summary of the asphalt mixtures designs 106

Table 3-10 – Results from the sealed specimen´s laboratory test and from the gyratory compactor´s calculation 109

Table 4-1 – Input parameters to control size and shape of virtual aggregates . 127

Table 4-2 – R_{index} and percentage of error values resultant from comparing the simulations with experiments 131

Table 5-1 – Physics' parameters configuration in Unity 3D 150

Table 5-2 – Comparison of the real and simulated number of particles 153

Table 5-3 – R values resultant from comparing the simulations with experiments 156

Table 5-4 – Ds and $Dcrit, 0.05$ of experimental and simulated vertical porosity curves 158

Table 5-5 – S_L Values for vertical and radial segregations for the virtual specimens 159

Table 5-6 – Average orientation angle, ε^{o}, for the virtual specimens 159

Table 6-1 – Boundaries Constraints of the Parameters to optimise. 172

Table 6-2 – DE parameters setting 174

Table 6-3 –Input to simulate the different porous asphalt mixtures 179

Table 6-4 –R_{index} and percentage of error values resultant after the DE optimization 183

Table 6-5 –Results of three simulation tests of the computed height (H) and porosity (AV) 190

Table 6-6 –Results of the air void prediction model 191

Table A-1 – Statistical parameters of the Weibull distribution curves 255

Table A-2 –Average Gradient ((εº)/mm) of Particle Orientation Angle versus Particle Size For the radial direction 260

Table A-3 – Aggregate grading and binder content used to produce porous asphalt mixtures 262

Table A-4 – Results of Mix 1 265

Table A-5 – Results of Mix 2 265

Table A-6 – Results of Mix 3 266

Table A-7 – Results of Mix 4 266

Table A-8 – Results of Mix 5 266

Table A-9 – Results of Mix 6 267

Table A-10 – Results of Mix 7 267

Table A-11 – Results of Mix 8 267

Table A-11 – Result of Mix 1 268

Table A-12 – Result of Mix 2 268

Table A-13 – Result of Mix 3 268

Table A-14 – Result of Mix 4 268

Table A-15 – Result of Mix 5 269

Table A-16 – Result of Mix 6 269

Table A-17 – Result of Mix 7 269

Table A-18 – Result of Mix 8 269

Table B-1 – T-test results with 95% Confidence Interval for Minor Feret 278

Table B-2 – T-test results with 95% Confidence Interval for Aspect Ratio 279

Table B-3 – T-test results with 95% Confidence Interval for Area 279

Table B-4 – T-test results with 95% Confidence Interval for Perimeter 280

Table B-5 – T-test results with 95% Confidence Interval for Roundness 280

Table B-6 – T-test results with 95% Confidence Interval for Circularity 281

Table C-1 – Comparison of results for the different collision meshes for sample G1 295

Table C-2 – Virtual compaction parameter´s configuration 296

Table C-3 – Void and particle characteristics derived from the ImageJ analysis of the virtual particle´s packed. 301

Table D-1 – T-test results with 95% Confidence Interval for Perimeter 314

Table D-2 – T-test results with 95% Confidence Interval for Area 314

Table D-3 – T-test results with 95% Confidence Interval for Roundness 315

Table D-4 – T-test results with 95% Confidence Interval for Circularity 315

Table D-5 – Regression analysis of the virtual force versus Air void content. 316

Table D-6 – Regression analysis of the virtual force versus the particle's overlapping 318

Table D-7 – Regression analysis of the virtual force versus the number of contacts 319

Table D-8 – Statistical significance of the independent variables and model summary 322

Table D-9 – Statistical significance of the independent variables 326

Table D-10 – Analysis of variance ANOVA 327

Table D-11 – Pearson Correlation of the simulation model factors 328

LIST OF FIGURES

Figure 2-1 – Typical Coarse Aggregates gradation 8

Figure 2-2 –Aggregate packing with low and high angularity respectively 10

Figure 2-3 –Width, Length & Thickness of a coarse aggregates 17

Figure 2-4 – Scheme of the analyses of a fractured Particle with one Fractured Face 18

Figure 2-5 –Aggregate Imaging System (AIMS) 20

Figure 2-6 –VDG-40 analysis process 22

Figure 2-7 –The Camsizer device 24

Figure 2-8 –Large-scale rock reconstruction 26

Figure 2-9 –Measurements of a stone in its more stable position by the Image Analysis software's: Camsizer and ImageJ. 29

Figure 2-10 –The Krumbein and Sloss chart classification of coarse sediment particles 32

Figure 2-11 –An examples of virtual beds with spherical particles 37

Figure 2-12 –Angle of repose simulation 39

Figure 2-13 –Virtual specimen of coarse aggregate skeleton in 2D 41

Figure 2-14 – a) Clump templates aggregate´s library. b) Dense compacted stone mixture sample 42

Figure 2-15 – Oedometer test simulation using convex polyhedron with different aspect ratio 42

Figure 2-16 –Two views of different representative aggregate shape´s library in BLOKS3D 44

Figure 2-17 –3D random aggregate geometric model with gradation considered 45

Figure 2-18 –Vibration of 3D random aggregate´s sample with PhysX engine 46

Figure 2-19 –Top and front view of the sample generated in Bullet 48

Figure 2-20 –Random dense particle´s assemblies with convex irregular shapes with different level of Perlin noise 49

Figure 2-21 –a) DEM´s contact model and b) PE contact model 55

Figure 2-22 –Basic structure of a Physique Engine 57

Figure 2-23 –Difference between DCD and CCD for two same objects colliding 60

Figure 2-24 –A collision representation of two aggregates modelled as rigid bodies: *i* (left) and *k* (right), respectively 66

Figure 2-25 – Virtual particle creation through the extrude process from real 2D images stones views 71

Figure 2-26 – Virtual Particle generation through Voronoi Tessellation procedure 73

Figure 2-27 – Virtual particle aggregate created using the QuikHull algorithm 74

Figure 2-28 – Virtual particle aggregate created using the Convex Hull algorithm 75

Figure 2-29 – Creation of convex irregular particle´s using a Perlin noise 76

Figure 3-1 –Pictures of some stone´s type analysed 82

Figure 3-2 –Pictures of some Aggregate chart classification based on the IA Sphericity and Circularity Index and the Krumbein chart, used for a visual evaluation 85

Figure 3-3 –Gradation curves of a) single-size aggregate´s mixtures and b) various-size aggregate´s mixtures 87

Figure 3-4 – Mixture (a) before the compaction; (b) after compaction 89

Figure 3-5 – a) different type of particle orientation. b) A Measurement of the orientation angle 93

Figure 3-6 – Aggregate´s assemblies with same porosity but different tortuosity 98

Figure 3-7 – Air Void percentage distribution along the height of the samples 100

Figure 3-9 – Summary of the gyratory locking points 108

Figure 4-1 – Flow diagram of the virtual particle's generation process 120

Figure 4-2 – Irregular particle´s creation chart 121

Figure 4-3 – Example of un-realistic and realistic virtual aggregates 124

Figure 4-4 – Raycast operation to detect holes by mapping the surface of the aggregates. 124

Figure 4-5 – a) The 3 views of a virtual particle and b) its measurement in its more stable position. 126

Figure 4-6 – Virtual imitation of the Krumbein and Sloss chart classification 128

Figure 4-7 – Visual comparison between real and simulated particles 129

Figure 4-8 – R square of the different shape´s factor 133

Figure 4-9 – Frequency distribution of minor Feret, area and roundness for the G2 aggregate sample 134

Figure 5-1 – Visual Particle with a) convex mesh collider; b) concave mesh collider with intermediate precision; c) concave mesh collider with high precision. 146

Figure 5-2 – Time step values VS Frame per Seconds for a gradation with 2300 particles 149

Figure 5-3 – Sequence of the virtual compaction process. 152

Figure 5-4 – Comparison between real and simulated packed aggregates. 154

Figure 5-5 – Distribution of vertical porosity for (a) G1 and (b) GS mixtures. 159

Figure 6-1 – How changes in the area and perimeter change particles angularity 168

Figure 6-2 – Flow chart of the DE optimization process 169

Figure 6-3 – Sequence of the virtual asphalt skeleton compaction process 178

Figure 6-4 –R square of the different shape´s factor after the optimization 184

Figure 6-5 –Regression graphs of a) the simulated air void, b) the overlapping, and c) the number of contacts as a function of the virtual compaction force. 185

Figure 6-6 – Perceptible penetration between particles in Mix 7 specimen compacted at 370Kpa 188

Figure 6-7 – Correlation between predicted and simulated porosity 191

Figure 6-8 –Multiple linear regression analysis response surface of the air voids errors with the number of aggregates and volume of mastic 193

Figure 6-9 –Comparison of the vertical air void distribution between the experimental and simulated Mix 3 195

Figure 6-10 – (Left) real asphalt sample of Mix 3 (~13.75% air voids). (Right) real asphalt sample of Mix 3 (~14.5% air voids) 196

Figure 6-11 –Virtual asphalt simulated 197

Figure A-1 – Original image (left) and binary image (right) of a sample of limestone (10mm) placed on a clean surface free of dirt 240

Figure A-2 – Original Original image (left) and binary image (right) of a sample of limestone (10mm) placed 241

Figure A-3 – Original Comparison of distribution models accuracy for the Minor Feret diameter 243

Figure A-4 – Cumulative distributions of the shape parameter Minor Feret for all the particle´s samples on Weibull plots with fitting lines 243

Figure A-5 – Comparison of distribution models accuracy for the shape´s factor Aspect Ratio 245

Figure A-6 – Original – Cumulative distributions of the shape parameter Aspect Ratio for all the particle´s samples on Weibull plots with fitting lines 245

Figure A-7 –Comparison between the different PSD curves obtained with the different techniques 254

Figure A-8 – The aggregate distribution in vertical and radial direction for the three specimens 259

Figure A-9 –Aggregate´s Orientation Angle, ε^{o}, as a function of the aggregate´s size 261

Figure B-1 – Marching Cube cases from the lookup table 275

Figure B-2 – A spherical Particle created in Unity 3D software using the Marching Cube 275

Figure B-3 – Minor Feret distribution curves 282

Figure B-4 – Major Feret distribution curves 284

Figure B-5 – Area distribution curves 286

Figure B-6 – Perimeter distribution curves 288

Figure B-7 – Roundness distribution curves 290

Figure B-8 – Circularity distribution curves 292

Figure C-1 – Layer perpendicular to a virtual particle´s assembly 298

Figure C-2 – 2D intersection between the stones and the layer 298

Figure C-3 – Real and virtual particle´s assemblies and they corresponding 2D slices for samples: a) Crushed Glass (CG) and b) Limestone 1 (L1) 299

Figure C-4 – Visual comparison between real and simulated packed aggregates 300

Figure C-5 – Air void curve of the different real and virtual mixtures 302

Figure D-1 – Comparison between the real and virtual limestone samples 306

Figure D-2 –Area curves distribution 307

Figure D-3 –Perimeter curves distribution 309

Figure D-4 –Roundness curves distribution 310

Figure D-5 – Circularity curves distribution 312

Figure D-6 – Different Linear regression graphs of the air void as a function of the virtual force. 317

Figure D-7 – Different Linear regression graphs of the overlapping as a function of the virtual force. 319

Figure D-8 – Different Linear regression graphs of the number of contacts as a function of the virtual force. 320

CHAPTER 1. INTRODUCTION

The morphological properties and aggregate gradations are of enormous importance for the durability of the granular materials used in civil engineering. The morphological properties are related to the aggregates assembly and, therefore, to the final product's quality. Nevertheless, aggregates' morphological properties are not entirely taken into account in current design practices because aggregate characterization is difficult. Besides, Computer Tomography (CT) can be used to aggregates characterization; however, its use is limited due to high economic cost. Furthermore, in civil engineering applications, such as concrete or asphalt, there is no standard to determine how the shape of aggregates influence the material's behaviour [1]. For these reasons, granular material simulation is an active research area to understand how changes in the grain's gradation and shape affect the material's behaviour.

1.1 A summary of granular material´s simulation

Numerical methods allow simulating granular materials through sophisticated software based on Discrete Element Modelling (DEM), which considers the granular system's discrete nature. Each grain collision must satisfy the conservation laws for linear and angular momentum (Cundall and Strack, 1979). Nevertheless, the characterisation and simulation of the morphological properties of particles are currently time-consuming or expensive. On the other hand, since a granular system can be composed of many particles, DEM simulations may require a high computational cost to compute their interactions. For these reasons, numerical models to simulate granular materials are normally simplified as an ideal medium consisting of spherical particles, representing the real grains [2].

Recently, new algorithms have been implemented on the DEM methods to improve their performance and to include nom spherical paparticlesFor example, the MIMES DEM simulation package incorporates superquadric particle shapes [3], and the DEM software, BLOKS3D, provides polyhedral shapes [4]. BLOKS3D enables the users to define the particles' morphological properties by using imaging-based shape indexes [1]. However, in these algorithms, it is necessary to compute each aggregate's shape by adjusting three 2D views, a laborious and time-consuming task. Another disadvantage of the DEM methods is that they do not consider the particles' concavities since their contact detention algorithms become more complex with the grain shape's complexity, and the computational cost increases dramatically.

A true understanding of how the size and shape of the grains influence the assemblies they form is required to design new materials and, this can be only reached practically through modelling. For this reason, novel and robust DEM methods are gaining popularity among researchers. In reference [5], Physics Engines (PE's), which have been used mainly to develop video games and animations, and include a type of simplified discrete element numerical methods, are proposed as a DEM's alternative.

PE's were created for real-time physics calculations with high computational efficiency that prevails over accuracy. However, by adjusting and optimising the collision algorithms' configuration and their iterative solver, finding a proper balance between efficiency and precision is possible. Nonetheless, it is still unclear how useful they are for conducting scientific simulations. Several researchers have started to use PEs to model complex shapes and simulate their assembly. For instance, in [6], convex polyhedrons are modelled as coarse aggregates using the Quickhull algorithm [7], and particles are imported to the physic engine swig-Bullet multi-body, based on the Monte

Carlo method [8], where the movements of aggregates in asphalt concrete during compaction were simulated. Another work that uses a PE to model granular soils can be seen in [9]. In this study, pluviation and vibration are simulated using the Bullet physics engine, where the gravel is represented through randomly shaped polyhedrons created with Voronoi's tessellation method. Reference [10] shows simulations of crushable granular media under load by combining the peridynamics method to analyse and simulate particles' breakage with Bullet PE to simulate the particles and their interaction [11]. Reference [12] demonstrates the Bullet physics engine's feasibility by validating, with experimental tests, the results of the simulations of direct shear testing on uniform-sized metal beads. The most recent work is found in reference [13], where realistic particles and a soil specimen collapse's process are simulated through the Bullet Physics Engine.

This work aims to demonstrate the capability of PEs to generate realistic aggregate packings by modelling particles with complex shapes, taking into account the concavities of aggregates. The packing's characteristics, such as porosity, were evaluated and compared with experimental results.

1.2 Scope and Outline

This study is intended to simulate coarse aggregates' packing, representing the solid skeleton of asphalt mixtures, using a physics engine, and validating laboratory tests results.

This thesis relates to the granular material simulation at the mesoscale level. A novel algorithm able to model realistic particles and their packing fast using a Physics Engine has been developed. Several experimental tests were carried out to compare the experimental specimen's packing characteristics with the simulated one. This could

enhance the predictive capability in civil engineering mixture design since the aggregates blend design is closely related to stone-based mixtures' performance. Accordingly, the Thesis can be split into three main parts, which correspond to the three journal papers written in this Thesis:

I. Paper 1. *3D The spherical primitive and Perlin noise method to recreate realistic aggregate shapes*. Section 3.1 and Chapter 4.

   An algorithm that can generate Coarse Aggregate with different morphology and size distribution by implementing statistical information from real stone's samples characterisation is developed. The real stone's samples were characterised through an Image-based method. The virtual particles were modelled from spheres using noise functions and the Marching Cube Algorithm to create realistic stone morphologies, including concavities. The algorithm allows the creation of a large number of stones in a matter of seconds. The validation was done by comparing the virtual morphological features of aggregates with the experimental ones.

II. Paper 2. *The use of a physic engine to simulate aggregate´s packing*. Section 3.2 and Chapter 5.

   A 3D packing simulation model is presented, which mimics an experimental compaction process. This Chapter reports the simulation of several aggregate packing comprising single-size and various grain sizes. Their internal structures were quantified for a comparison with their corresponding actual samples, which were scanned with a Computed Tomography scan, to demonstrate the viability of the proposed model.

III. Paper 3. *Porosity prediction model of porous asphalt mixtures through a physics engine.* Section 3.3 and Chapter 6.

Asphalt's skeleton virtual designs are proposed to predict the Air Void Content of several porous asphalts based on volumetric calculations. The Physics Engine simulation's porosities were verified by comparing them with the data obtained from a Superpave Gyratory Compactor of equivalent asphalts. Additionally, an optimization is implemented in the Aggregate generation model developed in paper 1, using a Differential Evolution Algorithm, under the purpose of fine tune the model and creating a more computationally efficient tool.

Below we outline the contents of the chapters.

Chapter. 2. Literature review.

Basic concepts of several topics are introduced: 1) Background of Coarse aggregate characterisation; 2) Background of Packing of aggregates characterisation; 3) Theory of the Physics Engine; 4) Different techniques used for modelling virtual aggregates. Throughout these topics, the most appropriate index for the aggregates and packing characterisation and the most suitable simulation tools are discussed, according to the instrument available and economic feasibility.

Chapter. 3. Experimental work.

Several experimental tests have been carried out to create and validate the virtual models developed in this research. These experimental works are described according to their use in each phase of the project.

Chapter. 4. A 3D aggregates generator models with realistic shapes.

An innovative simulation model of virtual aggregates with realistic shapes is presented. The main goal is to develop a tool capable of generating many stones in a matter of seconds, which follow a distribution of sizes and shapes realistically.

Chapter. 5. The use of a physic engine to simulate aggregates packing.

A 3-D particle packing model is developed to corroborate the physics engine's capability to simulate granular materials and, the simulations are validated with experiments.

Chapter. 6. Porosity prediction model of porous asphalt mixtures through a physics engine.

Once the physics engine capability to simulate granular material has been corroborated in the previous chapters, Chapter 6's goal is to create an asphalt skeleton's compaction simulation tool to predict the air void content of the porous asphalts. In addition, the aggregate generation model presented in Chapter 3 was optimized by employing a Differential Evolution algorithm, to improve its performance as well as to obtain more accurate results.

Chapter. 7. Conclusion and recommendations.

The main conclusions of the work are presented, along with recommendations for further investigations.

Chapter. 8. Possible applications.

Several futures application with the virtual model approaches presented might be possible, always having in mind that those are subjected to theance in technologyand that further research are necessary.

CHAPTER 2. LITERATURE REVIEW

2.1 Importance of size and shape of aggregates in stone-based materials

The morphological features of aggregates are important factors to consider to understand the stones-based materials performance, such as concrete or asphalt, due to their influence on the particle´s assembly.

Nowadays, several methods for the particle´s shape and size characterization exist, in addition to the traditional manual methods. For instance, digital methods have become very popular since they can measure the grains' properties in a more accurate and faster way.

This section explains the importance of gradation and the shape of aggregates in civil engineering. The different existing methods to measure the particles physical properties are presented, differentiating between manual and digital techniques. Finally, a description of the shape indexes used in this study is presented.

2.1.1 Gradation of Coarse Aggregate. Importance in civil engineering

According to the American Society for Testing and Material (ASTM), gradation is defined as the particle size distribution in a set of aggregates. Different gradations are chosen according to the material´s design requirements and economy (i.e., a highervoid between particles requires more binder or mortar amount, which leads to a more expensive design).

The traditional way to measure the size distribution is through sieve analysis. The coarse and fine aggregates are normally sieved separately, being the limit between them 4.75mm (No. 4) or 2.36mm (No. 8), depending on the regulations and requirements

[14]. The gradations are presented graphically, as the cumulative mass passing through sieves to evaluate if there is an excess or lack of certain particle sizes [15]. In this paper, we will focus on coarse aggregates, since the virtual modelling of fine particles is not part of this work.

There are three typical coarse aggregate´s gradations: well-graded grading, which includes all the stone´s size ranges; poor graded, which has a uniform gradation without changing the particle size; and gap-graded, which omits the size range of medium-sized particles (Figure 2-1).

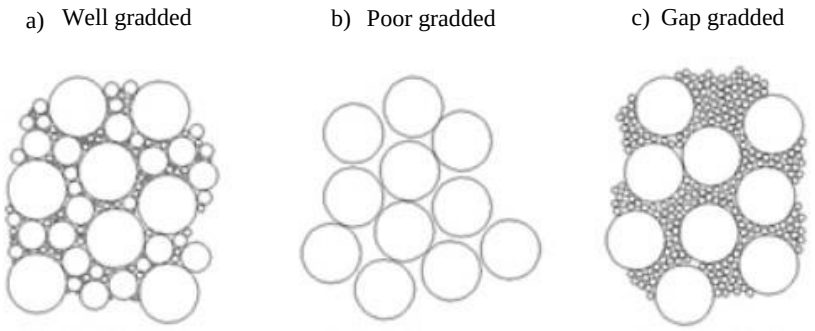


Figure 2-1 – Typical Coarse Aggregates gradation

Note. Image from: The role of aggregate in concrete countertop mix formulas [16].

A proper gradations choice is important since it is known to affect stability, durability, permeability and workability of the stone´s based materials, amount other properties. Otherwise, undesired effects could appear on the material´s performance. For instance, an excess of coarse aggregates could yield compaction problems [17]; on the other hand, a lack of coarse can produce rutting due to weak internal friction between particles, leading to stability problems [18].

Examples of aggregate-based materials for civil engineering are bituminous mixtures, ballast, and concrete.

Regarding the bituminous mixtures, an open-graded asphalt, such as StoneMastic Asphalt (SMA), has better resistance to cracking than others since its mineral skeleton has more stone-stone contact, which provides good resistance to stress of traffic loads. It also provides greater resistance to moisture damage due to its higher binder content than the well-graded asphalt mixtures, which show fewer voids betweenparticles and dry mixtures with poor durability [19].

Nevertheless, for the aggregate ballast used in the railways, it is more convenient to use a well-graded gradation since they lead to less settlement [20]. Concretes mixtures normally present well-graded gradations that provide maximum density´s and, therefore, the greatest strength.

In addition to the gradation, two important parameters must be mentioned, as they also affect the material´s performance: the Maximum Size and the Nominal Maximum Size, (*ASTM* definition is cited in [21]).

For instance, in concrete, the Maximum Nominal Aggregate Size (NMAS)should not be greater than one-fifth of the narrowest dimension between the sides of concrete´s shape [16]. It is also recommended to use the largest possible nominal size to save money by reducing the cement paste required [15].

The maximum sizes of aggregates influence the strength of the material. For example, for concrete, for the same cement-water ratio, the smaller the maximum size, the greater s the resistance to compression. In [19], a study based on the effect of the maximum nominal size of the aggregates on SMA performance showed that increasing NMAS increases the mixtures' resistance, but decreasing NMAS helps prevent cracking. NMAS also affects other mixes' characteristics, such as permeability, since lowering NMAS lower permeability, as explained in [22].

Although there are specifications for the gradation and maximum aggregate size of the different stone-based materials, those are not fixed, and deeper studies are necessary to understand how their variation affects the different material´s behaviour. Further research about how the gradation and the size of aggregate affect the performance of asphalt, concrete and ballast can be found in [23][24][20], respectively.

2.1.2 The shape of Coarse Aggregate. Importance in civil engineering

The importance of the particle´s morphologies has been recognized as a key factor that affects the porosity and contact between particles and, therefore, the mechanical characteristics of civil engineering materials such as: asphalt, concrete, and ballast; as several investigations have demonstrated over the years, as can be found [25][26][27] respectively.

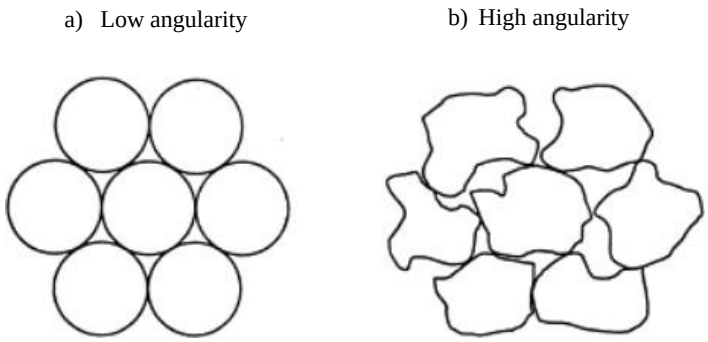


Figure 2-2 –Aggregate packing with low and high angularity respectively

Note. Image from: Sphericity, shape factor, and convexity measurement of coarse aggregate for concrete using digital image processing [28].

The aggregates' different angularity generates different assembly structures, (i.e., angular stones with concavities lead to aggregate´s assembly with more gap between grains than the rounded one), yielding to different mixtures behaviour (Figure 2-2). However, the particle´s shape's characterisation is difficult and subjective and their

classification could change according to the different standards and requirements demanded, according to the material in which they are used.

According to their shapes and how they could influence different civil engineering materials, a general description of aggregates is listed below.

Rounded aggregates**.** They are formed by natural erosion and present smooth surfaces [29]. These stones are generally used in concrete, requiring low resistance, as they produce a poor stone´s interlock and lower air void percentage than the angular aggregates. However, they provide good workability and rheological properties since the particles can move and rearrange themselves within the matrix in an easy way [15][30]. Nevertheless, for asphalt mixture, too much workability can lead to poor compaction since the particles tend to slip, causing distortion and shoving and particle segregation, which could increase the air void content and might produce premature damage due to greater water and air infiltration [31].

This kind of particles must be avoided for asphalt mixtures and ballast, and concrete which requires a high resistance due to their poor inter-particles friction, which can compromise the materials' stability.

Irregular or partly rounded aggregates. These grains are generated by attrition, and they give a slightly higher percentage of voids percentage than the previous ones, and therefore, a bit more resistant materials. However, they are still not appropriate for mixtures that require high strength [30].

Angular aggregates. They are obtained by a crushing process that produces well-defined stone’s edges. They give a high percentage of voids which leads to greater compressive strength.

Crushed aggregates are the most extensively employed civil engineering materials since they generate good internal friction between aggregates, thus, increasing their stability. Besides, their rough surfaces provide good adhesion between particles and binders or mortar, which influences permanent deformation [31][32].Regarding railways ballast, angular stones provide adequate drainage and proper loads distribution [33].

Flaky aggregates. An aggregate is often considered flaky when its thickness is around 60% less than its average size, although that definition might vary according to the different specifications [30].

Elongated aggregates. Stones are considered elongated when their length is 180% greater than their average dimension.

Flaky and elongated aggregates. They occur when the characteristics mentioned above are combined.

All the different civil engineering materials' regulations have restrictions regarding the percentage of the last three types of stones described above that can be used in the mixes. This is because they are easily breakable during the compaction process, resulting in a change in the mixes gradation and, therefore, in the void´s percentage, affecting the expected mixture´s performance [34][35].

2.1.3 Summary and conclusion

A summary of how the different shapes and particles size distribution of coarse aggregates might influence the stone-based material characteristics is presented below.

• Durability and Stability. Some of the most important qualities in civil engineering, since aggregates distribute the loads to which the materials are subjected respectively. Those factors are controlled mainly by the gradation, as it influences the stone-stone contact. Dense gradation of solid and resistant aggregates gives mixture with high-density packing, and therefore more durable [36].

Besides, contact between particles improves stability and resistance to degradation. For instance, and open-graded gradation in asphalt mixtures provides high resistance to shear deformation and cracking, as the mixtures' mineral skeleton is stable since the stones are not compromised by the finer particles [19].

Durability and stability also depend on the particles' shape and surface texture that influences the internal friction between aggregates. Stones without fractured faces might promote rutting since particle´s bind less securely [26].

• Permeability. The permeability depends directly on the air-void content in the mixtures controlled by the size distribution, the nominal maximum size, and the particle´s shape.

A high air voids percentage leads to more water infiltration, producing mixtures with insufficient strength and susceptible to premature ageing [37]. For instance, gap-graded gradation on asphalt mixture, which uses more binder, provides greater resistance to moisture damage than the well-graded asphalt mixtures. Mixes with smaller nominal maximum aggregate size tend to be more impermeable at the same void percentage [22]. Angular particles produce more impermeable materials, as demonstrated in [37].

• Workability. It depends on the gradation and the source of the aggregate. A high amount of coarse aggregates reduces workability since they might produce segregation during the materials' placement, which can difficult the compaction process. The same could occur when using rounded CA with smooth surfaces. However, its use is allowed in some type of concrete [15][30].

• Flaky and elongated particles. They are not used in civil engineering materials since they can break during the compaction process and produce low-quality materials [35].

In conclusion, the influence of the different stone´s shapes and particle´s size distribution on the mixtures should not be ignored since they are directly related to the material´s performance. However, as will be explained in the next section, the characterization and quantification of aggregate morphologies can be difficult, expensive or subjective. Finding the most suitable gradation for the different mixture´s requirements is difficult because of the numerous laboratory tests required, which are material and time-consuming.

For these reasons, one of this thesis's objectives is to create a virtual aggregates generation model in which particles follow realistic grain´s size distribution and morphological features. This could serve in the future to investigate how the particle's assembly properties change with the different stone´s types and gradations in a fast and practical way.

2.2 Different methods for the aggregates characterization

Numerous methods to measure the 3D and 2D geomorphological particle properties exist. Each one proposes different indexes to quantify the stone's size and

the stone´s shape at macro-, meso- and micro-scale levels, although also they present advantages and limitations. For instance, manual rock´s measurements give accurate information, but is a tedious task only useful for particle sizes greater than about 4mm [38]). At the same time, Digital Image Process techniques are accurate and simpler than manual ones, although they may require expensive equipment.

There are many shape factors and indices shape for the grain´s classification. In civil engineering these factors serve to ensure the minimum requirements that aggregates must have, to accomplish the indications given by the American Society for Testing and Materials (ASTM) (standards D2488 [39], D3398 [40] and D4791 [41]), and European Committee for Standardisation (CEN); EN-933-3 [42]; for the different materials.

In this section, the different methods to characterize the aggregate´s morphology are presented, differentiating between: manual methods, Digital Image Analysis methods and Dynamic Image Analysis methods, to have a global vision about their differences, advantages and limitations.

2.2.1 Conventional laboratory methods

Some of the most widely used methods existing in civil engineering regulations are described below.

ASTM standard:

- *D-3398. Standard Test Methods for Index of Aggregate Particle Shape and Texture.*

This test proposes a method to quantify the overall stone´s shape and the stone´s textures, and they indicate how they can influence the performance of the paving mixtures [43]. It consists of an indirect method based on the air voids content in the bulk aggregate by compacting each size fraction of the aggregates assembled in a standard cylindrical mould.

The angularity index is calculated with the weighted average of the angularity obtained from the various one-single size stones present in the sample. However, this method, in addition to being a bulk approximation, during the compaction by using a heavy weight, the stone´s shape could be artificially modified, thus giving errors on the results [43].

- *D-4791. Flat and Elongated Particles in Coarse Aggregates.*

This test procedure is implemented on the functional requirement in construction material´s design. The flat and elongated particles tend to fracture more easily under compaction, which might affect the Void in Mineral Aggregates required from the specification, as mentioned in the previous section. Besides, these types of aggregates fit and lock easily, which could cause difficulties during the compaction process [41].

The test procedure consists of the manual measurement of the three dimensions of the aggregates to calculate the ratio between them (Figure 2-3). One of the main drawbacks is the stones are measured one by one, in addition to being a tedious and time-consuming task, is that the particle´s size must be smaller than 80mm and bigger than 4mm.

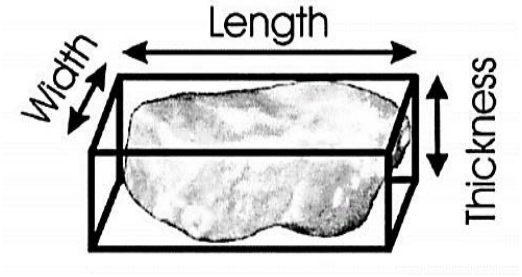


Figure 2-3 –Width, Length & Thickness of a coarse aggregates

Note. Image from: ASTM D-4791 flat particles, elongated particles, or flat & elongated particles [35].

- *D-5821. Standard Test Methods for Determining the Percentage of Fractured Particles in Coarse Aggregates.*

This test serves to determine the number of fractured faces of the particles in a sample, since a minimum percentage is required for construction materials, as they are related to the inter-particle friction and thus to the shear strength in the aggregate skeleton structures.

Different specifications might have different criteria about the number of fractured faces required on a fractured particle, and they also differ if they must be taken into account by the percentage by mass of the total sample or by the percentage by particle count [44].

The standard procedure of evaluating each particle individually is to count the number of fractured surfaces by visual inspection. A face is considered a "Fractured Face", only if the projected area of the fractured portion of the face (A_f) constitutes at least a quarter of the maximum cross-section area of the rock particle (X_{Max}) [44] (Figure 2-4). This test produces inaccurate results, and it is tedious and time-consuming. It can be used only for aggregates over the 4.75-mm (sieve No. 4).

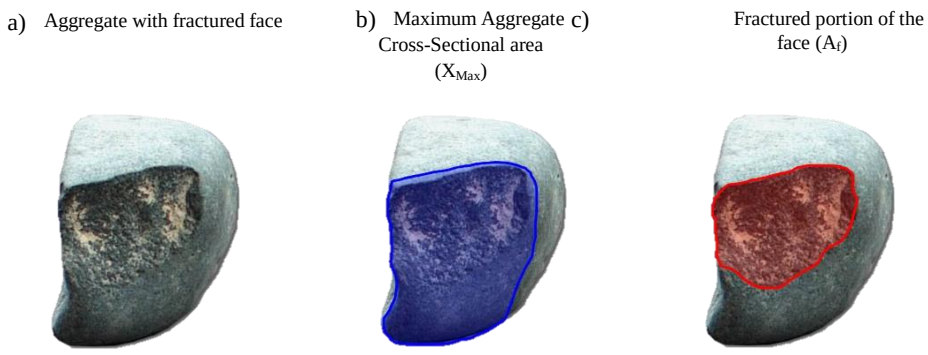


Figure 2-4 – Scheme of the analyses of a fractured Particle with one Fractured face

CEN standards:

- *EN-933-3. Test for geometrical properties of aggregate. Part 3. Determination of particle shape. Flakiness index.*

The biggest difference between this CEN standard and the ASTM D 4791 described previously (*Flat and Elongated Particles in Coarse Aggregates*), lies in that individual size fractions are sieved on special sieves, and the flakiness index is calculated by measuring the percentage of flaky particles of the total sample by mass, which pass. More details about this process can be found in [45].

- *EN-933-5/A1 Test for geometrical properties of aggregate. Part 5. Determination of broken surfaces in coarse aggregate particles.*

The method consists of the same procedure described for the American standard (ASTM D5821) and a manual and visual procedure.

- *EN 1097-3. Test for Mechanical and Physical properties of aggregates. Determination of loose bulk density and air voids.*

This method is the European version of the *ASTM C29/C29M. Standard Test Methods for Bulk Density ("Unit Weight") and Voids in Aggregates*, and uses the same procedure.

As seen in this section, several specifications for the aggregate´s characterization exist to evaluate stone samples and ensure that they meet the necessary mixing design criteria. Nonetheless, all these methods are manual, which could be laborious, subjective and time-consuming.

To reduce these drawbacks, numerous investigations have been carried out to develop new techniques based on image processing software, which streamline and improve particle characterization, as described in the following sections.

2.2.2 Static Digital Image Analysis methods

The digital advance has allowed the development of new methods to measure the size distribution and the morphological parameters of the particles. Each one uses different techniques and mathematical algorithms to evaluate: particle angularity, form indices or convexity, among other characteristics. Digital Image Analysis methods can measure more accurately and are less time consuming than the traditional manual methods. For these reasons, they have become very popular in recent years. A summary of the most popular DI methods can be found below.

One of the most popular image analysis methods is the Aggregate Imaging System (AIMS), developed by Eyad A. Masad in the Texas A&M University College Station, in cooperation with the Federal Highway Administration November in 2004 [46]. AIMS consist of a system equipped with a microscope, a light and software (Figure 2-5), able to analyse different properties of the morphology of both: coarse

aggregate and fines. A camera and different illumination serve to take images with different resolutions in 2D to provide the particle projections, from which the shape´s factors are measured [41].

The video microscope is used to measure the coarse aggregates' depth to obtain their third dimension. In addition to its capability to detect flat and elongate particles, the AIMS system has implemented an image analysis technique able to quantify: 1) the properties of the shape: angularity and form of the coarse and fine aggregate by using the fundamental gradient approach [47]; 2) the surface texture of the coarse aggregate by using wavelet methods [48].

The results are given by statistical analysis, based on the distribution of the shape characteristic in a sample and not by the average indices, which serve to evaluate, for example, how the different crushing methods affect the grain characteristics, or to create new aggregate specifications [46]. This method was recently improved and usedto develop new aggregate specifications, as shown in [49].

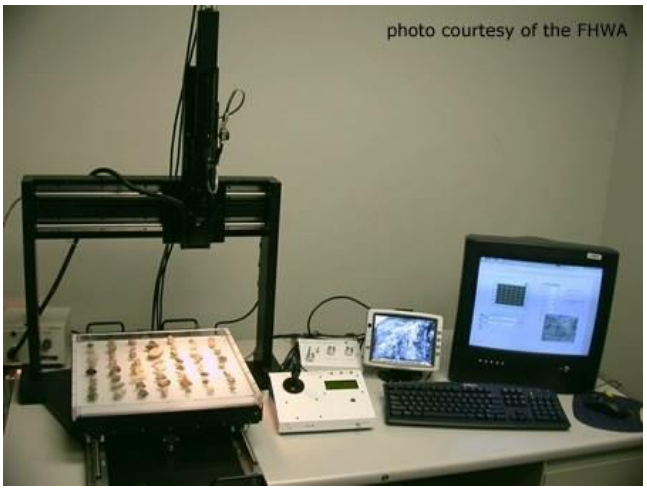


Figure 2-5 –Aggregate Imaging System (AIMS)
Note. Image from: characterization of aggregate resistance to degradation in stone matrix asphalt mixtures [41].

Another popular method to measure particle morphologies is: the University Of Illinois Aggregate Image Analyzer (UIAIA). It is a fairly simple method in which aset of coarse aggregates are photographed with a camera; then, a segmentation technique is applied to the images through algorithms that are incorporated in the UIAIA [50]. Image segmentation techniques are used to partition an image into severalsets of pixels to highlight the areas of greatest interest for its study [51]. Finally, the size and shape of particles are quantified.

This method also uses proprietary indices to evaluate the shape, angularity and texture properties of aggregates. The main difference from the previous method (AIMS) lies in that the images are only analysed in two dimensions. More information about the different indices the UIAIA and the AIMS systems use, can be found in [52].

In [53], the authors develop a method to measure each particle's three axial lengths in a coarse aggregate sample, consisting of captured images from the stones in two different positions: lying and standing. This technique takes high-quality images using a luminous layer of glow in a dark platform where the coarse aggregates are placed [38].

Other less known static methods for the 2D and 3D capture of the aggregate properties can be seen in [54][38].

Although these methods use relatively simple and not very expensive technology, their prototypes are not finished or expensive, making their acquisition very difficult. Furthermore, the indices they propose for the particle´s characterization indices are still pending acceptance for the standards used in construction materials.

2.2.3 Dynamic Digital Image Analysis methods

The most significant difference concerning the previous methods is that digital dynamic image analyses are automated methods in which the particles are moving through an optical microscope using high-speed flash illumination. Both methods have a similar way of capturing and analysing the aggregates. However, dynamic image analysis allows performing the analysis of a large number of particles in a more accurate and faster way. Some examples of these devices are described below.

One of the first Dynamic Digital Image Analysis to be commercialized is the French system VDG-40 Videograder. It was created mainly to determine the gradation of a sample of aggregates, it computes flatness and elongation factors or count the fractured faces

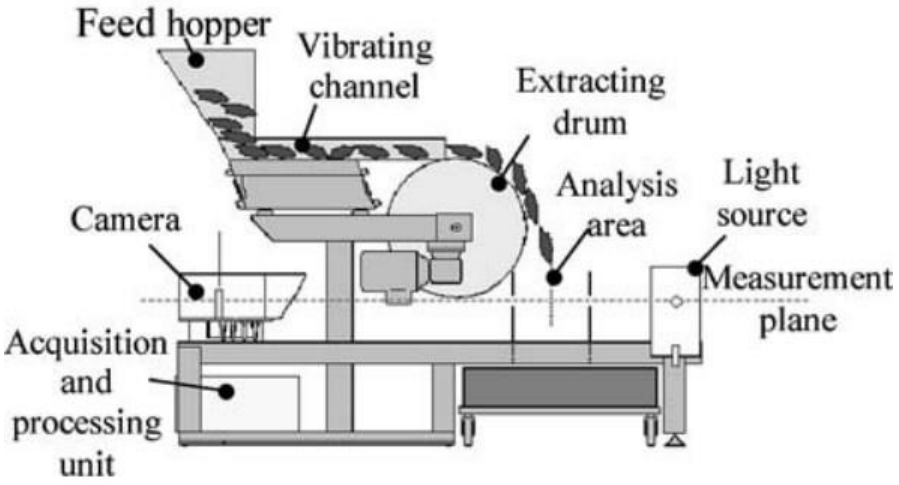


Figure 2-6 –VDG-40 analysis process

Note. Image from: Automated measurement of aggregate properties [42].

The stone´s images are captured using the shadowgraph principle, which consists of locating the particles between the light source and the camera to highlight the contrast between the object and the background [42]. An outline of the process used inthis system be seen in Figure 2-6. The camera takes successive photos of the stone´s

area while they fall, which can be monitored in real-time thanks to its acquisition and processing unit.

This system's biggest advantage is that it can perform an exhaustive analysis of stonesamples, ranging from 1 to 63 mm. However, the validation of the VDG-4A for the measurement of flat and elongated particles under ASTM D4791 was not finalized [55].

Similar systems to the VDG-40 are: The Micrometrics OptiSizer, the Video Imaging System (VIS), the Computer Particle Analyzer or the Buffalo Wire Works system, among others. Nevertheless, these methods are used mainly for particles sieving and shape approximation since they idealize the aggregates' shape without considering their angularity or surface texture [55].

A Dynamic system that addresses the evaluation of particle´s shapes including: roundness, sphericity and angularity, and the flatness, elongation volume of theparticle is the WipShape system [56]. The image of the individual stones are captured from two different front views and an Image technique is used to measure their morphological characteristics. This system is designed for a continuous automated operation and sorts the flat and elongated aggregates by different Aspect Ratios. Besides, it can measure large samples, thanks to a rotating circular table [57].

One of the newest and more sophisticated devices, used by many industries for evaluating grain´s both: fine and coarse, is the Camsizer device (Figure 2-7). It is a compact device created by *Retsch Technology*, which has two digital cameras capable of photographing both: fine and coarse particles at different resolutions, capturing more than 60 pictures per second [58]. The particles get through a funnel into a vibratory channel where they are transported until they fall. The software analyses the particle

size distribution and measures the grain´s morphological properties such as: sphericity, roundness, convexity and aspect ratio, among many other properties [57].

Camsizer also allows the storage of the images of each particle separately in a database. Results can be seen in references [58][59].



Figure 2-7 –The Camsizer device
Note. Image from: Particle Analyzer CAMSIZER ® [60].

As mentioned, Dynamic Image Analysis are automated methods that can determine the properties of the particle´s samples in a fast and accurate way. Nonetheless, their biggest disadvantage is they are expensive devices, not available in all industries or universities.

2.2.4 Particle reconstruction and their 3D index characterization

Thanks to new and sophisticated technologies, new methods to characterize the aggregates' shape in three dimensions through different acquisition systems such as 3D scanners or lasers have appeared. A few examples are mentioned below, since it is out of the subject of this investigation. However, it is worth mentioning that they are surely the key to improving the aggregate characterisation in a very precise way, which may

help to deeply understand how they influence the granular material performance at macro meso and microscale levels.

For instance in, [61] F. Lanaro et al developed a new 3D shape and roughness characterization method for aggregates, using a 3D-laser scanning. The evaluation of their morphological characteristics was done by using both: analytical Fourier analysis and geometrical analysis. The results were compared with the manual measurement data, ensuring the reliability of the proposed model.

Nevertheless, the precision of the digitized version of the original particles' volume and surface area needs to be addressed, as the surface area is overestimated due to the uncertainty arising from the partial volume effect of the surface voxels [62]. This implies a problem for the 3D characterization of the particles and implementation in stone-based material simulation, as the aggregate size and volume fraction influence the mixtures.

For that reason, most of the work related to the aggregates scan-based modeling, the stones are visualized by reconstructing their surface through different techniques as can be seen in [63] and [62]; where the authors use the Marching Cube algorithm on the voxel assembly to obtain a triangular mesh surface of the particles, to obtaina more accurate representation to measure their 3D shape´s parameters. A similar workthat aims to quantify aggregate angularity and surface texture in three dimensions, based on a Voxel-Based Approach, can be found in [64].

Recently, the University of Texas-Austin developed a Laser-based Aggregate Scanning System (LASS) to extract the stones' volumetric information [65]. In this technique, the particles are placed on a platform while a moving scanner is passing over the sample, then a triangulation technique is used to measure their volume. As the

bottom of the particles cannot be scanned, LASS uses an algorithm to estimate the final stone´s volume, based on the shape information of the upper part scanned. Nevertheless, the shapes factors such as angularity are measured in two dimensions, assuming that the aggregates are placed in the most stable position. Additionally, the particles are also quantified through group texture by using Fourier transform.

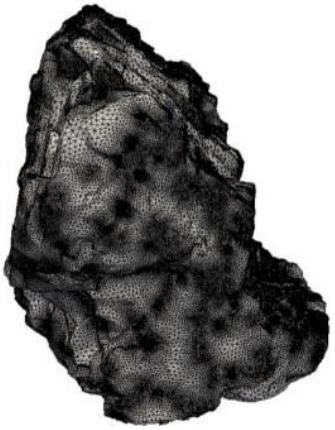


Figure 2-8 –Large-scale rock reconstruction
Note. Image from: Surface mesh generation of large-scale digital rock images in 3D [66].

It is also worth mentioning an interesting work to create large-scale digital rock images in 3D [66], in which the authors propose a combined use of the Marching Cubes algorithm [67] and Centroid Voronoi Tessellation (CVT) [68], in order to generate a surface mesh of large-scale rocks with great precision, for, in a future, accurately describe the rock structure (Figure 2-8) .

First, the Computed Tomography (CT) is used to extract the material information; then, using the Multi-material Marching Cubes algorithm a sub-ground mesh is created; finally, a simplified CVT is employed to generate the nodes that will be triangulated with a high degree of precision.

Other works related to the 3D morphologic features quantification of the aggregates can be seen in [69][62][70]. However, particle reconstruction is still expensive in economic terms. It only allows scanning one or just a few particles at a time, so these systems are not very efficient when evaluating large aggregates samples, although they have high precision.

2.2.5 Conclusion

The numerous existing methods for the characterization of aggregates have been described in this section, starting from the conventional laboratory methods, which are subjective, laborious and time-consuming; to the Image-based Analysis Methods,which are faster and more accurate than the manual one. Nevertheless those latter consist on expensive devices not available to everyone.

Further, new methods are appearing to characterize the aggregates in three-dimensions by using different acquisition techniques such as 3D-scanners, however these approaches are still at an early stage.

In this research, due to lack of access to the systems that have been discussed in the previous sections, a free license image processing software, ImageJ, was used to analyse the morphological properties of the aggregate samples in two dimensions at the mesoscale level, as described in the next section. That means that the surface texture of the particles will not be taken into account, since it is considered a shape factor that measures the micro-irregularities of the stones [50].

2.2.6 Particle´s shapes indices used in this research

In this work, an image-based analysis technique has been carried out to evaluate different aggregates´ samples to obtain their index shape´s factors with higher precision,

objectivity, and faster than the manual methods are tedious and time-consuming, as mentioned previously. The free license software *ImageJ* [71], has been used to extract the particles' morphological information in two dimensions, which are described below.

Many parameters and formulas measure the shape´s parameters of the particles since they have been studied, defined, and modified by numerous authors. A complete review can be seen in [72][73]. However, we will focus on the most common properties when it comes to quantifying aggregates morphological characteristics [73], and those which can be measured by the Image Analysis software: *ImageJ* and the Dynamic Digital Image Analysis, *Camsizer*, thanks to Tarmac company collaboration.

These two tools were, in principle, the ones available for the creation and validation of the Virtual Aggregates Generation model with realistic shapes presented in *Chapter. 4.*

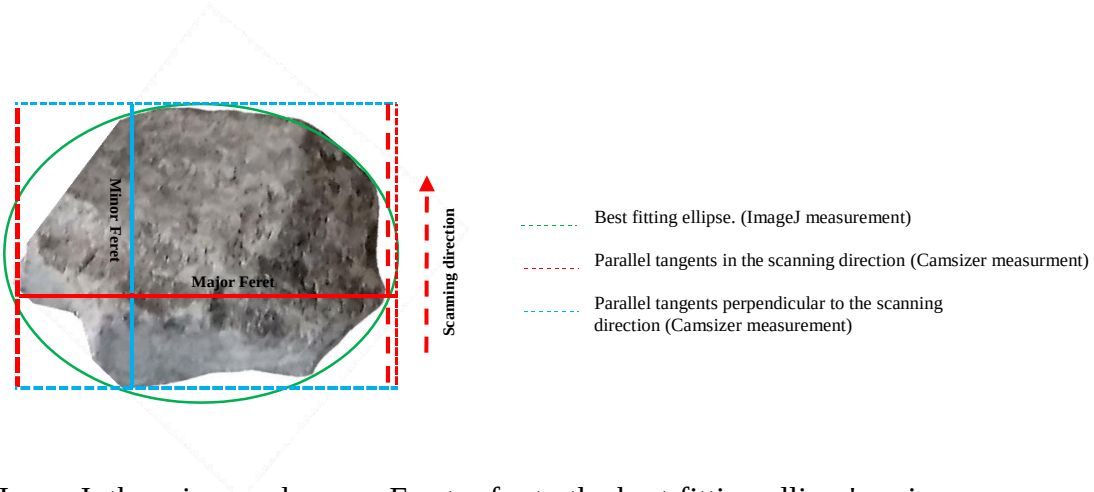
The following parameters have been used for the particle analysis.

Perimeter and Area of a particle projection. They refer to the boundary measurements of the particle projected in its more stable position and the area within the perimeter perspective. ImageJ analyses are based on densitometry [74]. The Camsizer device measures several areas and perimeters, since the aggregates begin to move through the channel that transports them, and the results vary according to the orientation of the particles at the moment of the capture of the images. To find the projected area and perimeter when the particle is in its most stable position, the Camsizer is guided by the major axis, as explained in the following paragraphs.

Length and width of particle projection. Several factors exist which define the length and width of particle projection [72]. In this work we will focus on the Minor and Mayor Feret diameter (calliper measurement) [75], for two main reasons: 1) the

Image Analysis software ImageJ and Camsizer, both used in this project, have the option to calculate Minor and Mayor Feret´s diameters and 2) the Minor Feret´s diameter gives the closest size to a sieve analysis [76][77][78].

Figure 2-9 –Measurements of a stone in its more stable position by the Image Analysis software's: Camsizer and ImageJ.



In ImageJ, the minor and mayor Feret refer to the best fitting ellipse's axis on a particle placed in their more stable position (green ellipse in Figure 2-9). Whereas Camsizer, as the stones are moving, several measurements are recorded to find the Minor and mayor Feret´s diameters, since they depend on the particle orientation and the scanning direction.

The Feret´s diameters are the largest and shortest distance between any two parallel tangents of a randomly oriented particle projection. The largest of a set of Feret´s diameter measured (which should coincide with that measurement made on the particle´s projection in its most stable position) is considered the length of particle or Mayor Feret (red lines in Figure 2-9).

The Minor Feret´s diameter is, the smallest measure distance calculated of all the Feret`s diameter recorded, measured between two parallel tangents perpendicular to scanning direction [79] (blue lines in Figure 2-9). Figure 2-9 shows the measurements

that the Camsizer would perform during the scanning of a stone in its most stable position.

As will be described later in Chapter 4, the Virtual Aggregates Generation model presented employed a proper algorithm, to detect the different views of the particle to then calculate the: area, perimeter, Minor and Mayor Feret´s diameter in its more stable position, to mimic the measurement of this Image Analysis software.

Aspect Ratio *(AR)*. Both, ImageJ and the Camsizer have multiples aspect ratios definitions, but the most used for aggregate´s characterization is the one that calculates the relationship between the maximum and minimum Feret´s diameters, perpendicular to each other (Figure 2-9). This later is also called Krumbein sphericity by the Camsizer manual [59]. In the present work, the Aspect Ratio is considered the ratio between the maximum Feret diameter and minimum Feret diameter (Equation 2-1). Its values range between 0 and 1, 1 is the value for circles or equilateral polygons, which decreases as it moves away from those regular forms.

$$AR = \frac{Minor_{\ Feret}}{Mayor_{Feret}} \qquad 2\text{-}1$$

Roundness *(R)*. It refers to the roundness of the corners of the projected particle´s area. The Roundness also has several definitions, for instance, the authors Tickell, Wadell, Krumbein, Riley, Pye and Pye amount others, have proposed several methods to measure this property, as it can be seen in [80][81].

The roundness measured by ImageJ refers to Pentland´s circularity [82]. It is an index between 0 and 1 obtained from the relationship between the particle area projected and the major longitudinal axis (Equation 2-2). This parameter gives information

about the shape of the edges of the projected particles.

The Camsizer device also calculates the Pentland´s circularity, although it can measure other specific shape particle factors, such as the Krumbein-Sloss Sphericity. Besides, thanks to its cooperation with the University of Leuven and SCR Sibelco NV, Retsch Technology implemented in the Camsizer an algorithm able to identify the Roundness and the Sphericity (or Circularity) of the particles based on the values of the Krumbein and Sloss chart [60].

$$R = \frac{4A}{\pi\,(Mayor_{Feret})^2} \qquad 2\text{-}2$$

Where,

A is the area of the particle´s projection.

Circularity or Sphericity (C). There are various definitions and indexes to quantify the particles' circularity (and sphericity) in two and three dimensions, since several authors have developed numerous methods, formulas, and definitions [73]. The most famous is the cox´s circularity (Equation 2-3), which quantify the difference between a circle and another geometric shape, [83][84]. The value oscillates from 0 (for filiform particles) to 1 (for a perfect circle)[85].

$$C = \frac{4\pi A}{(P)^2} \qquad 2\text{-}3$$

The Krumbein and Sloss chart is used by the rimegeologist to estimate, by visualevaluation, the angularity of the granular materials (Figure 2-10), and it is still the basis

of today's ISO 13503- 2 or API RP 56/58/60 standards [60]. The chart consists of the ordering of stones according to certain angularity indexes proposed by Wadell and Krumbein [60].

In this Thesis, as shown later in Chapter 3, we propose angularity indices based on 1) the morphological measurements of the aggregates made with the ImageJ and 2) the visual inspection using the Krumbein and Sloss chart.

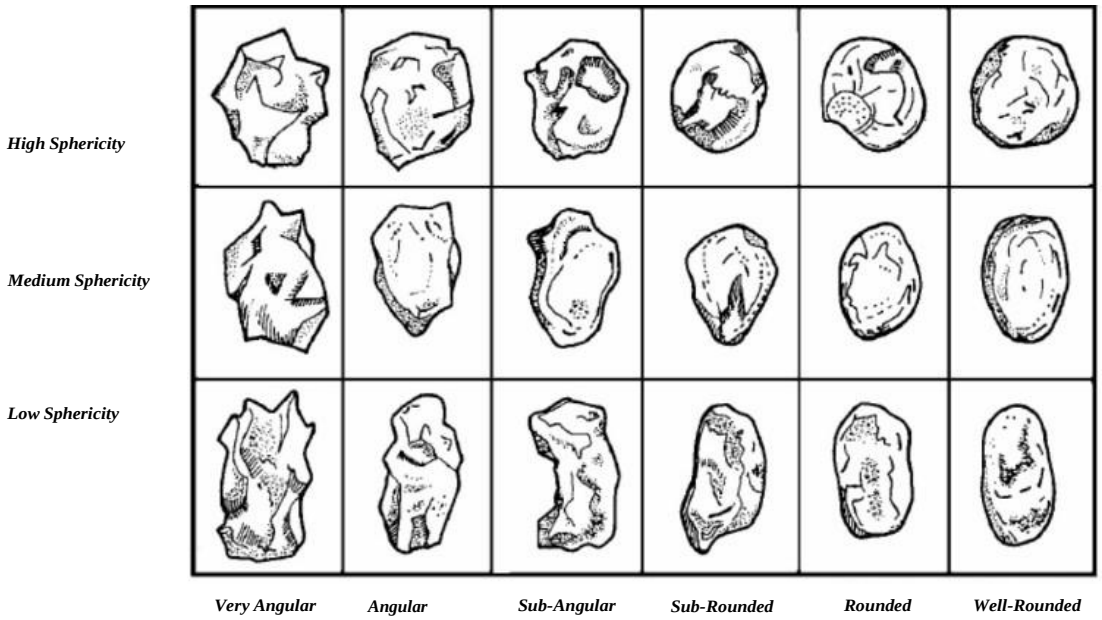


Figure 2-10 –The Krumbein and Sloss chart classification of coarse sediment particles

Note. Image from: Quantifying of particle shape differences of differently milled barite using a novel technique: Dynamic image analysis [86].

2.3 Background of study of granular materials

Granular material can be defined as an assembly of objects, whose interaction between them varies depending on their physical, morphological and chemical properties, among others, resulting in different skeletal structures and pore networks.

Several parameters to quantify the granular material exist, providing a better understanding and a better prediction of their mechanical properties and pore structures.

The present work focuses on how the different aggregate´s size, shape and combination of stones, provide different assembly structures and therefore, a variation in their pore-network and skeleton characteristics.

The investigation of this topic has been the object of numerous studies using both experimental and simulation tests since the aggregates blend design is closely related to the stone-based mixtures' performance [87]. Several research pieces have indicated the dependency of particles properties on the macroscopic parameters of a granular medium. A brief review on this topic is presented in this section differentiating between experimental and simulation works.

2.3.1 Experimental works

Various authors have focused their investigation on characterizing the macropore characteristics of soils and stone-based materials to study their flow transport the skeleton structures they form, directly related to the material performance. Some examples are described below.

For instance, in [88], several soil cores were analysed using X-ray Computed Tomography to characterise their macropore network. These researchers studied several parameters such as: the macroporosity, macropore node density, the average tortuosity and connectivity amount others. Some of the suggestions were that specimens with more macroporosity have a more interconnected pore network. Thus, the Euler number is not an efficient factor in characterising the macroporosity. Moreover, they concluded that the tortuosity does not have a strong correlation with the macroporosity, as its values varied in a close range for all samples analysed.

In [89], the topological properties of the void content are also studied for asphalt samples. Factors such as: porosity, Euler number, tortuosity or the shape of the pores, amount other, were analysed using the X-ray Computed Tomography, to study their relation with the hydraulic conductivities, and also proposes a model for its calculation. The authors found that almost all the pores above 24% of Air Void content are connected, remaining the Euler number a constant. They also found that the tortuosity is higher for samples with less air void content. Another conclusion was that both: the Euler number and the tortuosity values reached an almost constant value for samples greater than 26% of porosity. Regarding the air void shape properties, it was found that the void´s aspect ratio of the specimens with more than 10% of Air Void content was more elongated in the horizontal plane having similar values and that an increase in the porosity means a decrease in void´s circularity.

Another experimental work can be found in [87]. In this case, the study consists of aggregate packing characterization, taking into account the stones' shape and size. Two types of particles were selected and scanned to quantify their morphology. Then they performed packing tests with one-size particles and two-size particles. The results obtained from the one-size stone packing showed that the bigger the angularity, the bigger the porosity. This study demonstrated that a simple sphere assembly could not be directly used to describe aggregates' packing characteristics due to the stone morphology and the single-size aggregate do not have the same size.

2.3.2 Computer simulation works

Due to the need for a better understanding of how the particles' different characteristics affect the structure they form, to predict the stone-based materials performance, robust prediction and simulation computer models have been developed.

The aggregates packing simulations through these computer´s models are the objective of many pieces of research to: 1) test, by comparing with experimental data, how realistic the simulations are and 2) analyse the physical or mechanical properties of the stone-based materials in a more streamlined way than the experimental works.

Nonetheless, the most widely used discrete particle simulation programs, used for the simulation of granular material, are normally designed to represent the particlesas a sphere, as they require less computational effort. Thanks to advances in computer technology, these Discrete Elements Methods programs, DEM, have incorporated non-spherical particles, such as polyhedral. However, their use in terms of particle´s number and computational cost is still limited due to the contact algorithms they use for the particle interaction, as is explained in Section 2.4.

This has led to the search for an approach different to classic DEMs, which is the impulse-based Discrete Element Method (iDEM); which can be implemented in toolboxes, such as Physics Engines (PE) originally designed for the video game and animation industry. The contact solver they use for the particle interaction allows more computationally efficient simulations.

The following describes some works related to the simulation of stone-based materials divided into two groups: simulations made with spheres and simple shapes particles and simulations made with complex shape particles.

2.3.2.1 Simulation with a highly-idealized particle model

In [90], the authors investigate how the shapes of the particles affect the flow through granular materials by using the Representative Elementary Volume (REV) model [91]. They expand theory of Bear and Bachmat (1990) pioneer work to calculate

the tortuosity and permeability of mono-size spheres [92]. A general function is created for randomly packed homogenous granular materials, useful for various shapes such as: sphere, cube disc or cylinder. Here, the 3D particle shape descriptors: compactnessand Aspect Ratio [62], were used to analyse the porosity and the tortuosity that the different particle´s shapes give.

They found a convergence for the tortuosity for the different particle shapes when the porosity increased. Nevertheless, the difference between the tortuosity becomes smaller when the porosity reaches a certain value. Regarding the permeability, this paper concludes that the effect of grain´s morphology is less tangible for most of the porosity ranges, being significant only for those samples with low porosity. Comparing the results with experimental data indicates the capability of their theory on the prediction of the particle´s shape influence on their assemblies.

Previous work similar to the one described can be found in [93], where the authors formulated a porosity-shape and packing size-shape relations, to predict the porosity and to characterize the particles. The authors found different porosity-shape relationsfor loose and dense random packing conditions, as the accepted premise of lower sphericity increase the porosity is not true in the second case. The main conclusion was:the more spherical the particle, the better the particle rearrangement in random packingconditions to achieve the maximum packing density. They also conclude that more measurements are necessary to ensure the formulated relationship, especially when dealing with complex particles.

In [94], the tortuosity factor is studied using a DEM simulation, which uses spherical particles and a viscoelastic contact model (Figure 2-11). The main objective was to evaluate the particle size distribution's influence on porous beds' spatial

structures. The results show that the geometrical tortuosity increases slightly and nonlinearly with an increase in the standard deviation of the particle size´s distribution.

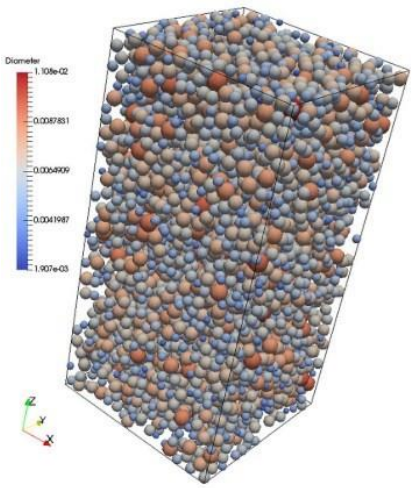


Figure 2-11 –An examples of virtual beds with spherical particles
Note. Image from: The influence of particle size distribution on parameters characterizing the spatial structure of porous beds [94].

A study about the influence of the grain´s shape in the granular material by comparing experimental and numerical data can be seen in [95]. A multi-scale analysis to see how the particle shape influences the granular materials' mechanical response is presented in this work, where a DEM was used to track the interaction between particles.

Although the shape factor has been taken into account, most of the simulation's tests have been done by using spherical particles because the contact detection algorithm for non-spherical particles needs a high processing time. This problem has been addressed by applying a rotational constraint on the particle´s contacts, which improves the sphere interlocking.

To see this approach's functionality, the results obtained for a laboratory rolling resistance test were compared with the results obtained by using multisphere and super

quadrics particles [96]. The conclusion was that this method is only capable of approximating certain characteristics of complex particle shape such as: angle of repose, dilative behaviour and shear strength, but only if the Elastic-Plastic Spring-Dashpot model is used [97], which is very time-consuming.

On the other hand, the superquadrics and multisphere particle´s approach feasibility was also tested through the angle of repose (Figure 2-12), Jenike shear and silo discharge. The comparison with the experimental data indicates that it is yet challenging to predict the flow profiles of the discharging granular material with complex shapes.

Some researchers have focused their studies on the analysis and simulation of aggregate mixtures skeleton, rather than the pore-network. For instance, in [98], a virtual experiment was conducted to see how the different particle sizes influence the mechanical response of Coarse Aggregate Mixtures structures. Continues and gap graded aggregate mixtures were modelled using the Particle Flow Code in three dimensions (PFC3D).

Although the particles were simplified to spheres with different sizes, to reflect the particle shape, which influences the contact surface between aggregates, different friction´s coefficients were employed. Furthermore, a penetration test was also carried out to analyse their penetrating resistance, average coordination number and void ratio of the mixtures.

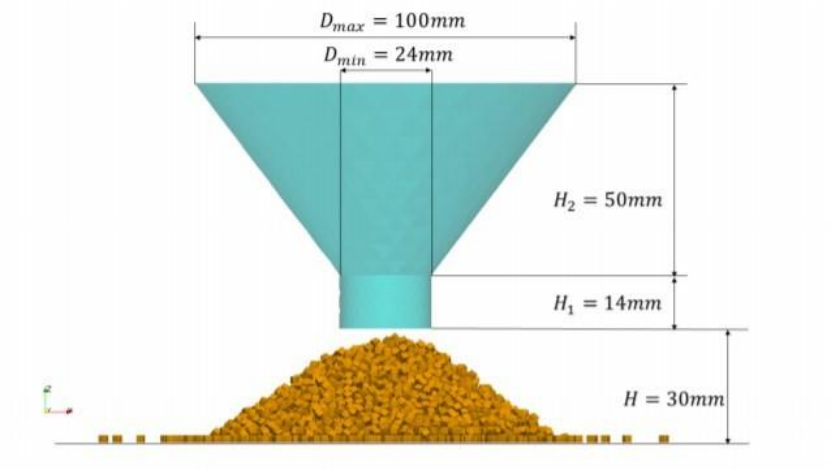


Figure 2-12 –Angle of repose simulation
Note. Image from: The Multi-scale analysis of the influence of particle shape on the mechanical response of granular materials [95].

The results obtained for the Skeleton Structure of Continuous Graded Aggregates show that the penetration resistance increases when the Void in Coarse Aggregate (VCA) decreases and proves that the larger aggregates are the main part of a skeleton structure. A similar conclusion was obtained for the gap graded aggregate mixtures, from which it was also found that the penetration resistance of structures with three particle sizes was larger than that with two particle sizes.

Another finding was that the smaller the difference between particle size, the more variation concerning to resistance to penetration. Regarding the coordination number, it was noticed that it increases with decreasing the VCA, and that its variation was greater for gap graded mixtures due to a greater particle's rearrangement.

A work similar to the previous one can be seen in [99], where the authors evaluate the particle sizes distribution effect on a particle packing by using the PFC3D DEM simulation. They model various Hot Mix asphalt skeletons and correlate their gradation with the volumetric properties of the sample. Further, the structures' contact force chain

was also calculated to evaluate how they transmit the stress through the stone-stone contact. Finally, they conclude that the particle size distribution is directly related to the volumetric and the aggregate packing contact characteristic.

2.3.2.2 Simulation with complex shapes particles

The purpose of the study found in [100], was to predict the mechanical behaviour of various coarse aggregate asphalt skeletons at the micro-contact level, through modelling a virtual penetration test. In this work, Ding, Xunhao et al, captured the morphological parameters of the real aggregates using the Aggregate Image Measurement System (AIMS), already described in Section 2.2.2, to model particles with realistic shapes by using the DEM software Particle Flow Code in two dimensions (PFC2D). Due to the large number of stones needed to represent an asphaltskeleton, the algorithm was optimized by selecting only the most representative shapesand reducing the processing time of the simulation (Figure 2-13).

The results suggest that the model proposed could predict the aggregate mechanical property effectively. The penetration tests were carried out for an Asphalt Concrete skeleton (AC) and a Stone Mastic Asphalt skeleton (SMA), achieving similar values between the laboratory and the simulation. They also found that the finer skeleton (AC) is more unstable than a thicker one (SMA). The number of contact points was also analysed and compared between the AC and the SMA structures, suggesting that the thicker the skeleton, the lower the contact points.

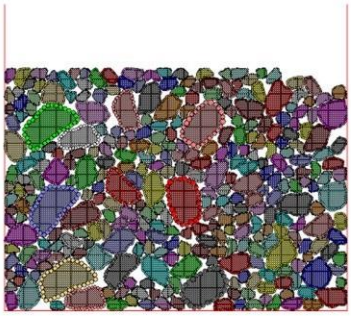


Figure 2-13 –Virtual specimen of coarse aggregate skeleton in 2D
Note. Image from: Morphological characterization and mechanical analysis for coarse aggregate skeleton of asphalt mixture based on discrete-element modelling [100].

Yu Liu et al published in 2017 a new model for the discrete element simulation of realistic particle shapes in stone-based mixtures [101]. Real stone´s samples were scanned individually using X-ray Computer Tomography. Then they were exported to the discrete element program Particle Flow PFC5.0 to create the geometries of aggregate through the clump templates techniques (Figure 2-14.a). They also created a code to generate a virtual sample of stone-based mixtures in loose and dense conditions (Figure 2-14.b).

The model validation showed great precision on simulating aggregate morphology, the main disadvantages found when simulating an aggregate mixture specimen were: 1) it does not take into account the real gradation than a sample of aggregate has, although to tackle this problem, they carried out a parallel work on a sieving analysis simulation with realistic shape [102]; and 2) the computational time, since a single packing simulation requires several hours.

a) Clump templates aggregate´s library

b) Dense compacted stone mixture sample

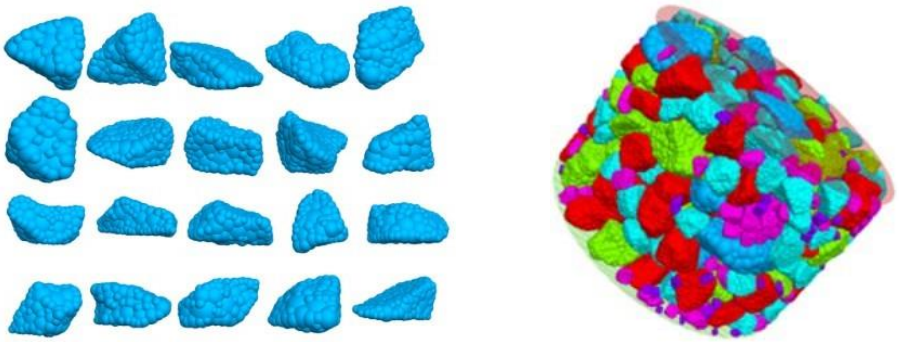


Figure 2-14 – a) Clump templates aggregate´s library. b) Dense compacted stone mixture sample

Note. Image from: Discrete element modeling of realistic particle shapes in stone-based mixtures through MATLAB-based imaging process [101].

In [103], Voronoi´s tessellation was used to represent the grain ballast by modelling convex polyhedral shapes to study the aggregate's aspect ratio influence on an Oedometer test simulation (Figure 2-15). The simulation revealed that flat grains ballast has a greater deformation than cubical ones. This latter also presents the smallest void ratio as the displacement of steel loading was larger.

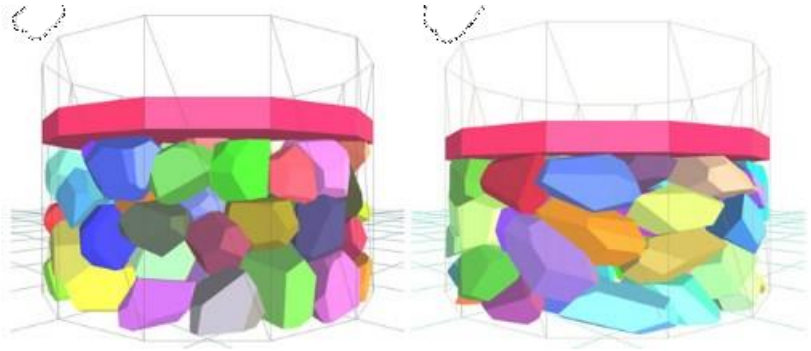


Figure 2-15 – Oedometer test simulation using convex polyhedron with different aspect ratio

Note. Image from: Image from: Simulation of railway ballast using crushable polyhedral particles [103].

Previous work to the one mentioned above, which also focuses its study on the railroad ballast simulation, can be seen in [1], where various types of grains were modelled using an imaging-based aggregate morphology characterization DEM program BLOKS3D [3]. The main purpose was to investigate the effects of aggregate´s shape properties: angularity, and surface texture on: the aggregate assembly strength, the ballast aggregate settlement, and the lateral stability test to improve railroad ballast design.

For this purpose, they create different element shape libraries (Figure 2-16), by previously extracting the morphological information from the real aggregate through the "University of Illinois Aggregate Image Analyzer" (UIAIA), already described in Section 2.2.2. Then, the ballast DEM model was validated by matching the experimental results of a direct shear box strength test with the experimental one, through the calibration of the DEM parameters.

Regarding how the gradation affects ballast performance, the simulation analysis revealed that more uniformly gradation yields higher air-void content. They also found that when the minimum particle size decreases, the air void content decreases due to finer particles, until further decrease on the minimum particle size leads to an increase in the air void content due to an expansion phenomenon when finer particle separates the larger one

On the other hand, regarding the simulations carried out to study the aggregate shape effect in the ballast performance, the authors conclude that ballast with angular aggregate´s samples and elongated aggregate´s samples have higher settlement resistance than the rounded one. Also, particles with a lower angularity index have a weaker particle interlock and lower lateral resistance. Besides, the authors conclude

that particles with smooth surface textures have a poor shear strength from the lateral stability test simulation, as expected.

Similar work was developed in [104], where the polyhedral library of BLOKS3D are used to model granular material. However, this research focused on improving computational performance within the conventional DEM framework and implementing an impulse-based Discrete Element Method (iDEM) in block3D, briefly described in Section 2.4.

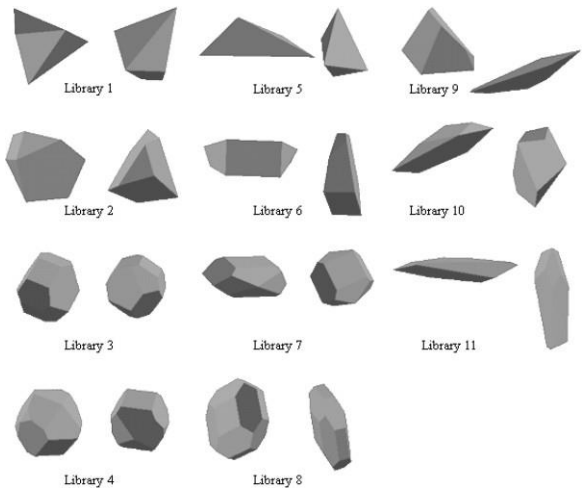


Figure 2-16 –Two views of different representative aggregate shape´s library in BLOKS3D

Note. Image from: Image from: Discrete element modeling of railroad ballast using imaging based aggregate morphology characterization [1].

As an alternative to the traditional DEM method, researchers have recently started using Physics Engines for simulating granular material due to their computational efficiency when entering complex shapes. However, their accuracy is disputed. Several authors focused their work on testing the physics engines' ability to simulate granular material, as shown in the following paragraphs.

For instance in [6] Peng Cao et al (2016), model convex polyhedron as coarse aggregates using the Quickhull algorithm [7], and particles are imported to the physic engine swig-Bullet multi-body, based on the Monte Carlo method [8]. The goal was to mimic the aggregates movements in asphalt concrete during a compaction process, as the physics engine recorded translation vectors and rotated vectors of every aggregate particle at each time interval.

They considered an aggregate gradation's statistical characteristics; and simulated a random aggregate arrangement of asphalt concrete. Then, the 3D random aggregate model is imported into a Finite Element program to simulate the asphalt dynamic modulus (Figure 2-17). The author demonstrated the 3D random aggregate pre-processing method's efficiency to simulate the random arrangement and shape of the stones.

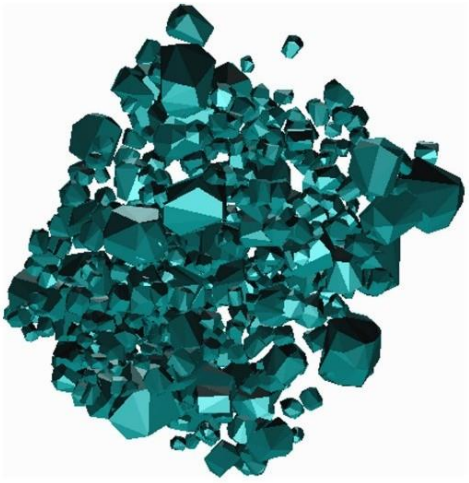


Figure 2-17 –3D random aggregate geometric model with gradation considered
Note. Image from: Prediction on dynamic modulus of asphalt concrete with random aggregate modeling methods and virtual physics engine [6].

In [105], compaction and packing of virtual aggregates are simulated using the PhysX engine [106], to study the pore space of the coarse stone assembly of different asphalt mixtures designs.

The authors generated convex aggregates with a given aspect ratio by using the Convex Hull algorithm. Then the stones were sized according to a gradation chosen by the user. Finally, the virtual aggregates were imported into the PE and placed into a container, where they are vibrated and compacted. The validation was done by comparing the virtual porous sample's morphological analysis with the experimental one, with good results.

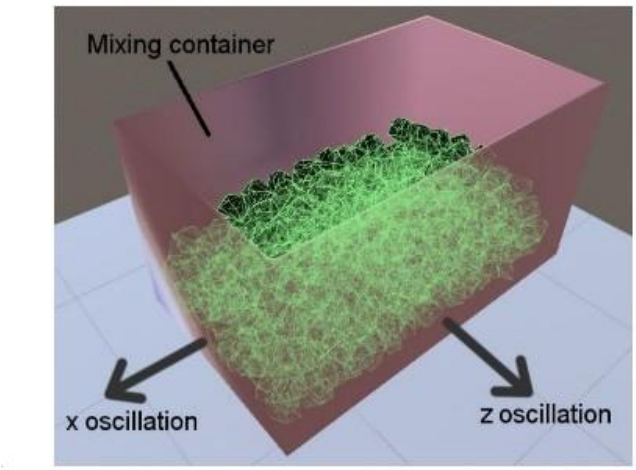


Figure 2-18 –Vibration of 3D random aggregate´s sample with PhysX engine

Note. Image from: Computational Packing of Aggregates for the Study Of Virtual Asphalt Samples [105].

E. Izadi et al (2013) [9] simulated a granular soil behaviour using the Bullet physics engine. The gravel was represented through randomly shaped polyhedrons created with the Voronoi´s tessellation [10], process that will be explained in more detail in Section 2.5. Pluviation and a vibration test were simulated and compared with the experimental data with good agreement. Another investigation from the same

authors, E. Izadi et al (2018), wanted to demonstrate the feasibility of bullet engine by validating, with experimental test, the results of the simulations of direct shear testing on uniform-sized metal beads [12].

Fan Zhu et al. (2018) [11], simulate a crushable granular media under load by combining: 1) a peridynamics analyses to study the breakage of the particles [107] and 2) the use of the Bullet physic engine for the breakage grain´s simulation and their interaction) [11]. Comparing the results with experimental observation demonstrates the hybrid computational model's predictive capability concerning the particle morphology and particle size distribution, the 1D compression on crushable sand and fractal dimension. Nevertheless, the author suggests incorporating particles with more complex shapes and more advanced techniques for solving constraint systems in the physics engines to allow multi-phase modelling. He also suggested a GPU level parallel computing approach to improve performance and computational efficiency [108][109].

In [13], realistic particles and a soil specimen collapse process were simulated with the Bullet Physics Engine, where the contact force chain was calculated. Five hundred thirty-two irregular uniform-sized stones with about 30 mm were scanned and imported into the Bullet engine, where the stones were placed under gravity in a cylindrical container (Figure 2-19). The collapse process simulation starts when removing the cylindrical container. Further, the author performed the same simulation with DEM, which took 33 hours versus 20 seconds using the physics engine, makingthe latter a very attractive tool.



Figure 2-19 –Top and front view of the sample generated in Bullet

Note. Image from: Simulation of Realistic Particles with Bullet Physics Engine [13].

The most recent work is found in [110], where grain packings are simulated to study how their porosity coordination number distribution changes with the angularity of frictionless particles. Particles with spherical and convex irregular shapes were modelled, by adding a controlled Perlin noise to a sphere [111], with prescribed size distributions based on the volume on a sphere of equal diameter. The assemblies were performed with the PhysX PE through the Unity 3D platform, where the particles were placed under gravitational and vibrational forces to generate loose and random dense packings.

They concluded that: denser grain´s packings have higher coordination values, and porosity decreases as the particles' angularity increases. These results were corroborated with previous experimental studies.

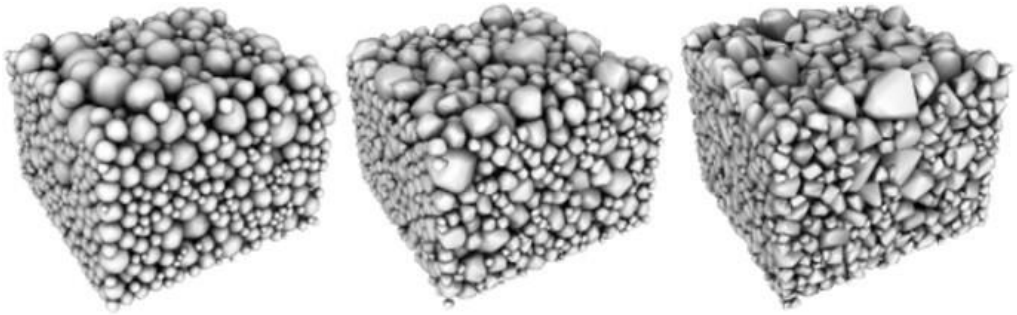


Figure 2-20 –Random dense particle´s assemblies with convex irregular shapes with different level of Perlin noise

Note: Image from. Particula: A simulator tool for computational rock physics of granular media [110].

2.3.3 Indices to quantify the granular material used in this study

This section summarises the most common parameters used to quantify the granularmaterial structures and those used in this thesis. These parameters are used to: 1) analyses the different internal structures formed from the real aggregate packing madein the laboratory and 2) validate the virtual aggregate packing model presented in this work, by comparing the experimental and the virtual results.

Porosity is an essential characteristic for better understanding and predicting the granular material performance and it is strong related to particle surface geometry [93]. There are a large number of methods to quantify porosity, and different ideas about what it means.

In this research, porosity is the volumetric macroporosity, which is the total volume of all macropores quantified in the in the volume of the sample studied, V_T (Equation 2-4). This factor plays a central role in the physical processes such as fluid flow and can be used, for example, for the optimization of the blend aggregates design.

$$p = \frac{V_v}{V_T} \qquad 2\text{-}4$$

Where,

V_v is the volume of voids and V_T is the total volume of the sample, which in turn is the sum of the volume of solid $V_s$and the volume is of voids V_v.

Tortuosity is used to characterize the pore structures of granular materials, as pore channels are not straight. A complete review can be found in [112].

In this work we refer tortuosity as the geometrical tortuosity τ_{geo}, (Equation 2-5), which is the ratio of the geometric length of the path, L_g, to the Euclidean distance, L_E, which is the straight-line length between the extreme points of the pore channel. If the ratio between both is 1, it means the absence of tortuosity, whereas the greater the number, the greater the geometrical tortuosity.

$$r_{geo} = \frac{L_g}{L_E} \qquad 2\text{-}5$$

Where,

L_g is the geometric distance and L_E, the Euclidean distance

Connectivity. The connectivity of the pore network is normally expressed with the volumetric Euler-Pointcaré characteristic (Euler number,v). The Euler number can be defined, as appear in Equation 2-6, as the number of non-interconnected clusters or isolated pores (I) minus the number or redundant connection or genus (C), which means the loops or the number of voids that can be removed without creating additional isolate pores, plus the number of completely enclosed cavities (H) in the volume sample

studied [113][114]. If there are no isolated pores, then, all the pores are interconnected from the surface to the bottom of the sample and the Euler number will be negative. The larger the Euler negative number, the more connectivity will present the analyzed sample.

$$v = I - C + H \qquad 2\text{-}6$$

Nevertheless, the study of tortuosity through the Image Analysis software ImageJ is still inaccurate, since measurement may give very different values for the same specimens. This could be due to image resolution, noise, the threshold and the segmentation algorithm chosen, or even the volume of the sample analysed. For these reasons an alternative to the parameter, Euler number, is used in this research, which will be named from now: *the average pore-connectivity,,* which is also useful to quantify the interconnectivity. These parameters are described below these lines.

The average pore-connectivity, z, is a function of the node density, *n*, which indicates when the pore network branches are connected, also named junction (Equation 2-7). It is a relevant factor to describe a large pore-scale connectivity instead of the Euler number [115]. Its value is calculated as the average number of branches of the medial axis meeting at a junction in the structure analysed [116].

$$z = \frac{3n_t}{n_j} x \frac{4n_q}{n_j} x \frac{5n_x}{2n_j} \qquad 2\text{-}7$$

Where,

$$\frac{n_x}{n_j} = 1 - \frac{n_t}{n_j} - \frac{n_q}{n_j} \qquad 2\text{-}8$$

Where,

n_j is the total number of junctions, n_t, n_q and n_x is the number of triple-point junctions, quadruples junctions and higher order of junctions, respectively, meaning the connection of 3 branches, 4 branches or more branches in the pore-structures [117] [118].

Void´s morphology includes shape factors such as circularity, roundness and aspect ratio used to characterize the stone´s shape already described in Section 2.2.6, which are also used to describe the pores' geometry. These three parameters are values that vary within the range 0-1. A value of these factors close to 1 indicates elongated voids, normally related to better connectivity.

Stone-stone contact helps to understand the granular material's stability, since they can evaluate how, an applied force, is transmitted through the structure [1].The most commonly used index for the average stone-stone contact quantification is the coordination number, or the number of touching neighbours per particle.

This parameter has been tried to measure through different techniques and plugins offered by Image Analysis ImageJ software such as the macro: *ND*, which is an ImageJplugin for image analysis of porous materials [119]. However, different values are obtained for the same sample making his evaluation very uncertain.

Therefore, as the main objective of this Thesis is not the in-depth analysis of the structuresof the granular material made in the laboratory, but the validation of its virtual simulation, a different approach is proposed, to represent an approximation of the aggregate´s skeleton structure. The parameter used for this approach is the particle´s node density n_d.

The particle´s node density (n_d) can be defined as the total number of junctions n_j of the aggregate skeleton per unit volume, as shown in the following Equation:

$$n_d = \frac{n_j}{V_T} \qquad 2\text{-}9$$

In this case the junction has the same definition as described for the average pore-connectivity, z, (Equation 2-7), but instead of referring to the number of branches connected on the pore structure, it refers to the number of branches connected on the aggregate´s structure.

2.4 Theory of the Physics Engines (PE). The Nvidia PhysX engine

The Physics Engines, used mainly in the video game and animation industries, are designed to solve real physics problems. They could have various software libraries which can produce different physical simulations such as: rigid bodies, deformable bodies, cloth and fluids. In this section we will focus on the dynamic rigid body system module, as is the one suitable to simulate granular material in dry conditions.

The rigid body system computes bodies' motions based on the Newton-Euler equations for both: translational and rotational motions, as they are designed to simulate the object´s movements in a physically accurate manner [120]. This section summarises how physics engines work in terms of collision detection and collision resolution.

2.4.1 Basics Structure

Generally speaking, we can say that both DEM and a real-time PE have a similar procedure to compute the physics for a dynamic contact which is: the particle modeling as a rigid body, the contact detection and the contact resolution, through solving

Newton´s and Coulomb´s laws, using a fixed time step. They differ in the time step due to the contact resolution models they use, as shown Figure 2-21.

The classics DEMs collision models use a penalty method, where the contact detection is solved before the collision occurs, allowing overlapping between particles. This penetration between particles defines the restoring forces' magnitude, which is modelled as a spring. This implies that the collision solver needs to act in small-time step increments to have numerical stability on the system, producing a long run time [12].

On the other hand, the Physics Engine use an impulse force approach, assumingthat particles are not deformable objects. In this case, the elastic contact response is not considered, and particles are assumed perfectly rigid to enforce the non-penetration via complementarity conditions. For this reason, PE´s can have larger time steps and less computational effort compared to DEM´s simulations [121]. The iterative solver has lower precision in the real-time Physics Engines in numerical terms, although increasing the processing time would increase the accuracy [122]. More detailed information regarding the difference between these two methods can be found in [9][12][121][123][124].

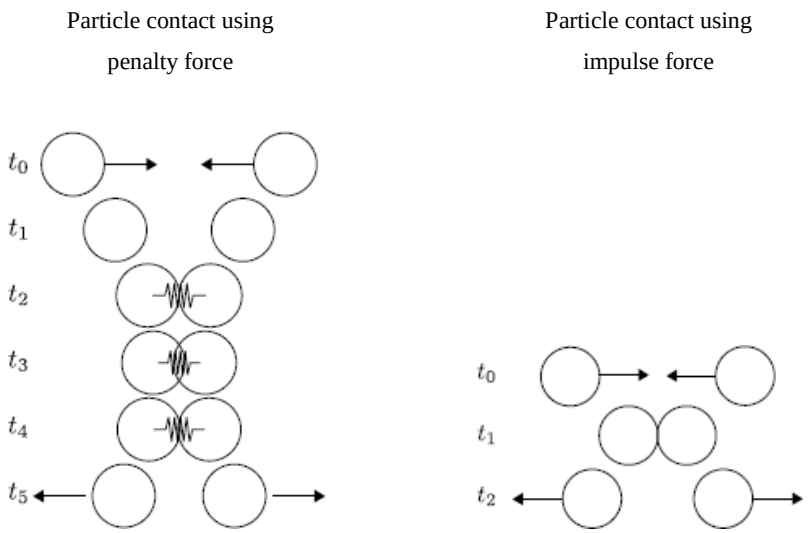


Figure 2-21 –a) DEM´s contact model and b) PE contact model

Note Image from: Smooth and non-smooth approaches to simulation of granular matter [121].

The impulse force approach used on the PE´s, also called the hard contact approach, was first introduced by J.J. Moreau in 1988 [125]. The particles are assumed perfectly rigid, which simplifies the force computation in the meaning that each force acting in a contact point can be represented as a force-torque impulse in the center of the object's mass. For the simulation, the particles' physical properties have to be defined in terms of coefficient of friction, coefficient of restitution and shape. Their motions are simulated once the force acting on the objects due to the inter-particle's collisions. They are computed and integrated to determine the new position ofthe rigid body.

Figure 2-22 shows a typical Physics Engine pipeline is outlined, which can be divided into 3 main parts: 1) the collision detection (which in turn is divided in two phases: the broad phase and the narrow phase); 2) the collision resolution, (which consist in two main steps: the calculation of the object´s velocity and the calculation of

the restoring forces); and 3) a force´s integrator (Update Solver), which reflects the movement produced during the collisions.

It should be noted that the collision impulse-based model is currently being implemented in the classics DEMs, named iDEM (impulse-based Discrete Element Method); to speed-up the simulations with complex shapes particles, with reasonable levels of simulation fidelity. Some works which use the iDEM for the granular material simulation can be found in: [104][126].

The following point briefly described how the collision system works, since they contact are complex topics outside the subject of this thesis. More detailed information about the contact resolution and the contact detection process can be found at [127][128].

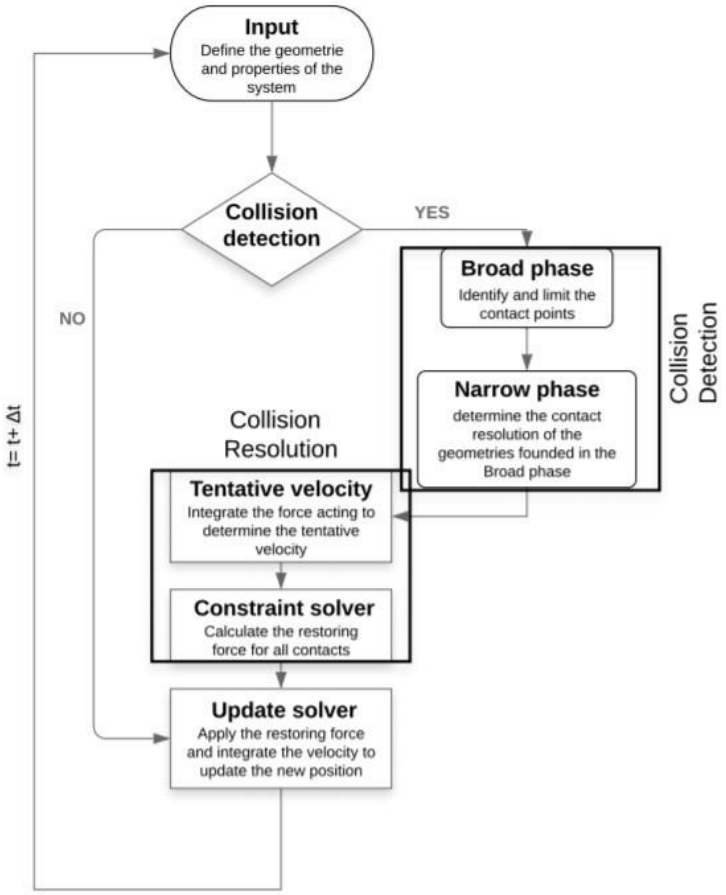


Figure 2-22 –Basic structure of a Physique Engine

2.4.2 Collision detection

Collision detection is the previous step to the collision resolution (or collision response), and it serves to determine the colliding bodies and determine their possible contact points [127]. The system considers two objects colliding when the shape of their rigid bodies are either: touching, intersected or falling under a threshold. Due to the computational cost of determining a large number of particles' contact points, especially if there are complex shapes involved in the simulation, this step is further subdivided into phases: the Broad phase and the Narrow phase (Figure 2-22).

Different methods for both phases of collision detection exist. However, these are complex and fall outside our field of research, a generalized idea will be given for both processes. This topic can be seen in more detail in [128] [129].

The Broad phase serves to discard the pairs of bodies far enough to not be in contact and to find those that may be colliding. The main idea is to approach the complex shape of the rigid bodies to simplest shapes like cuboids or spheres, in order to accelerate the process of finding potential colliding objects.

The simplified volume of objects is called the Bounding box and there are several types such as: Oriented Bounding Box (OBB) or Axis Aligned Bounding Box (AABB) among others [130] [127]. Then, a space portioning algorithm is used to reduce the number of bounding boxes that are overlapping. Finally, it is determined if the bounding boxes are colliding, if their projections intersect along the coordinate axis.

Different algorithms can be used for the Broad phase. We will briefly describe one of the most popular algorithms: the Sweep and Prune Algorithm.

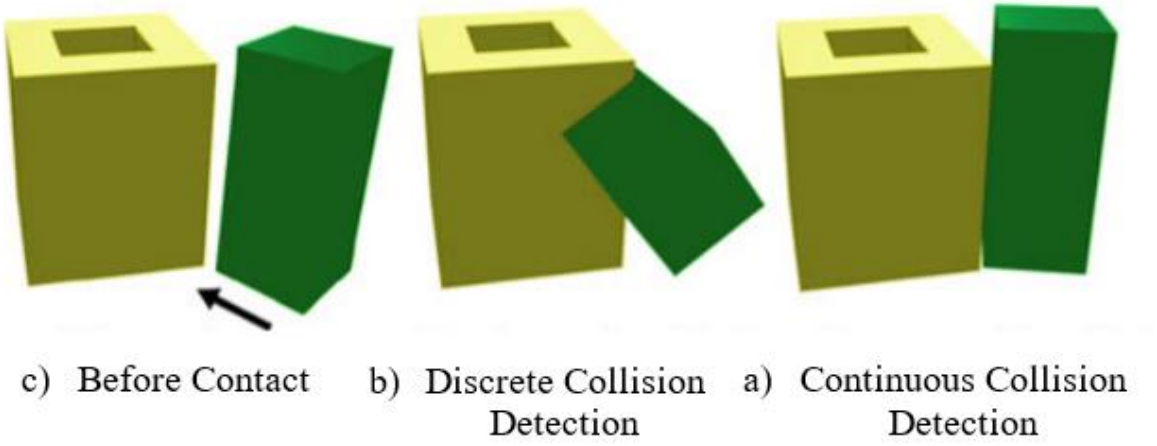
The Sweep and Prune Algorithm, also known as Sort and Sweep, divide the 3D space to find bodies close to each other and discard the remainder for further testing. All the bodies in the system are simplified using an Axis Aligned Bounding Box (AABB) [131]; the Sweep and Prune algorithm turn the AABB projection into a list of intervals (which are single coordinate axis) for each dimension. Then it sorts the list of intervals in ascending order by their scalar values. Finally, it “sweeps” the list finding possible intersected intervals. If the intervals of two bodies intersect in all dimensions, their AABB overlap and therefore, they can be proposed as a pair of potentially colliding bodies. More detailed information can be obtained in [132].

On the other hand, the Narrow Phase could be defined as a Broad phase refinement, which is in charge of finding if the potential bodies are: colliding, intersecting or falling under a specified tolerance threshold. The Narrow phase's final aim is to find the contact´s points between the colliding bodies to be able to later compute contact forces and the resulting motion during the contact resolution phase.

Several methods also exist for the Narrow phase process. For example, the Separating Axes theorems (SAT)[133], which is a disjoint convex sets in n-dimensional Euclidean space, to find their the separating axis; The Gilbert-Johnson-Keerthi Algorithm (JGK) [134], which compute the distance between two bodies and their closet points, which is useful to consider two bodies colliding below a certain limit. These last two methods are within the so-called "a posteriori", static collision detection or discrete collision detection (DCD) [130][135]; since they are intended to search for intersections between static objects and they have to be called at fixed time intervals to deal with the motion of objects. This results in missing collisions and objects may overlap severely. [136][137].

However, there is also the option of Continuous Collision Detections (CCD) (also called a priori or dynamic collision detection) [130], which is one of the most accurate since it can compute and predict the contact points between two objects during the previous step and the current one, avoiding like this some drawbacks of the static collision, such as: the intersection between geometries and the omission of some collisions that can occur during that time.

Figure 2-23 –Difference between DCD and CCD for two same objects colliding



Note. Image from: 6-DoF Haptic Rendering Using Continuous Collision Detection between Points and Signed Distance Fields [137].

The CCD works by calculating the Time of Impact (TOI) between two bodies to compute potential collisions by sweeping their forward trajectory using their current velocity. Then it goes back in time so that it determines the collisions at that point. Nevertheless, the accuracy depends on the complexity of the object´s shapes, the number of objects, and the resolution, making it impossible to completely avoid collision detection failure for complex simulation scenarios [137][138].

In Figure 2-23, a representation of the penetration caused by the Discrete Collision Detection configuration, and the nom penetration of the same objects with a Continuous Collision configuration is plotted.

Nonetheless, all these algorithms have the same drawback, they are designed to work with convex shapes. Besides, although some algorithms able to detect contact between non-convex particles exist, those require a high computational effort [130]. Note that the non-convex particles are referred to those irregular particles which present

concavities.

As one of the goals of this research is to generate aggregates with realistic morphology, including concavities, it is necessary to work with non-convex shapes. For this purpose, the convex decomposition algorithm exists [139], as it will be seen later in Chapter 5.

2.4.3 Collision resolution. Constrained Rigid Body Simulation

Once the contact points are found during the collision detection phase, the particles' movement is simulated by updating the rigid bodies' physical state by changing their position and orientation. Although PEs can also use penalty methods (see Figure 2-21), especially to simulate deformable objects, we focus on the constraint-based method, since they are the most appropriate to ensure a non-penetration between objects [137].

During the collision resolution, we can differentiate two main steps (see Figure 2-22): 1) the *tentative velocity*, which integrates the force acting on an object to calculate its tentative velocity and 2) the *Constraint solver*, which calculates the necessary impulses to avoid the violation of the constraints, which in this case it is the no penetration constraint [135].

On the one hand, the Update Solver is a numerical integrator that simulates the particle´s motion by using the Newtonian mechanic laws along one time step (Figure 2-22). When a collision is detected, the update solver is stopped, and the Constraint solver handles the instantaneous changes of the body velocity that the collision produced in a discontinuous way to avoid interpenetration. Then the Update solver is restarted, integrating the Constraint solver's impulses and updating the objects' positions.

Sometimes the interpenetration between objects cannot be avoided due to numerical error (as the constraints are formulated at the velocity level). This error is reduced by decreasing the size of the time step [5]. Several iterations of the Update solver are performed during on frame rendered to achieve realistic results.

The physics on which the contact resolution algorithm is based are described below.

Characterization of the non-deformable objects

The non-deformable object is characterized by: a translational vector $x(t)$, which indicate the barycenter of the body in the *world space*; and rotation matrix $R_i(t)$ that transforms the orientation of the particle from to the world space coordinate system.

The other constant parameters describing the physio-mechanical properties of the rigid bodies are: the mass; the moment of inertia in the body space (tensor); the Coefficient of Restitution (COR); which serves for the collision dissipation; the surface friction (μ); and its shape, through a triangular mesh.

Equation of Motion (EOM)

Thus, the translational and rotational state of a dynamic body in the world space at time t is described through the state vector $X(t)$ (Eq. 2-10). The Physic Engine compute the mass centre position based on the shape of a body.

$$X_i(t) = \begin{pmatrix} x_i(t) \\ R_i(t) \\ P_i(t) \\ L_i(t) \end{pmatrix} \qquad 2\text{-}10$$

Where,

$x(t)$ is the position of the centre of mass and is a time´s function;

$R(t)$ is the rotation matrix which is expressed in terms of quaternion

$P(t)$ and $L_i(t)$ are the linear and angular momentum respectively.

According to Newton´s second law, the Equation of Motion (EOM) is achieved by deriving the state vector, previously described. Thus the EOM is:

$$\frac{dX_i(t)}{t} = \begin{pmatrix} v_i(t) \\ \frac{1}{2} w_i(t) q_i(t) \\ F_i(t) \\ \tau_i(t) \end{pmatrix} \quad \text{2-11}$$

Where,

$v(t)$ is linear velocity.

$w(t)$ is angular velocity.

$F(t)$ is the total external force acting on the i object and,¡

$\tau_i(t)$is the torque around the centre of mass.

The angular velocity depends on the Inertia tensor $I(t)$ which is computed from the body-fixed moment of inertia tensor I_b and the rotation matrix $R_i(t)$ [141]. To track the change of the state vector over the time, at the iteration *m*, the algorithmic update for the ith body can be expressed as:

$$X_i^{m+1} = X_i{}^m + \Delta t^m \left(\frac{dX_i}{dt}\right)^{m+1} \quad \text{2-12}$$

Where, Δt^m is the time-step.

Impulse and Torque calculation

To know the correct movement of a body at each time, it is necessary to compute the total forces, $F(t)$, acting on the rigid bodies. This is done by using the

next equation:

$$F_i(t) = W_i(t) + F_{i,n}(t) + F_{i,t}(t) \quad \text{2-13}$$

Where,

$W(t)$ is the weight of the particles in the gravity direction;

$F_{i,}(t)$ and $F_{i,t}(t)$ are the normal and tangential forces respectively, acting on the aggregates due to the collision between other aggregates and the container's wall, outlined in Figure 2-24.

$F_{i,}(t)$ is associated with the repulsive forces and energy dissipation based on the restitution coefficient, inputted by the user, to define the system parameters before the simulation.

Tangential forces

The tangential forces, $F_{i,}(t)$, are related to coefficient the of friction. The total tangential force is the sum of the tangential force produced by two bodies in contact, plus the tangential forces produced by the test tube walls, as shown in the next equation.

$$F_{i,t}(t) = F_{iL,t}(t) + F_{iB,t}(t) + \sum_k F_{ik,t}\,(t) \quad \text{2-14}$$

Where $F_{iL,t}(t)$ and $F_{iB,t}(t)$ are the tangential friction forces caused by lateral and bottom walls, respectively, and $\left(F_{ik,t}(t)\right.$ is the tangential friction force due to collisional contact between the ith and kth bodies.

The tangential friction forces due to the collisional contact between the ith and kth bodies are computed at a contact time t_c according to Coulomb's theory as indicated by the following equation:

$$F_{ik,t}(t) = \mu f_{ik,n}(t)\hat{t}_{ik}(t) \qquad 2\text{-}15$$

Where,$F_{ik,t}(t)$ is the tangencial force acting when two bodies i and k are in contact, as ispictured in Figure 2-24; μ is the coefficient of friction; $\hat{t}_{ik}(t)$ is a unit vector of the space coordinate where the friction force act; $f_{ik,n}(t)$ is the magnitude of the normal force ate the contact point.

In turn, the colliding contact between particles where the friction force act, $\hat{t}_{ik}(t)$, is expressed through the velocities of the bodies at the contact point, as follow:

$$\hat{t}_{ik}(t_c) = \frac{(\hat{n}_{ik}\times v_{pik})\times\hat{n}_{ik}}{|(\hat{n}_{ik}\times v_{pik})\times\hat{n}_{ik}|} \quad and \quad v_{pik}(t) = v_{pi}(t) - v_{pk}(t) \qquad 2\text{-}16$$

Where, $\hat{n}_{ik}$ is the normal unit vector of the face of the particles where the contact point is located; and $v_p(t)$ and $v_{pk}(t)$ are the velocities of the faces ate the contact point of the bodies i and k respectively More information about the calculation of these velocities can be found in [141]. The same procedure can be followed to calculate the tangential forces at the body- container wall contact points: $F_{iL,t}(t)\ and\ F_{iB,t}(t)$, by equalizing the speed on the container's face to zero.

To evaluate if dynamic friction should occur at the contact point, a threshold ε is considered. Then, the collision can be classified into two categories: collision rejected or collision found, according to the relative velocity at the contact point and the limite considered, ε, which in this case is equal to zero (see Equations 2-17 and 2-18).

$$v_{pik}(t) = \hat{n}_{ik}\left(v_{pi}(t) - v_{pk}(t)\right) > 0 \qquad 2\text{-}17$$

$$v_{pik}(t) = \hat{n}_{ik}\left(v_{pi}(t) - v_{pk}(t)\right) > 0 \qquad 2\text{-}18$$

Equation 2-17 corresponds to the positive values, which indicate that the bodies are separated. The contact point is rejected and the update solver can continue working. Equation 2-18 indicate that a collision exists.The normal force acting on the object can be described as the sum of the normal force produced by two bodies in contact plus the normal forces produced by the container walls where there are placed.

Normal forces

The total normal force is calculated in a similar way as in the equation 0-14 for the tagential forces; considering the normal forces acting at the contact points of the ith body while is colliding with the lateral and bottom walls of the container; and normal force due to interaction between the ith and kth bodies.

After a collision detection, the normal relative velocities have to be reversed in a discontinuous manner to avoid interparticle penetration. Since the iteration of the physics engine works continuously to make the movements realistic, the one in charge of reversing the velocities is the *Narrow phase* of the collision detection algorithm (see Section 2.4.2). This will ensure a smooth variation of the state of the vector during the time interval. Thus, the Broad phase is stopped and the Narrow phase traces the trajectory of the object. This is done at the time step before the contact time (t_c).

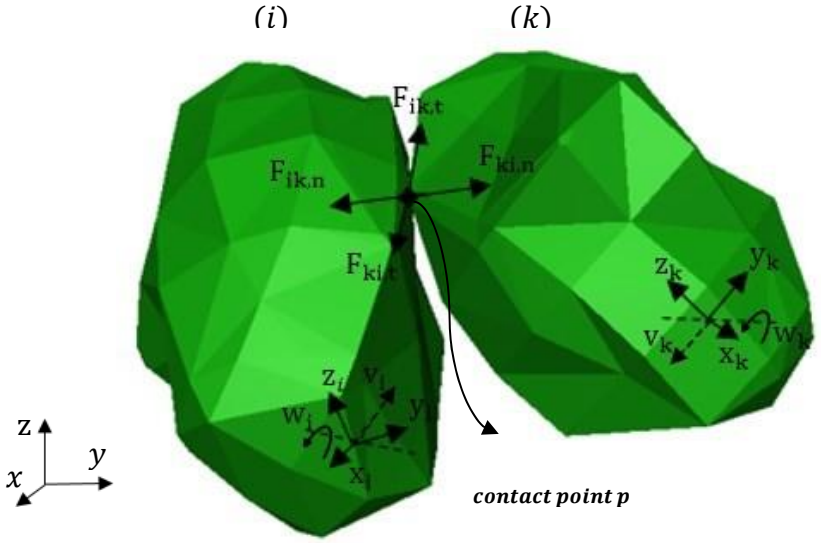


Figure 2-24 –A collision representation of two aggregates modelled as rigid bodies: *i* (left) and *k* (right), respectively

Note: the spatial variables of two colliding rigid bodies and the normal and tangential forces acting on the contact point are represented in both particles.

Impulse forces during a collision

This algorithm calculates the Time Of Impact (TOI) to go back in time to process the collision at that moment, assuming that: 1) the state of the colliding bodies do not change during the collision period (t_c) and 2) the state of the colliding bodies change discontinuously at the end of the collision period (t_{m+1}); knowing that a collision occurs during the Time of Impact t_c which belong to $[t_m, t_{m+1}]$.

The impulse can be defined as:

$$\mathrm{J}_i(t) = \int_{t_m}^{t_{m+1}} \mathrm{F}_i(\mathrm{t})\mathrm{dt} = \Delta P_I|_{t_m}^{t_{m+1}} \qquad 2\text{-}19$$

The above Equation indicates that the sum of all the collision forces acting on the body i correspond to the difference in its linear momentum.

Assuming that at the contact time t_c, the vertex contact point is the same for a pair of colliding bodies i and k: $\mathrm{p}(t_c) = \mathrm{p}_k(t_c)$, the velocities of the bodies have to be changed to avoid interpenetration. For this case, the Impulse for that is described as:

$$J_{ik}(t_c) = J_{ik,n}(t_c) + J_{ik,n}(t_c) \quad 2\text{-}20$$

$$J_{ik,n}(t_c) = j_{ik,n}(t_c)\,\hat{n}_{ik}(t_c) \quad 2\text{-}21$$

$$J_{ik,t}(t_c) = j_{ik,t}(t_c)\,\hat{t}_{ik}(t_c) \quad 2\text{-}22$$

Where, $j_{ik,}$ and $j_{ik,t}$ are the undetermined scalar magnitudes of the normal and tangential impulses respectively. These impulses act on the face of the body where contact occurs. Applying Newton's third law: if the impulse J_{ik} acts on the body *i*, the body *k* will be subjected tothe same but opposite impulse, $-J_{ik}$ to satisfy the momentum conservation law. The impulse acting on body i (due to its collision with body k) produces an impulsive torque, which can be expressed as:

$$\tau_{j,ik}(t_c) = (p_i(t_c) - x_i(t_c)) \times J_{ik}(t_c) \quad 2\text{-}23$$

The algorithm uses an empirical model that connects the velocities after and beforethe collision based on de Coefficient of Restitution (COR) as we can see in Equation 2-24. This serves to know the magnitude of the impulse in the normal direction $J_{ik,}$.

$$v^{+}{}_{pik,n} = (-\mathrm{COR})\, v^{-}{}_{pik,n} \quad 2\text{-}24$$

Where, $v^{+}{}_{pik,n}$ and $v^{-}{}_{pik,n}$ are tentative normal velocities in the contact point before and afterthe collision respectively; and COR is a number between 0 and 1 where 0 corresponds to a maximun kinetic energy lost and 1 means a perfectly elastic collision without energy lost. Finally, the impulse resulting from substituting the relative normal velocities in Equation 2-24, the magnitude of the normal impulse is written as follow:

$$j_{ik,n} = \frac{-(1+\mathrm{COR})\, v^{-}{}_{pik,n}}{\hat{n}^{T}{}_{ik}\, K_{ik,n}\, \hat{n}_{ik}} \quad 2\text{-}25$$

Where, $K_{ik,}$ is the collision matrix between the objects being in contact. More information can be found in [141]. In the case of an object colliding with the walls of the container (lateral or bottom), the magnitude of the impulse can be derived in a straightforward way, as described in [141].

On the other hand, the tangential impulse at the contact point is defined as theproduct of the normal impulse magnitude and the coefficient of friction μ, according to the Coulomb friction model:

$$j_{ik,t} = \int_{t_m}^{t_{m+1}} \mu f_{ik,n}(t)dt = \mu j_{ik,t} \qquad 2\text{-}26$$

Once the total impulse, $J_i(t_c)$, has been calculated, the changes of linear and angular velocities of body ith, are expressed through equation and equation 2-27 and 2-28 respectively.

$$v_i^{+}(t_c) = v_i^{-}(t_c) + \frac{J_{ik}(t_c)}{m_i} = v_i^{-}{}_i(t_c) + \frac{j_{ik,n}(\hat{n}_{ik}(t_c)+\mu\hat{t}_{ik}(t_c))}{m_i} \qquad 2\text{-}27$$

$$w_i^{+}(t_c) = w_i^{-}(t_c) + I_i^{-1}(t_c) \times \left(r_i \times J_{ik}(t_c)\right) \qquad 2\text{-}28$$

For the calculation of the linear and angular velocities of the body kth, the same two above equations can be used, by substituting $-J_{ik}(t_c)$ in the above equations. Finally, the impulses are calculated and the Update solver starts up again, integrates the impulsive forces into the system and calculates the new position of the bodies, (Equation 2-12), which is changing over time.

Through these sections we have briefly described how the collisions systems work in a Physics Engine, to calculate and detect the collisions, based on the impulse force approach, assuming not deformable objects.

2.5 Techniques employed for modelling virtual aggregates.

It is very important to choose a suitable method to generate virtual particle´s samples as realistically as possible, finding a balance between accuracy and performance during the simulation. In this section we focus on the user-defined techniques found in the literature, which refer to those virtual particles which are modelled directly throughcomputer programs.

The techniques for modelling aggregate can be divided into two main branches: 1) Imaging a physical specimen, as can be seen in [13] and [101], which present great precision but they can be very costly in both terms: economical and computational, when it comes to simulate full coarse aggregate gradation of stone-base mixtures, and 2) virtually building the model using a computer program (user-defined models), which has less precision but they can recreate full stone´s samples used in stone-based mixtures such as asphalt, ballast or granular soils. as we can read in [6], [103], [9] respectively, in a more streamlined way.

Virtual particles are modelled directly through computer programs, which allow the user to control some morphological parameters. Additionally, they have lower computational costs when implementing the particles to simulate stone-based materials, as the aggregates are simplified to polyhedral shapes.

As described in Section 2.3.2, some traditional DEMs offer user-defined particle models, ranging from highly idealized particles to complex shapes that resemble the actual shape of the aggregates. The latter can be found in [1], where a stone´s library with a range of morphological features was created in the DEM program BLOKS3D [3] (see Figure 2-16), under the purpose to study the effect of the gradation and shape of the aggregate in the ballast performance.

In that research, the authors performed a preliminary stone characterization by using the University of Illinois Aggregate Image Analyzer" (UIAIA), described in Section 2.2.2. Three orthogonal views of 2D images captured by UIAIA (Figure 2-25.a) were used to create the 3D model in the DEM software by first adjusting these three 2D orthogonal views intersection until the shape indices measured from the particle´s projections (area, Angularity Index and Flat and Elongated index) match the one computed in DEM. These 2D images were extruded, as shown in Figure 2-25.b, to finally generate the 3D virtual stone in the BLOKS3D DEM program Figure 2-25.c.

The particle size is given as a unit volume which can be scaled to match a real gradation. This virtual aggregate model's laboratory validation matched the experimental results of a box direct shear strength test with the simulated one by previously calibrating a set of DEM parameters. Others works that use the same techniques can be found in [104] and [142].

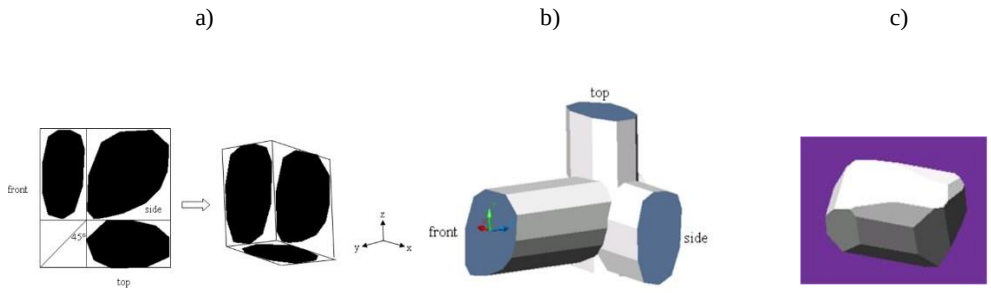


Figure 2-25 – Virtual particle creation through the extrude process from real 2D images stones views

Note: a) three 2D images from the orthogonal views of an aggregate captured from UIAIA, b) intersection of the three 2D orthogonal views extruded, c) 3D aggregate created in BLOKS3D. Image from: the role of aggregate in concrete countertop mix formulas [16].

2.5.1 3D random particle´s generation models

These models aim to generate virtual particles with irregular shapes without the need for stone´s pictures by using stochastic algorithms, which allow the generation of

a wide range of particles with different and unique morphologies in each aggregate´s sample.

For instance in [103], a large-scale simulation of an oedometer test is performed in the YADE open-source DEM software [138], to study the ballast grains behaviour. The particles were modelled as random convex polyhedrons using a Voronoi Tessellation algorithm [68].

The procedure is as follows: first, given a set of points with randomly coordinates in an initial volume (Figure 2-26.a), the Voronoi Tessellation algorithm subdivides the volume into n cells (one cell for each point being the points the nucleus of the cell) (Figure 2-26.b), with a minimum distance between nucleuses previously established; then the Voronoi cell corresponding to the central nuclei of the volume is extracted as a basic particle shape (Figure 2-26.c).

Although the particles are created by a random procedure, these are later rescaled so that their aspect ratios match the most common cases which are: cubic (x = 1, y = 1, z = 1), elongated (x = 2, y = 1, z = 1) and flat (x = 2, y = 2, z = 1). Additionally, the particle are randomly rotated to avoid directional bias.

In this work, the authors consider the shape of the particle when measuring the sieve size by calculating the smaller size of a square´s side, which encloses the particle's projection in the plane perpendicular to its lowest axis of inertia. Although the scaling of Voronoi cells allows the user to specify the particles' size and aspect ratio, these techniques do not consider the real size´s distribution and shape´s distribution of a real sample.

The Voronoi tessellation approach is also employed in [9], to simulate random stones´ shapes. In this case, a laboratory sieve analysis of a real gravel sample was made

to calculate the volume of stone that remained on each sieve size. Then, a virtual block is constructed for each grain size range, with the same volume as the one obtained in the laboratory; finally a Voronoi Tessellation algorithm was applied to obtain random polyhedrons from the block, by matching the average size of the virtual grain with the average size of two successive mesh sieve. This random shape gravel generation model was validated by counting the number of stones of each sample with the virtual stone generated.

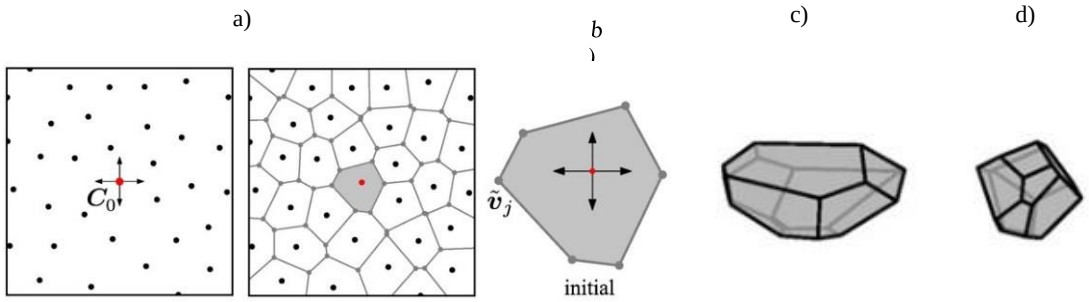


Figure 2-26 – Virtual Particle generation through Voronoi Tessellation procedure

Note: Figures: a, b and c Voronoi Tessellation procedure. Figure. D. Particles created with different aspect ratios. Image from: Simulation of railway ballast using crushable polyhedral particles [103].

Another technique found in the literature to simulate random geometry of different coarse aggregate´s sizes is the Quickhull algorithm, as can be seen in [6]. In this paper, the author model the rocks through the Bullet Physics Engine, to simulate the grain´s movements during an asphalt compaction process.

The procedure is as follow: first, arbitrary points are selected from the surface of a sphere under the hypothesis that real aggregates have between 16 and 20 vertices. A database with various coarse particle sizes is created, assuming that their sieve size equals the sphere's diameter. The sphere's radius is a random number among a set of numbers whose distribution probability is obtained from a sieve analysis from a real

sample. Then, the convex hull geometry of the aggregates is constructed using the QuickHull Algorithm [7].

The QuickHull algorithm is based on a divide and conquer approach where: 1) first, each aggregate is simplified to a cube which encloses all the cloud points which define it; 2) the vertices furthest from the cube are joined by forming surfaces to and they also form part of the Convex Hull of the particle. 3) The vertices inside the volume are ignored; those outside are used to form new surfaces between the points farthest from each other. This is a recurring process that ends when all the cloud points that form the aggregate are, either within the formed volume or form part of the convex Hull. In Figure 2-27 an aggregate created using the QuikHull algorithm is pictured.

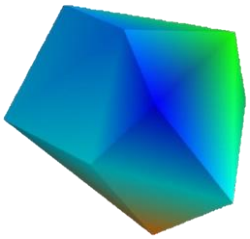


Figure 2-27 – Virtual particle aggregate created using the QuikHull algorithm

Note: Image from: Prediction on dynamic modulus of asphalt concrete with random aggregate modeling methods and virtual physics engine [6].

A similar procedure as the previously described can be seen in [105], where the authors use a convex Hull algorithm to define the surface of the stones, by joining the most “external points” of 50 points arbitrarily placed in the space, through the Matlab software, as plotted in Figure 2-28.

The stones' aspect ratio is taken into account by choosing an arbitrary interval in each direction for the point’s generation. Although different samples can be created, the same sample's virtual aggregates will have the same aspect ratio previously definedby

the user. Further, the stone samples follow a gradation chosen by the user, by whichthe particles (or the distribution of points in space) vary their size randomly accordingto it; although there are no explanations about from which axis or plane of the particlethe grain´s size is modified.

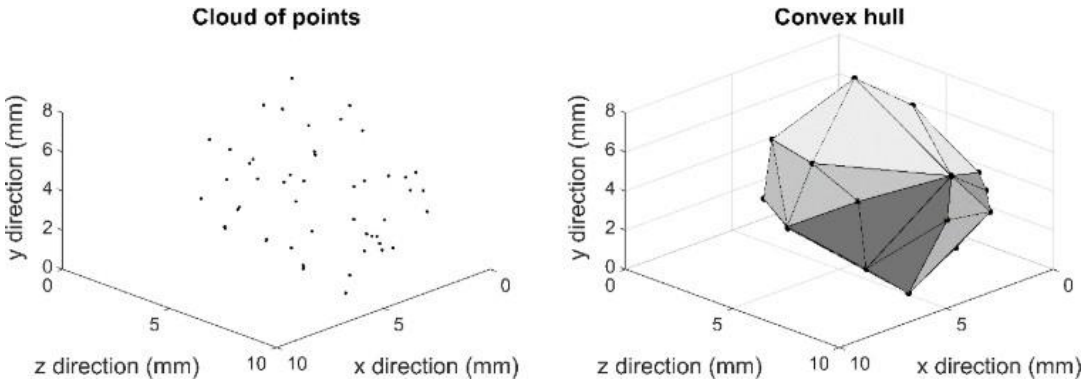


Figure 2-28 – Virtual particle aggregate created using the Convex Hull algorithm

Note: Image from: Computational Packing of Aggregates for the Study Of Virtual Asphalt Samples [105].

In [110], the authors also generated random convex aggregate as follow: 1) a spherical particle is generated; 2) a Perlin noise with a controlled amplitude factor is added to the sphere to achieve irregular shapes (see Figure 2-29); 3) particles are rescaled base on either: the equivalent radius of a sphere or on the particle´s volume. 4) Finally, the aggregate´s samples are based on user defined parameters as: number of grains, size´s distribution and Perlin noise amplitude.

The virtual particles are characterized by their surface mean curvature and their sphericity. The definition of these shape´s factors can be found in [143] and [144] respectively. Nonetheless, this characterization is not based on or compared to real stone samples.

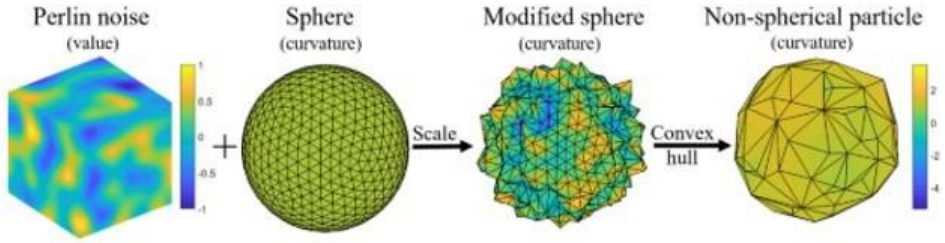


Figure 2-29 – Creation of convex irregular particle´s using a Perlin noise
Note: Image from: Particula: A simulator tool for computational rock physics of granular media [110].

2.5.2 Conclusion

All the works described above have meant a great improvement when it comes to simulating irregular particles, since these stochastic techniques are very efficient when it comes to generating a great variety of grain´s morphology features; since the user can modify them to adjust their size and shape distributions. Besides, they allow to compute larger number of particles during the simulation of granular materials.

However, they still have some disadvantages such as: 1) the methods show low accuracy when it comes to representing particle´s shape; since those are simplified to simple polyhedral shapes, do not take into account neither the concavities that the aggregates used in stone-based materials usually have, nor the different levels of angularity (except in [110] where different level of angularity are taken into account); 2) the lack of parameterized morphological features, to compare the real and the virtual aggregate´s samples.

2.6 Conclusion from the literature review

As we have seen throughout this literature review, in civil engineering, the mesoscopic study of stone-based materials is a complex subject, due to its discrete

nature, where changes in the morphology of their aggregates lead to changes in the behaviour of the materials (Section 2.1).

Several parameters to quantify both: the individual aggregates and the granular material exist, to provide a better understanding of their final performance (Sections 2.2 and 2.3.3); as well as several specifications and restrictions about the grain´s size distribution and grain´s shape exist; in order to select the proper aggregate´s sample for the mixtures design. Nonetheless, there is no standard to determine how particle´s changes influence the overall material behaviour [145].This is mainly because laboratory studies for the aggregate´s characterization and for the material behaviour are subjective, expensive and time-consuming, since they are subjected to trial and error tests.

For this reason, the investigation of this subject has been the object of numerous studies by using both: experimental and simulation tests (Section 2.3). However, experimental studies' biggest disadvantage is systematically incorporating different particles to understand how the grain´s shape influences the porosity, stone interlock, amount of other properties, and the structure they form.

To overcome this limitation, and thanks to computer technology advances, robust prediction and simulation computer models are available. The most widely used computer tool for the simulation of particle´s assemblies, is the traditional Discrete Element Method (DEMs), as mentioned in Section 2.3.2. Nonetheless, those are normally designed to represent the particles as spheres or regular polyhedrons, due to the computational cost of computing the interaction between the numerous particles of a granular assemblage, especially if those present concavities [146].

For this reason new alternative to simulate realistic grain´s pack has begun to be used, such as the impulse-based contact model (summarized in Section 2.4): The impulse Discrete Element Method (iDEM) or the Physics Engine (PE), which are computationally more efficient, as is demonstrated in [13]. Although several authors have demonstrated the feasibility of the PEs to simulate granular material (see Section 2.3.2.2), their accuracy is still disputed.

On the other hand, creating realistic virtual irregular grain´s shape samples can be computationally challenging, as was described throughout Section 2.5. It is necessary to create a computer model capable of generating a wide variety of stone´s sizes and shapes to simulate full coarse aggregate´s samples of stone-base mixtures in a streamlined way. For that purpose, the user-defined techniques are the most appropriate, as explained in Section 2.5.

Most of the previous works oversimplified the particles to convex polyhedrons in which neither the different levels of angularity nor the concavities are accurately represented. They also simplify the particles' sieve size to the equivalent diameter of a sphere, and the sizes distribution and shape´s distribution are not taken into account (Section 2.5.2).

Motivated by the above mentioned, in the present thesis, a user-defined technique to create an aggregates generation model with realistic shapes, developed in a Physics Engine, is presented. Unlike the works described in the literature, the virtual stones present realistic morphological features, including concavities and can create stone´s samples which follow realistic size and shape distributions. This aggregate generation model approach was validated by comparing real morphological parameters withvirtual ones.

The final goal is to compute several particles' packing, under the purpose to demonstrate the feasibility of the PEs to reproduce particle´s structure accurately.

CHAPTER 3. EXPERIMENTAL WORK

For the development and validation of the aggregate´s packing simulation model developed throughout this work, a series of experimental works have been carried out which are divided into:

(i) Particle's samples characterization. The characterization of aggregates using an Image Analysis software. This information is used for the creation of the random virtual aggregate generation model presented in Chapter 4.

(ii) Preparation and analysis of the real aggregates packing specimen. The manufacture of different coarse aggregate packing in dry conditions and the analysis of their structures to relate the stones' size and morphology to the packing characteristic. This is used to validate the virtual aggregate packing model presented in Chapter 5 by comparing results obtained during the simulation with the actual data.

(iii) Preparation and analysis of the real aggregates packing specimen. The manufacture of various asphalt mixtures with different gradations and percentages of bitumen in order to measure their porosity. This will enable verification of the virtual asphalt skeleton packing model presented in Chapter 6, by studying its ability to predict the Air Void content based on volumetric calculations.

3.1 Particle´s samples characterization

The first objective of the experimental program is to define the morphological features of coarse aggregates to parameterize any particle samples.

An Image Analysis software, able to generate values of different stones' shape characteristics, was used in conjunction with a visual evaluation chart classification for stones to evaluate, quantitatively, the coarse aggregate samples used in this study. The proposed evaluation test is an alternative to the tedious and time-consuming manual methods and the economically inaccessible Image Analysis system, both described in Section 2.2.

The parameters derived from the experimental analysis were used for the creation and validation of the random aggregate generation model described in Chapter 4, using simple mathematical and statistical models.

Several 2D morphological factors have been extracted from eleven coarse aggregate´s samples by using an Image Analysis technique. The technique selected had the advantage that both qualitative and quantitative size and shape information can be obtained simultaneously (see Section 2.2.2). Besides, the gradation´s curves obtained by the Image Analysis are more accurate since sieve analysis is not sensitive to shape´s variation [147]. The output of this analysis has been statistically studied for the particle´s samples quantitative characterization.

Additionally, the distribution curves of four coarse aggregate samples (from the same sources but with different sizes), were subjected to a comparison between the traditional sieve analysis, the Image Analysis previously mentioned, and the Dynamic Digital Image Analysis through the Camsizer device (described in Section 2.2.3); since

the different procedures for size measurement can lead to different particles size distribution results for the same particle sample. It is necessary to consider these differences when entering the probabilistic data into the virtual aggregate generation model presented in this work. This study can be found in Appendix A.1.4.

3.1.1 Image-based method to quantify Coarse Aggregates samples

Eleven different aggregate samples with different particle shape characteristics have been used in this study named: Glass Spheres (GS); Crushed Glass (CG); Slag (S); Round Gravel (RG); two types of granites, both having14mm maximum size (G1 & G2) and two types of Limestone. One of the limestones was given the symbol L1, while the other was available at four sizes so is given four symbols – L2, L3, L4 and L5, with maximum sizes 6, 10, 14 and 20mm respectively. Figure 3 1 illustrates some of the stone types studied.

The aggregates used are those found in the laboratory of the University of Nottingham and those provided by the Tarmac Company, which collaborated with this research.

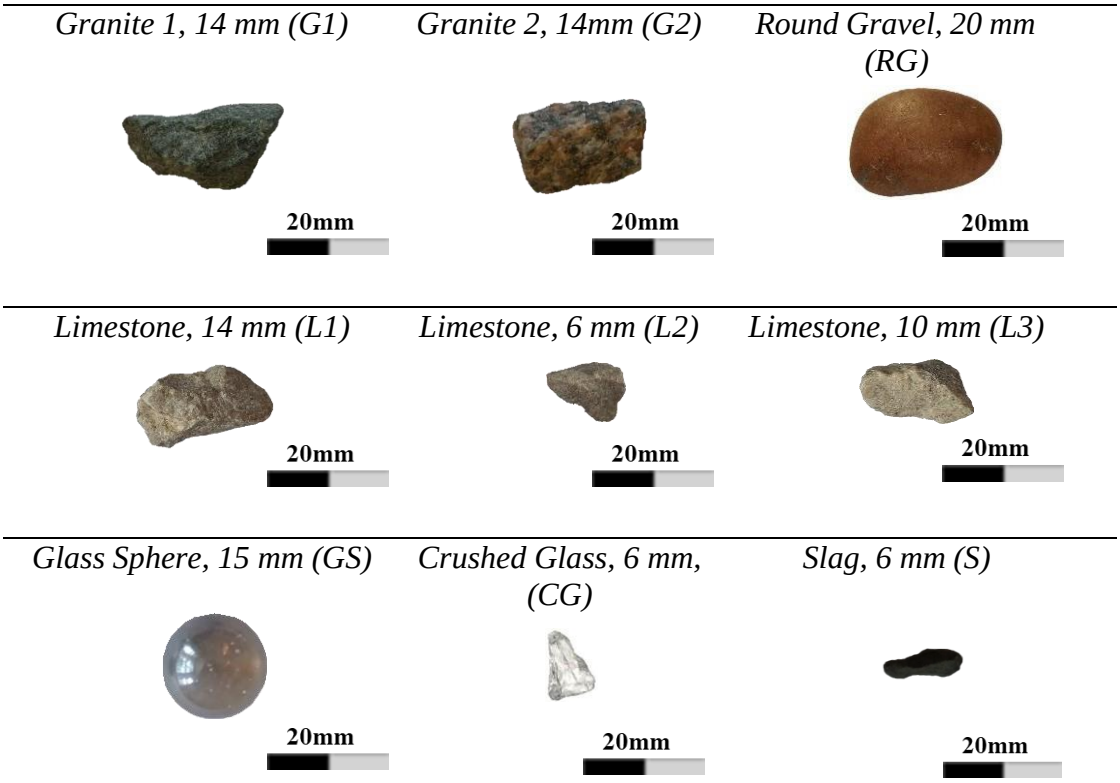


Figure 3-1 –Pictures of some stone´s type analysed

About 1kg of each aggregate was evaluated through the Digital Image Processing software *ImageJ* to determine the size and shape of the individual particles. Pictures have been obtained from the aggregates resting in their most stable position. The analysis of their 2D morphological feature has been carried out by the digital image processing software ImageJ, following the procedure described in Appendix A.1.1.

The dimension which determines the sieve size of the aggregate is represented by the Minor Feret diameter (MF) since it gives the closest size to a sieve analysis [52][76][148][149] . The remaining shape factors are Aspect Ratio (AR), Roundness (R), Area (A), Perimeter (P) and Circularity (C). The definition of these parameters can be found in Section 2.2.6, and their definitions are summarized in the following table.

Table 3-1 – Morphological parameters for particle´s characterization

Parameters	Formula	Reference	*Equation*
Major and Minor Feret diameter Perimeter, P Area, A	Measured with ImageJ	[76]	
Aspect Ratio, AR	$AR = \frac{Min\ Feret}{Max\ Feret}$	ISO9276-6, [150]	0-1
Sphericity/ Cox's circularity, C	$C = \frac{4\pi A}{P^2}$		0-2
Roundness/Pentland's circularity, R	$R = \frac{4A}{\pi\ Max\ Feret^{\ 2}}$	[82]	0-3
Thickness, β	$\beta = \frac{Min\ Feret}{Average\ height}$		0-4

Finally, the third dimension (height) of the aggregate samples was measured. It has been defined by a new parameter, β, which we will call thickness from now on, and is the ratio between the major Feret and the average height of 50 stones, measured with a calliper.

The distribution of the properties: minor ferret and aspect ratio (Table 3-1) has been analysed in the Minitab software to see which distribution function (i.e. Weibull, lognormal, amount other as can be seen in Appendix A.1.2) is the most representative of all the gradations, by using the Anderson-Darling adjustment, which indicates the goodness-of-fit to a distribution curve. The results showed that the Weibull distribution fits better for almost all the samples analysed (see Appendix A.1.2), which is in agreement with previous research [63] [151] [152].

The minor Feret and aspect ratio are the shape factors chosen to be implemented in the virtual aggregate model (Chapter 4), based on the hypothesis that they represent the gradation and the general form respectively of each type of particle. For that reason, their distribution curves are each characterised by Weibull statistics parameters. The scale and shape Weibull distribution parameters were found using an iterative least

square's technique, as explained in Appendix A.1.3. Further, the results obtained for all the morphological information presented in Table 3-1 are ranked in an Excel file, to quantify them in terms of the 50th percentile, (Per_{50}).

3.1.2 Results. Statistical parameters of the measured shape factors

Table 3-2 shows the values that characterize the aggregates studied, obtained from the statistical analysis described in Section 3.1.1 which are:

1) the statistical parameters of the Weibull distribution curve of each aggregate type for the minor Feret and aspect ratio shape factors. This distribution is defined by two parameters named: shape, k, and scale, λ and;

2) the 50th percentile values (Per_{50}) of all the morphological features analysed in this study (Table 3-1).

Further, Figure 3-2 shows the aggregate classification based on the index values obtained from the Image Analysis (IA) for circularity (or angularity) (C_{50}) and sphericity (R_{50}), as defined in this study, and the Krumbein and Sloss chart classification, used for visual evaluation of the coarse particles (see Section 2.2.6).

From Figure 3-2 it can be seen that only the slag sample (S) is classified as angular, while most of the samples are sub-angular, except for the RG and GS samples which are defined as sub-rounded and rounded respectively. Most of the samples have been labelled with low sphericity, except for the L1 and RG, which have medium sphericity; and the sphere (GS) sample which has high sphericity.

Table 3-2 – Experimentally measured shape factors that characterise the aggregates studied

	G1	**G2**	**RG**	**L1**	**L2**	**L3**	**GS**	**CG**	**S**
k_{MF}	12.57	5.69	4.45	13.84	3.76	3.66	379	5.12	8.54
λ_{MF} (mm)	15.07	13.27	19.18	15.22	7.86	5.23	14.9	8.7	5.34
$Min\ Feret_{50}$(mm)	14.62	12.36	16.82	14.68	7.19	4.76	15.12	8.04	4.92
k_{AR}	7.15	7.9	7.82	9.59	6.73	6.59	385	7.84	6.63
λ_{AR}	0.75	0.73	0.8	0.79	0.71	0.67	0.99	0.66	0.68
AR_{50}	0.71	0.69	0.76	0.78	0.68	0.65	0.99	0.64	0.66
$Max\ Feret_{50}$ (mm)	20.42	17.57	23.25	19.19	10.61	7.14	15.26	11.2	7.43
P_{50} (cm)	6.03	5.5	6.72	5.81	3.15	2.13	4.79	3.25	2.11
A_{50} (cm^2)	2.08	1.56	2.73	2.02	0.52	0.24	1.83	0.64	0.25
R_{50}	0.65	0.7	0.71	0.69	0.61	0.61	1.00	0.65	0.61
C_{50}	0.71	0.67	0.73	0.78	0.67	0.68	1.00	0.70	0.68
β	0.52	0.6	0.53	0.54	0.44	0.39	1.00	0.49	0.42

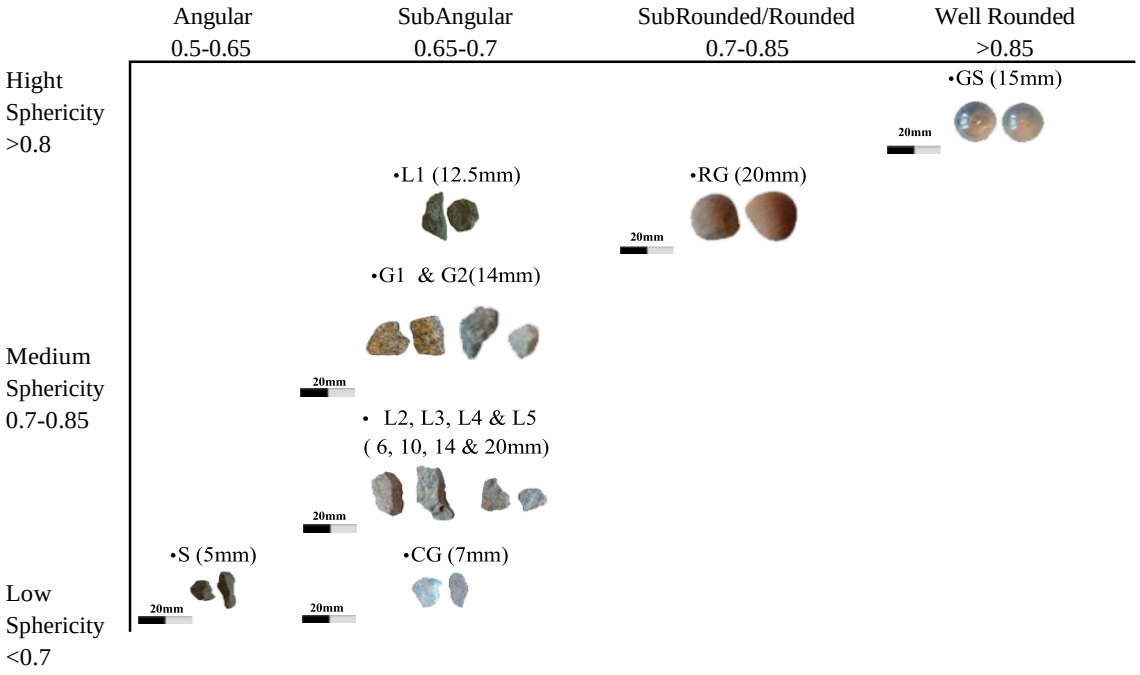


Figure 3-2 –Pictures of some Aggregate chart classification based on the IA Sphericity and Circularity Index and the Krumbein chart, used for a visual evaluation

From Table 3-2, it can be seen that, although with some errors, the characterization of the particles is correct as the Glass Spheres sample (GS) shows an aspect ratio approximately equal to 1, minor Feret approximately 15 mm, and area and perimeter like those of the circle with equivalent diameter. The GS sample is not

considered further in the analysis that follows as it is composed of identical spherical particles (i.e. there will be no useful statistical variability).

The samples that present the most uniform size distribution (highest k_{MF} values), are: L1, G1 and S; whereas L5 is the one with the least uniform size distribution, which makes sense since it is the limestone with the largest maximum nominal size (20mm).

Regarding the distribution of the aspect ratio, k_{AR}, the highest values are found for RG and L1, whereas L2, L3 and S present the lowest values.

Furthermore, the samples having thicker stones, with β greater than 0.5 are RG, G2 and L1, whilst the flakier aggregates are found in the samples L3, S, and CG.

3.2 Preparation and analysis of the real aggregates packing specimen

The present section focuses on how the different aggregates' size, shape and combination of stones provide different assembly structures and, therefore, variations in their pore-network and skeleton characteristics. Since the second goal of the present thesis is to corroborate the capability of the Physics Engines to simulate granular material realistically, the output of this study will serve for the creation and validation of the virtual aggregate packing model presented in Chapter 5.

The samples were scanned using X-ray computed tomography after compaction. Then, an Image Analysis technique, once again using the software ImageJ, was used to analyse them. The pore-structures were employed to quantify each specimen´s assembly characteristics, described in Section 2.3.3.

3.2.1 Specimens' preparation and compaction method

Single-sized packed specimens of each of the materials presented in Table 3-2, G1, G2, RG, L1, L2, L3, GS, CG and S were made (Figure 3-3.a). Besides, L2, L3,L4, and L5 materials were combined in mixtures, with maximum nominal size: 20 mm (VS20), 14 mm (VS14), and 10 mm (VS10) and gradation curves that follow different of a Bituminous Macadam open graded asphalt gradation (BM) (Figure 3-3.b), in accordance to the British Standard BS 4987-1:2005 specification [153]. These coarse aggregate proportions can be found in Table 3-9 (mixtures: Mix7, Mix 5 and Mix 3 respectively). The reason to select this gradation is simply to demonstrate that we can simulate and compact realistic aggregate gradations. We do not intend to compare the end compacted materials with actual asphalt mixture compaction. Note that bitumen was not used in the experiments or simulations.

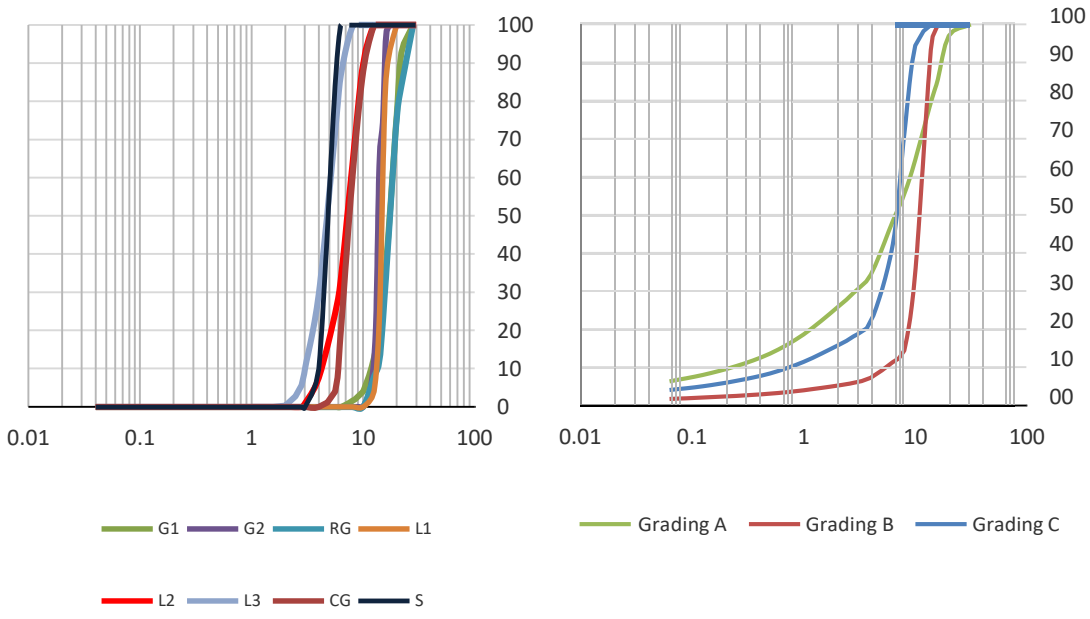


Figure 3-3 –Gradation curves of a) single-size aggregate´s mixtures and b)various-size aggregate´s mixtures

All the packings weighed approximately 500 g ± 5 g. These were placed in a transparent cylindrical container to compact the aggregates, whose dimensions are 9.4 cm in diameter and 8 cm in height (see Figure 3-4). The compaction test was conducted using a vibratory table, as used to compact concrete, which has flexible springs and an engine speed of 3600 r.p.m (60 Hz), making the table vibrate horizontally and sinusoidally. The displacement in the 3 axes ranged between 0-1.58 mm. Note that it is not intended to reproduce the compaction of asphalt in the field but use a methodology that can be easily reproduced in the physics engine to compact the aggregates.

The compaction was performed in two steps: firstly, the specimens were vibrated for 2 minutes; secondly, a metal cylinder weighing 2.7 kilograms of 8 cm diameter was dropped from a 5 cm height above the aggregates and, the vibration continued for 5 additional minutes, to achieve the highest compaction possible of the particles. The number of stones in each mixture and the final height of all the mixtures after compaction were measured. These were used later in the packing process simulation presented in Chapter 5.

Finally, all the specimens were scanned by an X-ray Computed Tomography as described in [154], and the images were treated with the Image processing software ImageJ following the procedure described in [89] for further analysis, as described in the next section.

a) Sample before the compaction b) Sample after the compaction

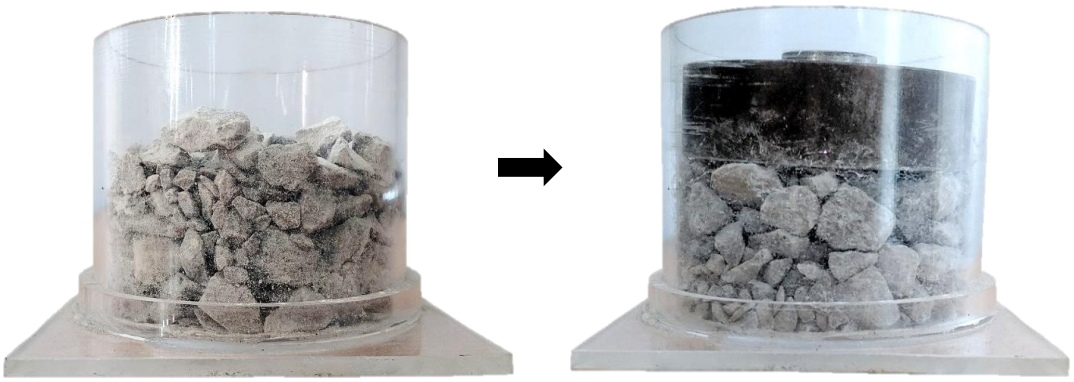


Figure 3-4 – Mixture (a) before the compaction; (b) after compaction

3.2.2 Internal Structure factors used for the aggregate assembly characterization

The analysis of the different specimens was made by using the image processing software ImageJ to characterize different aspects of the void´s properties and the particles' properties. All the factors measured, and the algorithm plugin employed are summarized in Table 3-3.

The voids' properties are considered as basic descriptors of the porosity network [115][116][154]; while the particles' properties represent an approximation of the aggregate's skeleton structure, as previously described in Section 2.3.3.

The particle´s specific gravity, the number of particles in the mixtures as well as the height of the samples after compactions were also measured.

The percentage of air voids across the specimen height was also examined. The segregation and orientation of the aggregates for the specimens having different sized particles were also studied. The measurement process of these last parameters is explained below.

Table 3-3 – Parameters to characterize the internal structure of the specimens

Air Void Properties				Particle properties			
Measure	Plugin algorithm	Ref	Eq	Measure	Plugin algorithm	Ref	Eq
Porosity (p)	Volume Fraction/BoneJ	[155]	2-4	*Average & Maximum particle size ($\bar{\Phi}$)*	Thiknesses/BoneJ	[156]	-
Average & Maximum pore size ($\bar{\Phi}$)	Thiknesses/BoneJ	[156]	-	*particle's node density (n_d)*	skeletonize & analyze skeleton/BoneJ	[157]	2-9
Void density (n_v) and void shape: Aspect Ratio (AR) & Circularity (C)	particle analysis	[158]	2-9 2-1 2-2	*Particle segregation and particle shape: Aspect Ratio (AR) Circularity (C) and Roundness*	particle analysis	[158]	2-1 2-2 2-3
Tortuosity (τ) & Global pore-connectivity (Z)	skeletonize & analyze skeleton/BoneJ	[117]	2-5 2-7	*Particle Orientation*	Minimum Distance (Macro)	[159]	-

The correlation between the various parameters studied is described here using the Pearson's correlation coefficients (r), which values vary between -1 and 1, being 0 when there is no correlation at all, 1 when a perfect correlation and -1 when a perfect inverse correlation [89].

As mentioned, all the parameters presented in Table 3-3 were described in detail in Section 2.3.3. Nonetheless a summary is presented below.

The porosity is the volume of pores (V_p) that exist in the total volume analyzed (V_T).

The voids density , n_v, is the total number of voids per unit volume.

The particle's node density, n_d, is the total number of junctions, n_j, per unit volume. It serves as an approximation of the interconnected particle structure.

The average pore-connectivity, z, is calculated as the average number of branches of the medial axis meeting at a junction [116] see Equation 2-7.

The average geometric tortuosity, r_{geo}, (Equation 2-5) was calculated as the ratio of the geodesic length (d_g) to the Euclidean length (d_e) of all the pores in the volume examined [160][115].

Particle´s segregation

In this study, vertical and radial segregations were quantified by dividing the samples into different sectors with equal areas, differentiating between the top and bottomregions and the outer and inner regions, as is described in [161].

The image gradations were plotted as a function of the minor Feret diameter and compared between the different sections. The ratio of average aggregate size between sections was calculated using Equation 3-5. When SL is equal to zero this indicates no segregation. A positive value indicates a greater presence of coarse aggregate in the outer or top regions whereas a negative value means the opposite.

$$S_L = \left(\frac{Average\ diameter\ of\ aggregates\ in\ the\ outer\ or\ top\ region}{Average\ diameter\ of\ aggregates\ in\ the\ inner\ or\ bottom\ region} - 1\right)\ x\ 100 \qquad 3\text{-}5$$

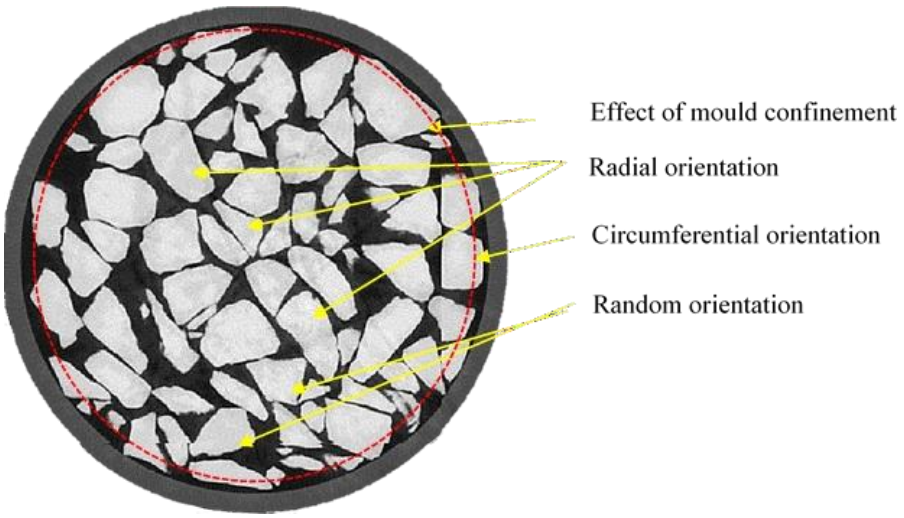
Particle's orientation

The orientation of the individual stones within each assemblage has been measured using ImageJ to determine the angle between their longest axis, or major Feret diameter, and the radial lines from the compacted specimen's centre to the centroid of the particles, as is shown in Figure 3-5.b The directional distribution of the particle is quantified using the average angle of particle orientation, ε, through Eq.3-6.

The orientation angle $\bar{\varepsilon}$ ranges from 0º to 90º, where 0º is a particle perfectly aligned with the radial axis (radially oriented); and 90º is a particle oriented in the circumferential direction and and values around 45º are considered to indicate random alignment [161] [162].Figure 3-5.a illustrates these different orientations.

$$\bar{\varepsilon} = \frac{\sum_i^n \varepsilon}{N} \qquad 3\text{-}6$$

a)



b)

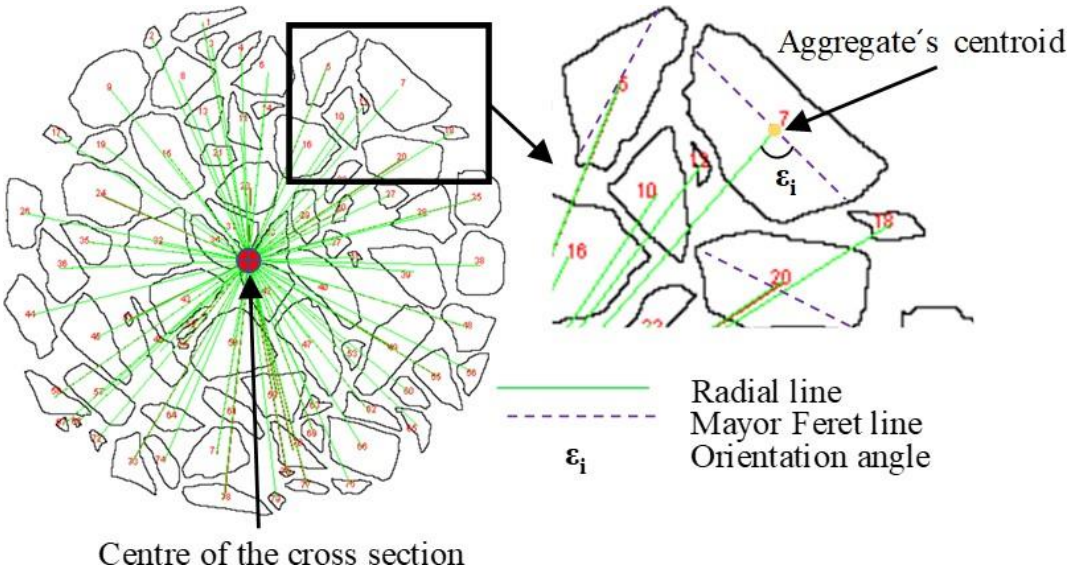


Figure 3-5 – a) different type of particle orientation. b) A Measurement of the orientation angle.

Note: a) shows the image previous to the ImageJ orientation´s analysis of the VS20 packing´s sample.

3.2.3 Results.

3.2.3.1 Internal structure characteristics of the experimental aggregate´s packings and their Pearson correlation

The particle´s density and the number of particles in the mixtures as well as the height of the mixtures after the compaction process are plotted in Table 3-4. The first two parameters serve to ensure that the number of particles generated during the simulation process agrees with the physical samples. In contrast, the height of the mixtures, once compacted, serves to stop the computed aggregate packing process (Chapter 5).

Table 3-4 – Particle´s density, number of particles in the mixtures and, height of the mixtures

	G1	**G2**	**RG**	**L1**	**L2**	**L3**	**GS**	**CG**	**S**	**VS20**	**VS14**	**VS10**
Number of particles	170	104	83	160	2304	905	102	1340	2230	789	507	581
height after compaction (cm)	5.3	4.9	5.4	3.4	4.5	5.5	6.2	5.6	3.4	2.8	3.2	3
Particle´s density (gr/cm^3)	2.81	2.79	2.59	2.68	2.70	2.70	2.58	2.31	2.72	-	-	-

*VS is Various-Sized Mixtures and the numeric indicates their different nominal maximum size (in mm).

The packing characteristics and Pearson's correlation coefficient (r) are shown in Table 3-5 and Table 3-6. For the various-sized specimens taken from the same source (VS20, VS14, VS10), the shape of the particles has not been taken into account for the analysis. However, the ratio between the average aggregate size and maximum aggregate size has been added to see how this influenced the mixtures.

Table 3-5 – Void and particle characteristics derived from CT-scans for the particle´s mixtures

			G1	G2	RG	L1	L2	L3	GS	CG	S	VS20	VS14	VS10
Air Void Properties	Porosity (p)	%	34.3	35.6	28.2	31.1	39	38.1	36.4	31.5	42	34.5	36.3	39.3
	Average pore size ($\bar{\Phi}_v$)	cm	0.32	0.412	0.394	0.395	0.17	0.253	0.582	0.182	0.178	2.97	2.99	2.19
	Maximum pore size ($\Phi_{v,max}$)	cm	0.72	0.94	0.812	0.891	0.43	0.757	0.979	0.476	0.436	7.24	10.02	5.14
	Tortuosity (τ)	-	1.26	1.3	1.41	1.39	1.3	1.33	1.19	1.64	1.24	1.21	1.17	1.17
	Connectivity (z)	-	3.23	3.17	3.2	3.18	3.3	3.21	3.17	3.28	3.29	1752	2221	2804
	Pore density (n_v)	-	0.0556	0.0094	0.004	0.011	0.025	0.03	0.009	0.132	0.0357	3.2	3.23	3.1
	Aspect Ratio (AR)	-	0.47	0.41	0.34	0.42	0.469	0.439	0.5	0.449	0.45	0.47	0.48	0.48
	Circularity	-	0.6	0.45	0.52	0.428	0.6	0.552	0.41	0.589	0.572	0.55	0.5	0.53
Aggregate Properties	Average aggregate size ($\bar{\Phi}_a$)	cm	0.57	0.795	1.063	0.782	0.256	0.452	1.48	0.416	0.279	6.77	5.43	4.01
	Maximum aggregate size ($\Phi_{a,\,max}$)	cm	1.181	1.18	1.608	1.269	0.525	0.947	1.631	0.874	0.593	17.83	11.42	7.25
	Node density (n_d)	-	0.076	0.015	0.013	0.019	0.115	0.072	0.002	0.211	0.116	0.04	0.02	0.04
	Roundness	-	0.65	0.7	0.71	0.69	0.61	0.61	1	0.65	0.61	-	-	-
	Circularity	-	0.71	0.67	0.73	0.78	0.68	0.67	1	0.7	0.69	-	-	-
	Aspect Ratio	-	0.69	0.71	0.76	0.78	0.65	0.68	1	0.64	0.66	-	-	-
	Size's ratio	-	-	-	-	-	-	-	-	-	-	2.63	3.35	2.34

Table 3-6 – Pearson's correlation coefficients between all the parameters studied

		Characteristics of air voids							
		p	$\bar{\Phi}_v$	$\Phi_{v,max}$	τ	z	(n_v)	AR	C
Void Properties	Porosity (p)								
	Average pore size ($\bar{\Phi}_v$)	-0.35							
	Maximun pore size($\Phi_{v,max}$)	-0.71							
	Tortuosity (τ)	-0.53	-0.37	-0.28					
	Connectivity (z)	0.16	-0.69	-0.72	0.02				
	Pore density (n_v)	0.16	-0.67	-0.67	0.48	-0.62			
	Aspect Ratio (AR)	0.67	-0.06	-0.06	-0.56	0.27	0.43		
	Circularity (C)	0.47	-0.86	-0.8	0.25	-0.83	0.55	0.42	
Aggregate properties	Average aggregate size ($\bar{\Phi}_a$)	-0.49	0.87	0.86	-0.24	0.67	-0.61	-0.72	-0.79
	Maximum aggregate size ($\Phi_{a,\,max}$)	-0.79	0.89	0.9	-0.07	-0.6	-0.67	-0.54	-0.67
	Node density (n_d)	0.15	-0.84	-0.88	0.57	-0.65	0.92	0.3	0.77
	Aspect Ratio (AR)	0.67	0.91	0.75	-0.47	0.67	-0.56	0.24	-0.8
	Roundness (R)	-0.69	0.84	0.6	-0.35	-0.43	-0.3	0.33	-0.62
	Circularity (C)	-0.43	0.83	0.62	-0.42	-0.42	-0.36	0.43	-0.72
	Size's ratio	-0.95	0.75	-0.95	0.94	0.61	-0.78	-0.35	-21

Table 3-5 reveals the uniform size sphere glass specimen (GS), having a porosity of 36.4%, which confirm that the spheres were near a close random packing condition, as explained in [163].

From Table 3-6, it can be seen that the porosity is inversely related to the maximum aggregate size – which is consistent with previous research [164][87] – having a Pearson coefficient (r) equal to -0.79.

The porosity is also inversely related to the size ratio, for the various-size specimens (r=-0.95). That means that a bigger difference between particles´ sizes yield to a lower porosity content, since the small particles fill the gaps between the bigger stones [165].

Particle´s morphology has also significant influence on the packing´s porosity, as the Pearson coefficient values between aspect ratio and porosity and between roundness and porosity are 0.67 and -0.69 respectively.

A general trend is observed in which particles that deviate from the spherical shape generate a greater pore content, as has also been found by previous studies [166][167].An example is observed when comparing L3 and CG, which have similar gradation andaverage maximum grain size; but their porosities are 38.1% and 31.5%, respectively.

Regarding the tortuosity, this has values ranging between 1.1 and 1.7 (Table 3-5). The lowest values were for the sphere and slag samples, which could be due to particle´s samples uniformity, producing fewer interruptions to the pore conduit [168].

Regarding the various-sized specimens, the size ratio and the tortuosity have a Pearson value of r=0.94. Adding larger materials to the mixture produce the coarser

mixtures have a wider range of sizes This could indicate that tortuosity increases with the range of sizes.

Moreover, the tortuosity increases when the porosity decreases (Table 3-6), which had also been observed in previous investigations, such as [89][115][169][170]. However, there is no strong correlation (r=-0.53), since the tortuosity is a parameter that depends on many variables, making its analysis difficult.

The analysis suggests that particles with close to a spherical shape have lower particles node densities *(n_d)* (i.e. fewer stone-stone contacts) with leads to shorter pore paths, as there is less material to obstruct the route; and thus less tortuosity [171][170].

This phenomenon can be observed by comparing specimens G1 and G2 (see Figure 3-6), whose sphericity (or Cox´s circularity) are 0.71 and 0.67, respectively (Table 3-5).

This comparison suggests that when the gradation (see grading curves for G1 and G2. Figure 3-4.a), and the porosity of the mixtures are similar (see porosity for G1 and G2 in Table 3-5), the tortuosity changed depending on the morphology of the particles.

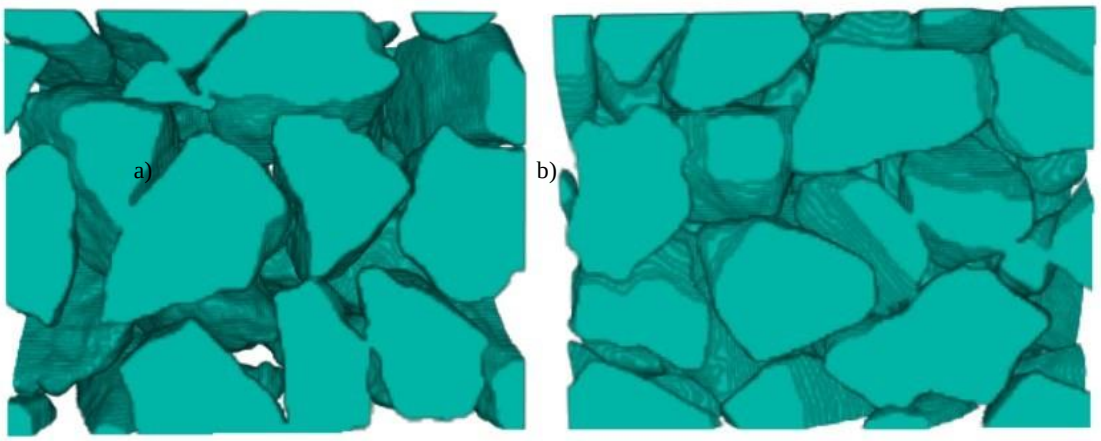


Figure 3-6 – Aggregate´s assemblies with same porosity but different tortuosity

Note: a) G1 packing´s sample, with higher stone sphericity, fewer stone contacts and lesstortuosity; b) G2 packing´s sample, with lower stone sphericity, more stone contact and more tortuosity.

Concerning the void sizes, those are directly related to the grain sizes (r = 0.87). Larger grain´s size leads to a larger void size probably due to 1) larger stones yield a larger volume of space between particles than small stones; 2) larger stones have higher compaction resistance, due to aggregate´s interlocking cause less mobility of the particles [171]. However, some exceptions are found, for exampleif we compare the RG and L1 packing samples. Nevertheless, due to the great variety of factors involved, it is difficult to establish why.

The connectivity is also directly correlated with the grain size (Table 3-6). Previous researchers found that coarser aggregates have more surface contact that leads to the formation of elongated voids (which in turn is related with the connectivity) [171]. In other words, smaller particles means larger number of particles, which in turns deal with more stone-stone contact creating like this more circular and isolated pore. Nonetheless, the global pore-connectivity (z) is within a narrow range around 3.2 for all samples (Table 3-5) which seems to indicate that, above a certain porosity, the

connectivity does not vary significantly, as was also found by previous research [89]. However, other researchers have found that there is greater connectivity in samples with bigger particle sizes, due to their more elongated voids [171], although more tests would be needed to confirm this statement.

It is also observed that the spherical particles sample are those that give a larger size of voids and as a consequence lower particle's node density, (less contact between particles).

The air void shape's properties also have a strong inverse correlation with the particle size. This could be because bigger particles tend to form more elongated voids [89]. The particle node density properties also show a strong Pearson correlation with void circularity (r=0.77). Smaller particles have a larger value of particle node density (more stones are in contact), creating more circular and isolated pore paths.

3.2.3.2 Air Void distribution

Figure 3-7 shows the graphs of void content (%) distribution with depth for all samples analysed. The height of the samples was converted to a non-dimensional height ratio [171] where 0 is the top and 1is the bottom.

For ease of presenting the data, the samples were divided into three groups which present different behaviour: group (a) single-size grain samples with a more pronounced sinusoidal distribution of air voids with depth; group (b), single-size grain samples with more constant values but random distribution and group c) various-sized grain samples.

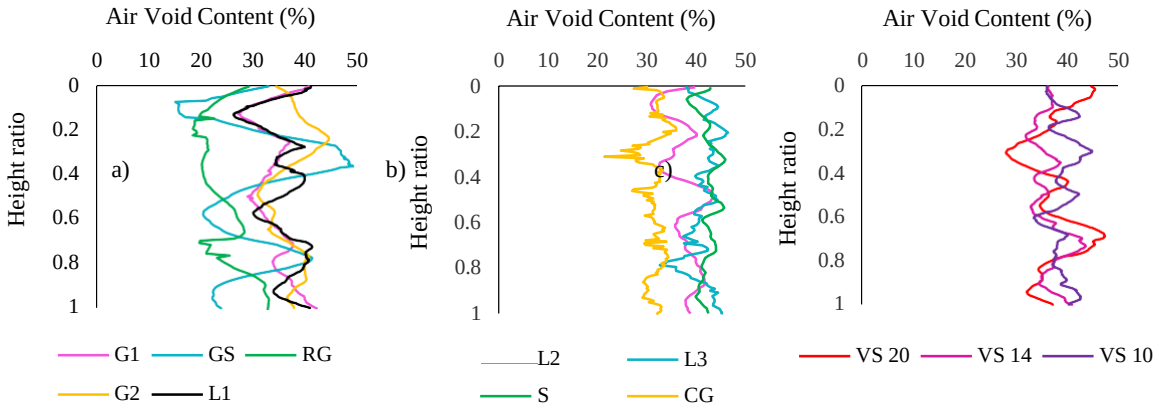


Figure 3-7 – Air Void percentage distribution along the height of the samples

As a general trend, we can see that group a) is composed by the particles which present bigger size and a more spherical form (GS, RG and the granites: G1 andG2), which lend to a more sinusoidal distribution of air voids with depth, as the porosity percentage shows greater heterogeneity throughout the sample. In contrast, group b) consists of the specimens which present smaller sizes (excepting the L1 sample), and which have greater angularity. Their Air Void content is more constant throughout the sample, but the differences present a more random distribution.

From Figure 3-7.a. it can be noticed that the highest air void content is found at the top and bottom ends of the samples, except for the sample of the uniformspheres glass (GS), which presents a damped oscillating response [172].

The sample´s packing density following an approximately sinusoidal pattern could be due to the compaction platen and base at the ends of the specimens which act to restrict possible packing arrangements, causing less mobility and limiting reorganization of the particles.

Regarding group c, the sample VS20 presents a higher degree of heterogeneity in the porosity distribution, having a peak value towards the bottom that reaches a porosity

of 28%, while towards the top of the sample, another peak reaches 47%. This is probably due to segregation produced by the vibration to which the samples were subjected, which caused the smallest particles to creep towards the bottom and, therefore, to fill the gaps, as it was found also in [173]. Samples VS14 and VS10, present a lower heterogeneity in comparison with VS20. The segregation phenomenon is analyzed in the following section in more detail.

3.2.3.3 Particle segregation and particle orientation for the various-sized aggregate specimens

Particle's segregation

The ratio of average particle´s size between sections, S_L, is presented in Table 3-7, from it can be seen that vertical segregation presents higher values than radial segregation.

Table 3-7 – S_L Values for Aggregate's Vertical and Radial Segregations

Specimen	Vertical Segregation			Radial Segregation		
	Average Minor Feret (mm)		S_L	Average Minor Feret (mm)		S_L
	Top	Bottom		Outer	Inner	
VS20	5.6	4.35	28.74	3.7	4.4	-15.91
VS14	4.25	3.41	24.63	3.98	4.2	-5.23
VS10	2.74	3.2	-14.38	2.69	2.81	-4.27

The samples: VS20 and VS14 have a positive S_L value, indicating larger coarse aggregates in the top region. This could be due to a larger difference between the stone´s sizes, which makes small particles migrate between the larger ones, finding their way towards the bottom of the container, thanks to the vibration.

Conversely, the VS10 specimen presents less segregation, which might be because of it is only comprised of two parent materials, not 4. Therefore, there is less opportunity for segregation. A negative S_L indicates larger stones settle to the bottom,

probably due to the gravity, whereas the smaller particles have gone upwards due to the vibrational effect, as was found in previous research [161].

Regarding the radial segregation, all samples have a negative S_L value, indicating a larger number of small stones at the edges than in the central part of the core. This is the well-known edge effect, or mould confinement effect [161], where large particles cannot position themselves randomly due to the mould constraint so that they leave spaces that are preferentially filled by small particles. Nevertheless, the results indicate that this effect was not very pronounced, being the bigger segregation in the VS20 sample due to its bigger difference between its particle sizes which lead to greater availability of small particles to fill the voids between larger ones.

Details about the aggregate distribution in the vertical direction for the three specimens are plotted in Figure A-8 in Appendix A.2.1.

Particle's orientations

the average orientation angle, $\bar{\varepsilon}^{o}$ for the various-sizes samples is summarized in Table 3-8,which shows an $\bar{\varepsilon}^{o}$ value close to 50^{o} for all specimens, indicating random particles orientations throughout the specimens. However. Towards the bottom, the $\bar{\varepsilon}^{o}$ values are higher than at the top. This might be due to the first particles filling the edges of the container tending to a circumferential alignment.

The same phenomenon is observed in the radial direction, where $\bar{\varepsilon}^{o}$ is greater in the outer zone than in the inner zone because of mould confinement.

Additionally, the gradient of the distribution of angular orientations as a function of the size of the aggregates has been calculated for the radial direction since it is where a bigger difference between angles is found. The results are shown in Appendix A.2.1.

Table 3-8 – Average Orientation Angle of Aggregates, $\bar{\varepsilon}^{o}$

	Vertical direction			Radial direction	
	Average Angle of Aggregates, $\bar{\varepsilon}^{o}$				
Specimen	Top	Bottom	Throughout specimen	Outer	Inner
VS20	48.7	53.95	51,03	56.34	46.07
VS14	53.74	55.05	53.40	56.8	47.2
VS10	50.35	53.52	52.07	53.69	48.93

3.3 Preparation and Air Void measure of real asphalt specimens

This section explains the experimental results comprising the manufacture of asphalt mixtures using gyratory compaction and their Air Void content analysis.

The output of this analysis will serve to test the suitability of using a Physics Engine to predict the Air Void content of porous hot-mix asphalts, through the simulation of the compaction of their aggregate´s structures, based on the volumetric properties of the mixtures, in a similar way that the Superpave Gyratory Compactor (SGC) does (Chapter 6).

The output of this analysis will serve to test the suitability of using an impulse-based approach through a Physics Engine (see Section 2.4) to predict the Air Voids content of porous hot-mix asphalts, through the simulation of the compaction of their aggregate´s structures, based on the volumetric properties of the mixtures, in a similar way that the Superpave Gyratory Compactor (SGC) does (Chapter 6).

To ensure the reliability of the results obtained from the back-calculationof the SGC, those have also been verified by testing the same manufactured specimenswith the sealed specimen's procedure in the laboratory, with good agreement.

3.3.1 Asphalt mixture specimens design

Eight different porous asphalt mixtures were designed for this investigation by blending, in different proportions, the crushed limestone materials presented in Section 3.1.1, labelled: L2, L3, L4 and L5. These materials come from the same source and have the same density and their properties can be seen in Table 3-1. The objective of using the same type of limestone is to have the same aggregate properties with limited influence of abrasion and angularity during the compaction process.

Porous asphalt is predominantly composed of coarse aggregates. The fine fraction (or dust) plus bitumen is used to provide a mastic that will reduce the voids between the coarse aggregate while limiting the bitumen content needed to ensure an adhering mix. Provided the amount of mastic is not too high, this is achieved without interfering with the aggregates' skeleton [174]. Thus, when simulating a porous asphalt, the mastic can be treated as though it is part of the void space. It doesn't affect the coarse aggregate skeleton so that, from the perspective of a coarse aggregate packing simulation, it can be ignored.

In general, porous asphalt designs use as a starting point the combination of coarse aggregates in specific percentage so that, their matrix reaches the maximum density since it is the one that provides greater stability by having more stone-stone contact and therefore, an adequate strength as well as a minimum permanent deformation. Then the fine fraction is added to achieve the concept of designing "target porosity" or "target density" and the optimums bitumen content.

In this study, the volumetric proportion of the different stone´s sizes and the binder content of each mixture´s design used to achieve their target density (ρ_{Target}), was selected based on the research team experience, with mixtures used in the Nottingham Transportation Engineering Centre (NTEC) laboratory. These combinations included variation in: maximum aggregate size (6, 10, 14 and 20mm), gradation (the granulometric composition of each open-graded mixtures was chosen to have a variety of curves between the low and high limits, including those provided by the (BS EN 13108) specification) and percentage of fines.

The range of different asphalt mixtures designed will be used to validate the asphalt air content prediction model presented in Chapter 6 with a wide range of aggregate structures. A summary of the mix design is presented in Table 3-9.

Sieve No. 4 (4.75mm)was used to sort the aggregates into coarse and fine (dust) groups, as indicated by the British Standard, BS EN 13043:2013 [14],which are the specifications used in the Nottingham Transportation Engineering Centre (NTEC). The full aggregate gradation of these porous asphalts can be seen in Appendix A.3.1.

For ease of handling data, it was decided to name the asphalt samples as follows: Mix 1 and Mix 2 to the specimens with a maximum size of 6mm and target Air Void of 10% and 21% respectively; Mix 3 and Mix 4 to the specimens with a maximum size of 10mm and target Air Void of 10% and 21% respectively; Mix 5 and Mix 6 to the specimens with a maximum size of 14mm and target Air Void of 10% and 26% respectively, and Mix 7 and Mix 8 to the specimen with a maximum size of 20mm and target Air Void of 5% and 20% respectively.

Table 3-9 – Summary of the asphalt mixtures designs

	Mix 1		Mix 2		Mix 3		Mix 4		Mix 5		Mix 6		Mix 7		Mix 8	
								mass								
Aggregate size	%	gr	%	gr	%	gr	%	gr	%	gr	%	gr	%	gr	%	gr
20 (mm)	0	0	0	0	0	0	0	0	0	0	0	0	20	213	45	401
14 (mm)	0	0	0	0	0	0	0	0	60	601	90	765	20	213	5	45
10 (mm)	0	0	0	0	75	745	91	819	15	150	0	0	15	160	10	89
6 (mm)	67	657	85	745	7	70	0	0	5	50	5	43	15	160	25	223
Dust (4mm)	33	324	15	131	18	179	9	81	20	200	5	43	30	321	15	134
Binder Content (%)	6	63	5.3	49	5.5	58	4.3	40	5	53	4	35	4.7	53	5	47
Total mass (gr)	1044		926		1052		941		1055		885		1119		939	
ρ_{mc} *(gr/cm³)*	2.461		2.486		2.48		2.526		2.487		2.536		2.507		2.491	
ρ_{Target} *(gr/cm³)*	2.215		1.964		2.232		1.996		2.238		1.877		2.375		1.993	
Target Porosity (%)	10		21		10		21		10		26		5		20	

3.3.2 Mixing and compaction

Six replicates of each porous asphalt mixture´s design were compacted using the Superpave Gyratory Compactor from the laboratory of Nottingham Transportation Engineering Centre (NTEC) of the University of Nottingham. The replicates were employed to ensure the reliability of the measurements; they were used to detect variations that could have been caused by differences in manual manufacturing.

As mentioned, the specimens were prepared by mixing limestone aggregates coming from the same source, but they differ in gradations and maximum aggregate sizes (6, 10, 14 and 20 mm). For all the mixtures, the same bitumen 60/40 penetration grade was used. The maximum density used in this study corresponds to the theoretical maximum density (ρ_{mc}), determined by the mathematical procedure specified in BS EN 12697-5 [175], as has been done in previous research from the same department [176]. The weight of the samples was calculated based on that maximum density and the volume of the cylindrical mould (100 mm diameter × 60 mm tall), according to the

formula specified in BS EN 12697-3 [177]. The total mass and the maximum density (ρ_{mc}) of each mixture are summarized in Table 3-9.

Before the mixing process, the aggregates were dried by heating them overnight in an oven at 105°C, to ensure all moisture is removed. Then, the oven-dried stones were placed into the mixer device and blended without bitumen for 30 seconds to ensure uniformity and homogeneity. Subsequently, the exact amount of bitumen required for each specimen was poured into the mixer with the aggregates and the mixtures blended until every stone was uniformly coated with bitumen. Then the bitumen-aggregate mixtures were fed through a funnel into the cylindrical mould before being placed in the gyratory compactor at 140 °C. The cylindrical moulds were coated with glycerides soda oleate according to the indications in BS EN 12697-3 [177].

The gyratory machine was set at a vertical pressure of 0.6 MPa and an external angle of gyration of 1.25°. The compaction process was standardized at 350 gyrations to maintain the same condition for all the specimens.

The appropriate number of gyrations was reached from several laboratory trials to determine the 'locking point'. The locking point is the moment when the aggregate structure becomes stable during the asphalt mixture compaction; and it is achieved when the sample´s height remains constant over three measurements (definition of Georgia Department of Transportation [178]). This was achieved at 350 gyrations in all the samples tested.

It was verified that Mixture 7 was the one that required the largest number of gyrations (290) to reach the locking point, as can be seen in Figure 3-9. This Figure also shows that, for the densest samples (Mix 1, Mix 3, Mix 5 and Mix 7), the number of gyrations increases as the maximum aggregate size increases; while for the less dense

samples (Mix 2, Mix 4, Mix 6 and Mix 8) the number of gyrations remains close to values 90 and 100.

Finally, the cylindrical cores were cooled at room temperature overnight and extruded for further testing.

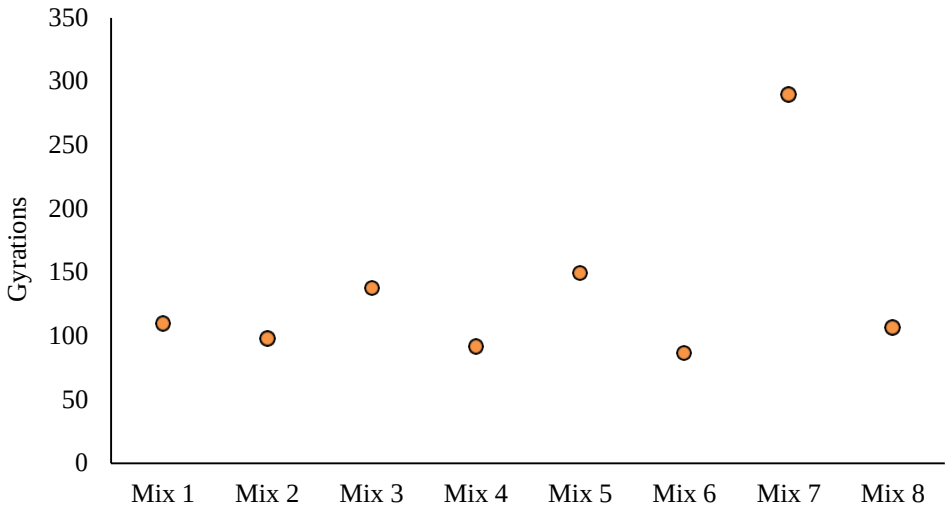


Figure 3-9 – Summary of the gyratory locking points

3.3.3 Measurement of bulk density and Air Void content

The mean values of the bulk densities (ρ_b) and the air voids content (AV) for each set of specimens were calculated from two methods: the sealed specimens(analysed in the laboratory) and the output data from the gyratory compactor (ρ_{bsea}, ρ_{bgyro} and AV_{sea}, AV_{gyro} respectively). Both procedures were carried out in accordancewith BS EN 12697 8:2018 and BS EN 12697-6:2012 [179] (See Appendix A.3.2).

This decision was made to determine if there is a significant difference in the air void content calculated from both methods. Uncertainty is undesirable because of the need to compare the results with data from the SGC as presented in Chapter 6.

The Bulk density and the Air Void content measurement for both procedures were carried out in accordance with BS EN 12697 8:2018 and BS EN 12697-6:2012 [179]. (See Appendix A.3.2).

3.3.4 Results

The results of the mean values of the measured bulk densities (ρ_b) and the Air Voids content (AV) for each set of specimens from both methods explained above: the sealed specimens analysed in the laboratory and the output data from the gyratory compactor (ρ_{bsea}, ρ_{bgyro} and AV_{sea}, AV_{gyro} respectively); are summarized in Table 3-10.

Additionally, the standard deviation of the Air Voids contents has also been calculated. The standard deviation of the Air VoidS content is used to judge the variability of both techniques. Detailed information for each of the 42 core mixtures is provided in Appendix A.3.3.

The mean heights for each specimen were also extracted from the output of the SGC (Table 3-10).

Table 3-10 – Results from the sealed specimen´s laboratory test and from the gyratory compactor´s calculation

Specimen	Mean ρ_{bsea} (g/cm^3)	Mean AV_{sea} (%)	Standard Deviation AV_{sea}	Mean ρ_{bgyro} (g/cm^3)	Mean AV_{gyro} (%)	Standard Deviation AV_{gyro}	Height (mm)
Mix 1	2.155	12.29	0.91	2.158	12.4	0.95	6.17
Mix 2	1.966	20.88	0.52	1.976	20.48	0.84	5.96
Mix 3	2.167	12.61	0.64	2.138	13.75	1.54	6.22
Mix 4	1.961	22.35	1.57	1.950	22.78	1.58	6.1
Mix 5	2.160	13.11	0.99	2.167	12.86	0.83	6.19
Mix 6	1.928	23.9	0.89	1.889	25.5	1.2	5.93
Mix 7	2.322	7.36	1.3	2.341	7.52	1.04	6.07
Mix 8	2.025	18.7	1.11	2.013	18.9	1.49	5.91

From Table 3-10 it can be seen that the two methods produce similar air voids content values for most of the samples, with less than 1% difference. However, this difference is larger for the specimens: Mix 3 and Mix 6.

The data indicate a general tendency of a higher air void values with the dimensional method than the sealed specimens. This could be due to the SGC being prone to an underestimation of the bulk specific gravity leading to an overestimation of the air voids of the compacted specimens, due to the surface irregularities that the porous asphalts have. For the sealed samples, this phenomenon is less pronounced.

The same trend is observed for the standard deviation which, on average, is higher for the results obtained from the SGC. In general, the variation in the porosity values obtained by the sealed specimens is between 1% and 3%; while the variation of the air void content obtained with the SGC is between 2% and 4%.

Further, the samples: Mix 1 and Mix 2, with a maximum aggregate size of 6mm, present lower standard deviation than the samples: Mix 7 and 8 samples (with a maximum aggregate size of 20mm) since surely, coarser aggregates produce more irregular surfaces and therefore more variation in measurements; although it should keep in mind that those variations in the calculation could be also due to differences in thefinal masses and aggregate and bitumen content of the different specimens.

The densest mixtures seem to be not affected by the over compaction, since all of them present a porosity higher than the established target as well as higher sample height.

Mixtures with coarser aggregates and higher density design criteria are more susceptible to over-compaction since they present porosity values lower than the target

one. The reason could be the breakage of the large aggregates during the compaction process, as demonstrated by previous researchers, as can be found in [41][178][180].

3.4 Summary of the experimental program

3.4.1 Particle characterization

In the first section of this chapter, an image-based analysis has been carried out to characterize the size and shape characteristics of coarse aggregate samples so as to obtain greater precision and objectivity than can be obtained by the manual methods, which are also more time-consuming, as it was described in Section 2.2 of the literature review.

The software ImageJ is used to extract the morphological information of the stones in two dimensions, focusing on those properties that are common when it comes to characterizing aggregates quantitatively [73].

This study's output will be used to create and validate the virtual aggregate generation model described in Chapter 4.

3.4.2 Preparation and analysis of the real aggregates packing specimen

The results from this study indicate that the particle morphology strongly affects the structure they form; which is in agreement with some general concepts such as:

1. The porosity is highly influenced by the size and the angularity of the grains (smaller sized stones generate greater porosity [164][87]; particles with higher angularity leads to higher porosity [166][167]).
2. Void size strongly correlates with the particle size, increasing with larger aggregates [171] [89]. Nevertheless, this factor is also influenced by the

stones' size ratio, were samples which bigger differences between particle sizes presented smaller másimum voids due to the small particles filling the gap betweenthe bigger stones.

3. Tortuosity and porosity are inversely correlated [89][115][169] [170].
4. A uniform gradation gives bigger void sizes and less tortuosity, due to more resistance to compaction and less particles rearangment (there will be fewer small particles to fill the voids)[115] [168].
5. Our results suggest that particles closer to a spherical shape, usually have less stone contact, and therefore shorter and less tortuous pore channels (less materials which obstruct the path) [170] [171]. However, more tests are necessary to corroborate this theory.
6. Bigger pore connectivity values were found for samples with bigger particle sizes due to the more elongated void channel they form [109].
7. It was also found that the grains' size is related to the connectivity.

   This could be due to coarser particles tending to give more elongated voids and thus have greater connectivity. Nevertheless, more tests areneeded to corroborate this hypothesis.

Despite all the analyses carried out, it is still unclear how the different properties of the stones and their sizes distribution affect the assemblies' internal structures due to the large number of factors involved. For that reason, new simulation methods on this topic are necessary to allow further study on this topic and to create appropriate tools, able to find the optimized blend coarse aggregate without the need to carry out several

trial tests in the laboratories, to which granular materials' designs are currently subjected.

3.4.3 Preparation and Air Void measure of real asphalt specimens

This section presented the laboratory procedure for the design and manufacture of different porous asphalts using the Superpave Gyratory Compactor (with 350 gyrations in order to be sure to reach the material's locking point); in order to measure their bulk density and air void content by two different methods: the data output generated by the SGC and the laboratory procedure using the sealed samples techniques, according to the BS EN 12697-6:2012. The specimens were prepared using the same type of limestone to avoid variation due to the influence of different abrasion behavior and angularity during the compaction process.

From the study of the looking point of the different mixtures, it was found that the sample with the lowest target porosity was the one that needed the greatest number of gyros to reach the locking point.

From the analysis of the bulk density and air void content, the results obtained ensure the reliability of the density measurements made by the SGC, since it provides very similar values to those obtained through sealed specimen's test.

The air void contents calculated by the SGC for each set of specimens tested in this study present a variation between 2% and 4%. Further, the samples with coarser aggregate are the ones with highest standard deviation, probably due to their more irregular boundary surfaces, and this leads to more variation in the results.

CHAPTER 4. A 3D AGGREGATE´S GENERATOR MODEL WITH REALISTIC SHAPES

Automatic realistic aggregate generation is an important requirement to reconstruct 3D meso-structures that can represent different stone-based materials, due to the need for a better understanding of how the different characteristics of the particles affect the structure they form.

A technique based on a free-shape volume generation which are visualised through avoxel-based approach (the Marching Cube algorithm), is implemented in a Physic Engine, (the PhysX engine in Unity 3D platform), for the creation of different stone samples with user-defined characteristics in real time, with a decent consumer-grade hardware

This technique can be divided into three major procedures:

(i) First, several shape´s parameters and the size of the particles were characterised statistically in 2D, following the methodology based on an Image-based analysis presented in Section 3.1.1; in order to use this information as inputs of the virtual aggregate generation model, as well as for its validation.

(ii) Second, a scalar function which combines a spherical primitive with random noise is defined to design stone´s shapes of appropriate angularity through the adjustment of several parameters which control the level of noise added to the initial sphere.

(iii) Finally, the virtual particles created are rescaled and deformed until they follow realistic size and shape distributions; and certain morphological properties are equivalent to those wanted.

The efficiency of the model proposed has been tested by reproducing eleven types of aggregates from different sources. The results indicate that the generated irregular-shaped particles can reproduce the real coarse aggregates samples in terms of, particle size distribution and morphology.

4.1 Introduction

The first step to create computer simulations of granular materials, is to reproduce realistic grains virtually. The building of a realistic geometry of the particles are normally based on imaging physical specimens, by using different acquisition systems as can be seen in [13] and [101]. In spite of they can re-create grain shape features with great precision, they have certain disadvantages that cannot be overlooked: they are expensive and not available to everyone (due to the device needed for the particles scan); they require a high computational effort when entering the particles in the granular material software simulation, (since they manage a great number of data); and they are unsuitable to build virtual particle´s samples in an efficient way (as they only allow the particle´s modelling one by one or just few at the same time).

It is necessary to create computer models able to generate particles in a wide variety of shapes and sizes, in order to simulate full coarse aggregate´s samples of stone-base mixtures, in a more streamlined way than imaging real specimen's methods, such as the user-defined techniques.

These techniques refer to those virtual particles which are modelled directlythrough computer programs, which allow the user to control some morphological parameters, as described in Section 2.5.1. In this way, the aggregate samples do not needto be scanned in the laboratory and therefore, the process is easier and more efficient thanthe individual particle imaging process. Additionally, they have lower computational cost when implementing the particles to simulate stone-based materials, as the aggregates are simplified to polyhedral shapes. Some research using these techniques to simulate coarse aggregate gradation used in asphalt, ballast or granular soils can be read in [6], [103], [9] respectively.

The user-defined techniques use semi-stochastic algorithms which can create several 3D random aggregates with different sizes and shapes in a fast way. Nevertheless, generating realistic randomized 3D particles can be computational challenging, due to the lack of mathematical models to represent irregular shapes [181].

Their main drawback is that particles are oversimplified to convex polyhedrons in which, neither the different levels of angularity nor the concavities that the aggregates used in the stone-based material might have, are taken into account. Further, most of the works found in the literature toward this topic (simplify the sieve size of the particles to the equivalent diameter of a sphere [101] and they do not reproduce the real size distribution or shapes of the particles in a real sample, since the aggregates are normally picked randomly from the database [68].

For that reason, the objective of this chapter is to develop a novel user-defined technique to create realistic shapes (including concavities). The algorithm consists of a spherical density function and a Perlin noise to deform the spherical density function; under the purpose to create a cloud of points that re-creates the shape of the aggregates in

a realistic way. Then, the points are triangulated by using the Marching Cube algorithm (see Appendix B.1). Finally, the particles are deformed until they match the aspect ratio and minor Feret of the aggregates that we want to simulate. The model proposed is able produce a wide variety of stone´s samples which follow realistic size and shape distributions, by considering the statistical parameters of real aggregate´s samples extracted from a 2D Image-based analysis.

4.2 Experimental measurement of aggregates' morphological properties

The laboratory procedure to extract the morphological information of the real particle´s samples have been explained in detail is in Section 3.1; below a summary of the work done is presented.

Several particles' samples were analysed using the Image analysis software ImageJ; in order to mathematically characterize them through different 2D shape´s factors which are: Minor Feret (MF), Aspect Ratio (AR), Roundness (R), Area (A), Perimeter (P) and circularity (C). Those factors are explained in detail in Section 2.2.6 and summarized in Table 3-1.

In addition, the constant β, which represent the ratio between the thickness and the mayor Feret diameter (Equation 3-4) were also drawn, as it will be used to adjust the third dimension (which cannot be measured through the Image-based technique), in our virtual aggregate generation model.

A total of eleven coarse aggregate´s samples were analysed: Glass Spheres (GS); Crushed Glass (CG); Round Gravel (RG); two types of granites, both having14mm maximum size (G1 & G2) and two types of Limestone. One of the limestone was given

the symbol L1, while the other was available at four sizes so is given four symbols – L2, L3, L4 and L5, with maximum size 6, 10, 14 and 20mm respectively.

The Minor Feret and the Aspect Ratio are the shape´s factors chosen to be implemented in the virtual aggregate model, on the basis of the hypothesis that they represent the gradation and the general form respectively of each type of particle respectively [182][183]. For that reason, their distributions curve was characterised by the Weibull statistics parameters; which was the distribution model more accurate for all the samples (see Appendix .A.1.2).

The morphological information of the aggregate types has been quantified in terms of the 50th percentile, (P_{er50}), for a later comparison with the computed results.

The Weibull distribution parameters obtained as well as the P_{er50} results were plotted in Table 3-2.

4.3 3D aggregate generation model

In the long run, the aim of this thesis is to develop an algorithm able to perform particle´s assembly systems (Chapter 5 and Chapter 4). For this reason, the particle generation model is created in the free license Unity 3D platform, which use the PE PhysX engine, since it offers a good balance between accuracy and computational effortthanks to its integrator solver and its capability of detecting contacts between complex particles, as explained in Section 2.4.4.

Therefore, both simulations: the particles one and their packing, will be performed on a single platform, offering thus a comfortable, easy and fluid process when modelling, analyzing and rendering.

The random aggregate generation is based on the use of scalar functions to design free-form surfaces, which is visualized thanks to a voxel-based approach (see Appendix B.1), specifically the Marching Cube algorithm [67], which is explained in detail in Appendix B.1.1.

This model can be divided into two major procedures:

1) Creating a scalar function which mixes a spherical density function and a Perlin noise to deform the spherical density function; to represent and manipulate irregularly shaped particles with different levels of angularity.

2) Achieve realistic morphological features by implementing Weibull distribution parameters of certain properties in the algorithm, and taking into account the mathematical characterizations of the real samples through different shape's factors.

Finally, the real and virtual 50^{th} Percentile of all the shapes factors studied are compared for the model validation.

The flow diagram drawn in the figure shows the process of generating aggregates, which will be explained in detail in the following sections.

1) Creation of irregular particles	2) Adjust particle´s size and shape	3) Optimization of the parameters of noise

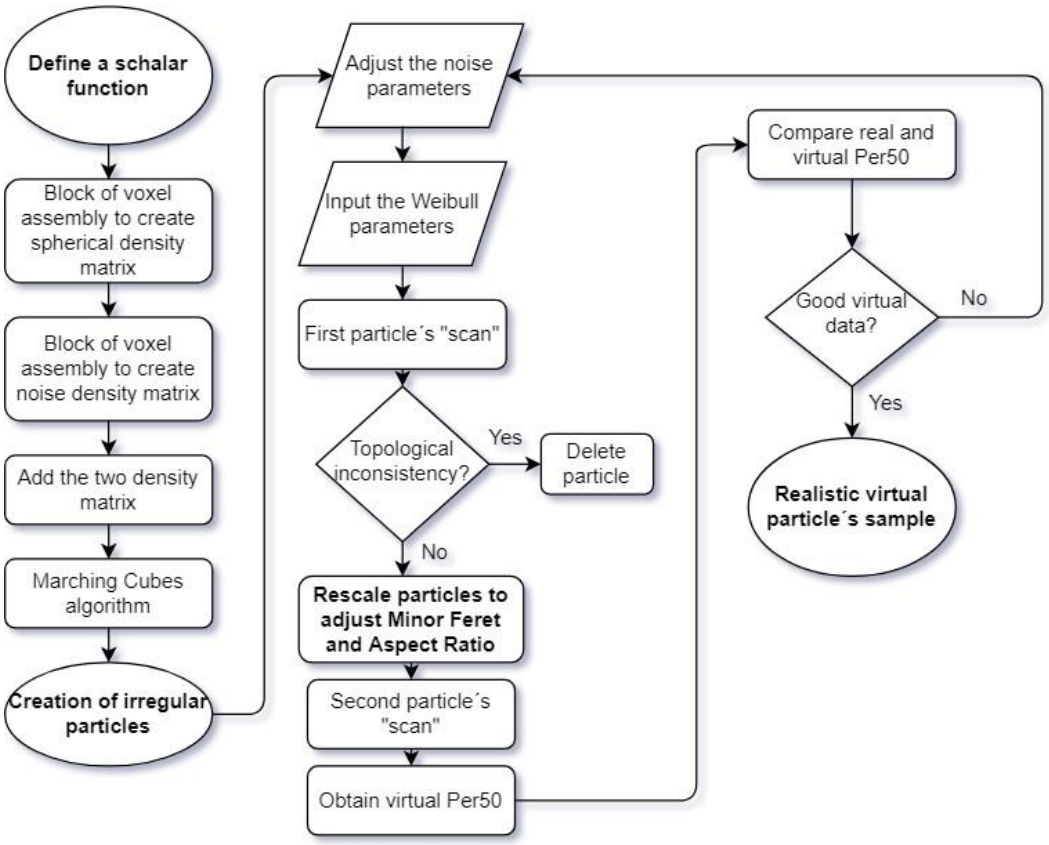


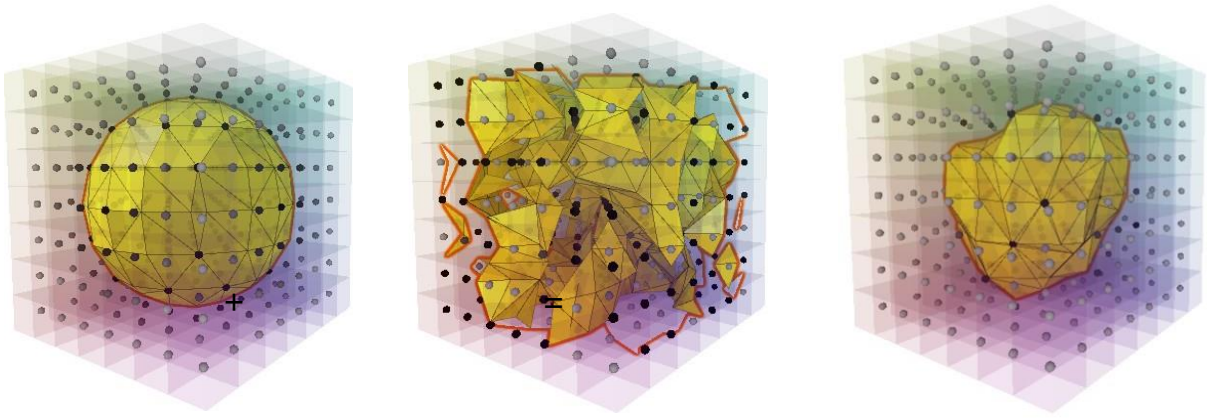
Figure 4-1 – Flow diagram of the virtual particle's generation process

4.3.1 Creation of irregularly shaped particles with different level of angularity

This section presents a semi-random algorithm which use scalar functions to describe arbitrary shapes which are then visualized through a polyhedrization process thanks to the Marching Cube algorithm (MC).

To re-create irregular particle´s shape (including concavities), the method consists on the mixture of a spherical function with a Perlin noise function, which deformsthe spherical primitive until its roundness and circularity becomes like that of an

aggregate, based on the Krumbein and Sloss chart classification (see Figure 2-10). Figure 4-2 summarises the process.



a) Spherical density matrix b) Noise density matrix c) Irregular particle

Figure 4-2 – Irregular particle´s creation chart

Firstly, a spherical primitive with radius *r*, is inscribed in a grid of *N* x *N* x *N* voxels, see Figure 4-2.a, where *N* is the number of voxels and *d*, the distance between the nodes of the voxels. The radius of the sphere is defined by the following equation:

$$r = d \times \frac{N}{2} \qquad 4\text{-}1$$

An increase in the number of voxels, *N*, implies an improvement of the resolution of the 3D model but has the inconvenience of a higher computational cost. In this paper, a grid of 7 x 7 x 7 voxels with 1 unit distance between them has been chosen by trial and error, since it has been found that this permits a good resolution while avoiding large memory requirements [184].

Then, a 3D density function (x, y, z) that determines which lattice nodes, with coordinates: x, y, z, are inside or outside the sphere is defined. In this case, the nodes outside the sphere are positive, the nodes inside the sphere are negative and, the nodes at

the surface has a value of 0; as explained in Appendix .B.1.1. The formula that defines the values of the density function is Equation 4-2. In addition, a weight parameter (S_c), has been added that increases the density values of the spherical functions. This factor is used to control the influence of the spherical density function in the final shape of the aggregates. In other words, it controls the sphericity of the virtual aggregates. As the number of voxels is 7, and the distance between them 1, the radius, from the centre of the voxel, is 3 and the density function values inside the sphere range between 0 and 3.

$$f(x, y, z) = (-r + v) * S_c \qquad 4\text{-}2$$

Where,

r is the radius of the sphere defined by Eq.4-1.

v is the distance from the voxel's centre to the centre of the sphere.

S_c is the Sphericity Control, the weight parameter which controls the influence of the spherical density function in the final shape of the aggregates.

In order to introduce dents in the particles that are like those in crushed aggregates, a second grid with 7 x 7 x 7 voxels with node´s values from a Perlin noise generator (P_v), is generated and combined with the spherical one, see Figure 4-2.b. The coordinate x, y, z, ($(x, y, z) = P_v$), of the second grid have values which may range between 0 and a maximum value of 4, depending on the degree of deformation desired. The highest value of the Perlin's was labelled as noise range, *f*. The advantage of using a Perlin noise is that is fast and simple as well as smooth and spatially coherent, achieving like this natural looking surface [185].

If the value is 0, there is no deformation. If the value is 4, the deformation is the

highest. This value has been determined by the trial-and-error method and, a higher value could produce unreasonable shapes (see Figure 4-3).

Each aggregate has a different matrix, which makes this a stochastic algorithm. To control the angularity of the aggregates, P_v can be multiplied by a factor that we have named Intensity of deformation, I_{def}, which weights P_v in order to refine and improve the final mesh. Nonetheless, it is necessary to mention that the concavities that the real stones samples present have not been calibrated, and the virtual stone´s concavities obtained through this method are completely random (like those concavities that occur when stones are subjected to a crushing process).

The final density function of the aggregates, $F(x, z, y)$, is created by adding the spherical density function and the noise density function as follows, see also Figure 4-2.c.

$$F(x,z,y) = f(x,y,z) + R(x,y,z) = [(-r + v) * S_c] * [P_v * I_{def}] \qquad 4\text{-}3$$

Finally, the Marching Cube algorithm is used to triangulate the shape of the virtual aggregate. The points in the voxels' edges that cross the density values equal to 0 are identified by the Marching Cube and triangles are generated that transverse themand form the shape of the virtual aggregate (see Appendix B.1.1).

In some occasions, the method presented above may produce virtual aggregates with topological inconsistencies, such as holes in the mesh, as shown in Figure 4-3. In this research, it could happen for two reasons: (i) due to the intrinsic nature of the Marching Cube algorithm, which in occasions may interpret that some faces correspond to holes
[67] and (ii) the random density matrix which can deform the particles excessively. These ambiguities occur when there is more than one possible way to triangulate the points at

the voxels' corners and these can be interpreted as holes

Such virtual stones are discarded by a particle control routine that uses the Raycast operation from Unity 3D [186]. This operation "sends a virtual laser ray" that can detect holes surfaces. In this way the surface of aggregates is mapped in different directions and the casting rays detect possible holes depending on whether some rays going through the stone (see Figure 4-4). When any is found, the stone is discarded.

Finally, to avoid that simplistic aggregates appear, an additional variable, *R*, has been considered, which selects aggregates with a minimum number of vertices *R*, and discards them. This parameter has been set by trial and error selecting the numbers from previous research [6], and it remains the same for all the samples simulated.

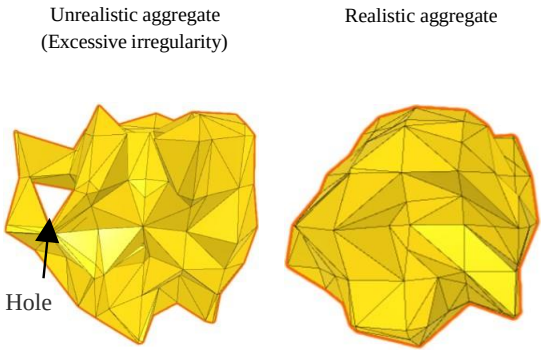


Figure 4-3 – Example of un-realistic and realistic virtual aggregates.

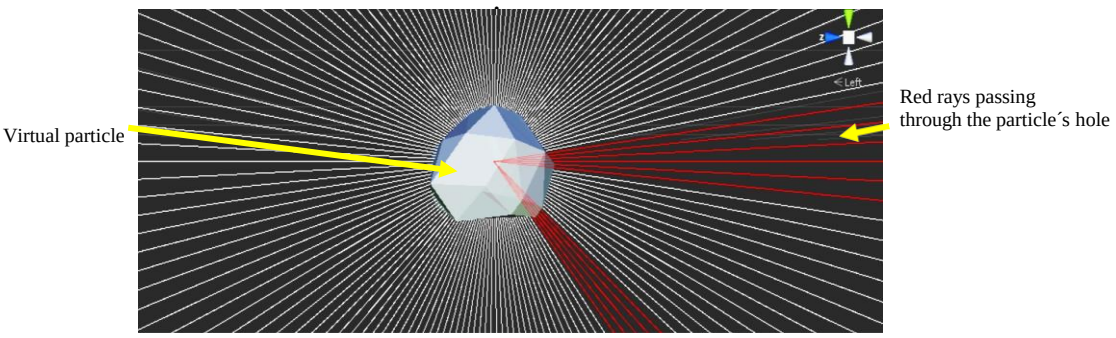


Figure 4-4 – Raycast operation to detect holes by mapping the surface of the aggregates

4.3.2 Particle rescaling to adjust their minor Feret and Aspect Ratio to that of real aggregates.

The algorithm described in the previous section produces angular particles whose minor Ferets and aspect ratios do not fit those of the actual aggregates. For that reason, a method has been created that rescales the angular particles, to achieve the desired minor Feret and aspect ratio (note that as mentioned previously, the concavities that the virtual stones presents are random and they are not controlled through the gradation and aspect ration parameters).

It is important that the virtual particle´s samples follow realistic size and shape distributions, so that they fit as closely as possible to the real one. As described in Section 3.1.1, the aspect ratio and minor Feret of each aggregate type have been characterised using the Weibull distribution function, since it was the distribution model which has the best fit over all the samples (see Appendix.A.1.2).

This distribution is defined by two parameters, one of shape, k, and another one of scale, λ, which have been measured experimentally (see Table 3-2); and are used as inputs in our algorithm by using a software library called MathNet.Numerics.Distributions [187], which, by using its Weibull distribution option, gives minor Feret and aspect ratio which fall within the same distribution of aggregates measured experimentally.

The virtual aggregates are then rescaled in order to follow the employed distribution, according to the measurements taken in their most stable position, imitating thus, the experimental 2D measurements carried out.

First, the same Raycast operation from Unity described in the previous section, is used to detect the different views of the particle (Figure 4-5.a), based on the position of the points on the particle's surface "hit" by the ray. This serves to identify the most stable

position of the virtual aggregates.

Then, an array with the coordinates of the outline plan view is created, which enables the definition of the perimeter and its calculation. From this array, the Minor Feret and the Mayor Feret are also identified and measured (Figure 4-5.b). If the minor Feret and the aspect ratio computed are not within the Weibull probability range set by the inputs, the particle is rescaled until it is adjusted and the plan view identification and measurement routine are repeated.

The area is measured based on the triangle division formed by the segments between the perimeter array and the centre of the particle´s projection (Figure 4-5.b); following the procedure described in [188].

Finally, the thickness (third dimension) of the stones is adjusted based on its ratio with the major Feret, which is the *(β)* parameters defined in Section 3.1.1, so that their thickness fits the one determined experimentally. The remaining shape´s factors (Sphericityand the Roundness) are calculated by the formulas explained in Table 3-1.

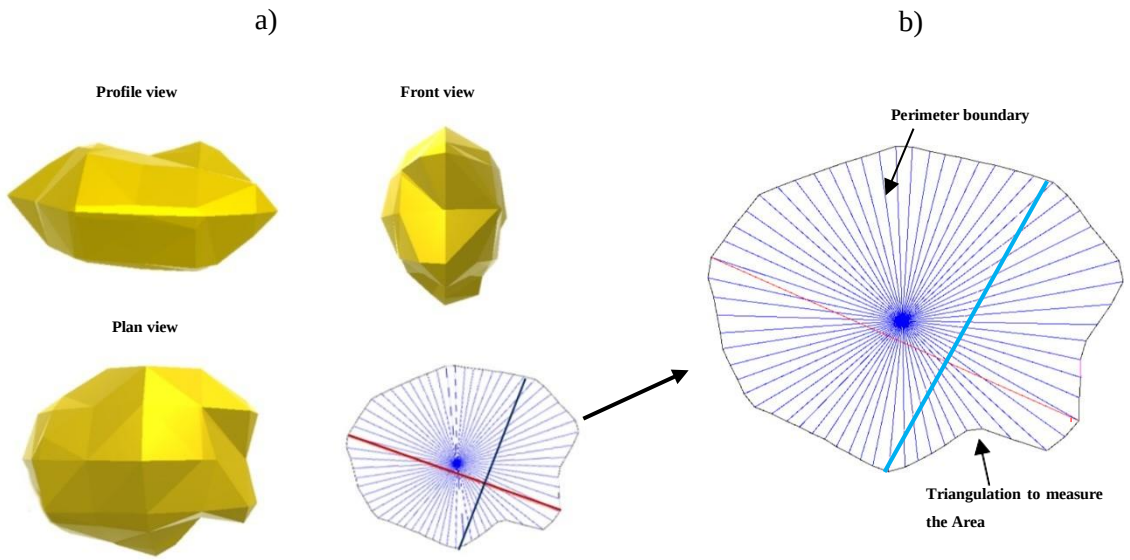


Figure 4-5 – a) The 3 views of a virtual particle and b) its measurement in its more stable position.

4.3.3 Summary of the inputs parameters which define the size and shape of virtual aggregates

A summary of all the parameters required to reproduce the aggregates can be found in Table 4-1. The input parameters which control the level of roundness and sphericity for each type of aggregate have been found by modifying the parameters of the virtual stones until they fitted the experimental results both: numerically and visually. In Appendix B.2 it can be found a video link to see how particles with different levels of angularity are generated.

In addition, since every virtual aggregate is measured individually, this method allows computing their 50th percentiles.

Table 4-1 – Input parameters to control size and shape of virtual aggregates

	Input Parameters
Roundness and sphericity	Controls sphericity, S_c
	Intensity of deformation, I_{def}
	Perlin's noise range, f
Size and shape	Minor Feret, shape factor, k_{MF}
	Minor Feret, scale factor, λ_{MF}
	Aspect Ratio, shape factor, k_{AR}
	Aspect Ratio, Scale factor, λ_{AR}
Thickness	β

4.4 Results

The eleven particles' samples described in 3.1 have been virtually recreated. In this section are presented the results of:

1) the adjustment of the inputs parameter which control the different levels of circularity and sphericity, according to the Krumbein chart classification; and

2) the comparison of the 50th percentile (P_{er50}), of all the morphological features studied in this work since every virtual aggregate is measured individually.

4.4.1 Input parameter´s setting to generate different level of angularity

The input parameters that controls the scalar density functions are: Sc, I_{def}, f (see Table 4-1); which serve to control the degree of particle deformation and contributes to achieve various levels of circularity and sphericity, as explained in Section 4.3.1.

Nevertheless, the shape cannot be deformed intuitively as those factors do not have any geometrical meaning, so their combination has been found by trial and error through a visual inspection using Krumbein and Sloss chart classification (see Figure 2-10), to generate a table of results with different level of angularity and sphericity (Figure4-6)

Figure 4-7 shows the input values adjusted to achieve a wide range of morphologies that corresponds to those commonly found in the aggregates used in civil engineering; this allows the users to choose the most convenient for their study in an easy way.

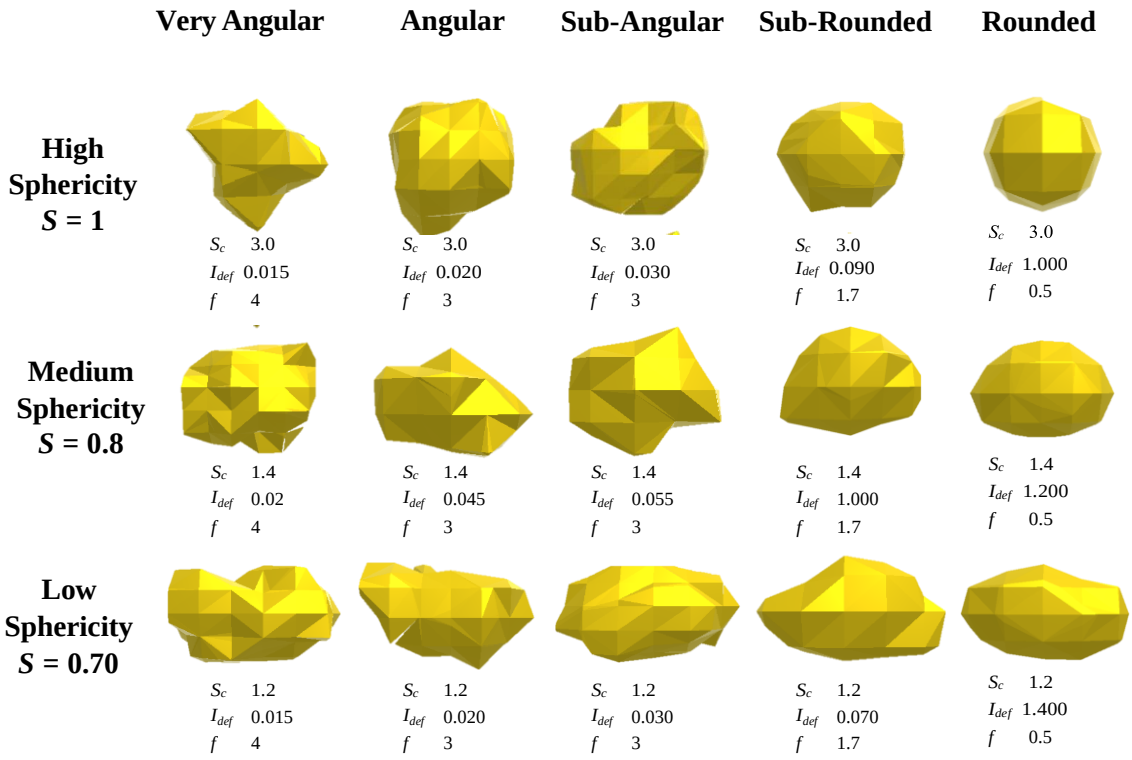


Figure 4-6 – Virtual imitation of the Krumbein and Sloss chart classification

To recreate virtually the different types of aggregates analysed in this study, these combinations of parameters have been set according to the classification of the real stone samples presented in Figure 3-2. A visual example between the real and virtual particles and their Krumbein and Sloss chart classification is plotted in Figure 4-7.

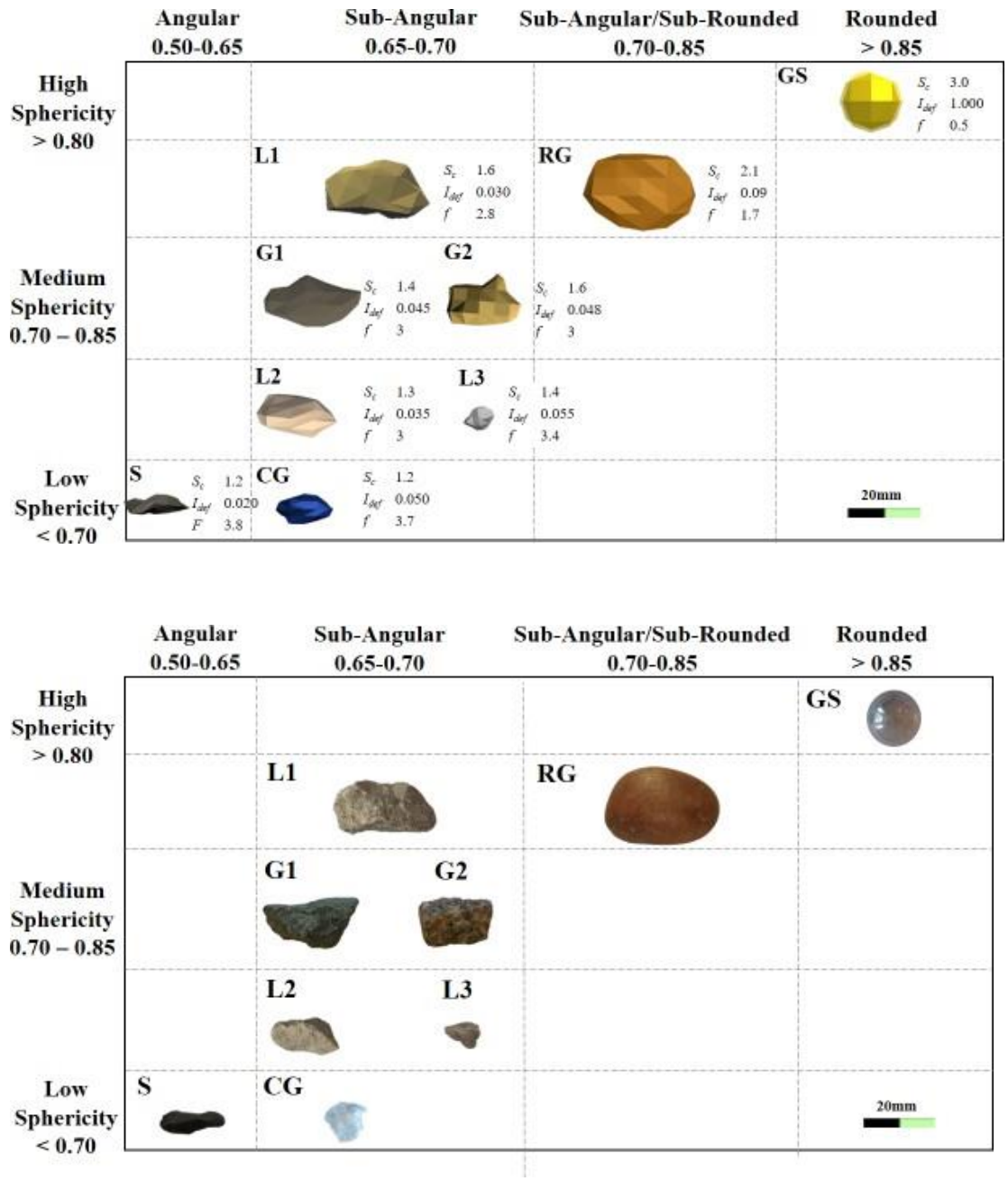


Figure 4-7 – Visual comparison between real and simulated particles

4.4.2 Comparison of the virtual data with the experimental data

The validation of the proposed model has been done by the ratio between the P_{er50} of the virtual stone´s samples with the actual one (R_{index}), in order to know the quality of fit of the virtual aggregate model with the real data measured from ImageJ. When the R_{index} value equals to one indicates a perfect adjustment (Equation 4-4).

$$R_{index} = \frac{P_{er50\ virtual}}{P_{er50\ real}} \qquad 4\text{-}4$$

Table 4-2 shows the R_{index} values obtained from the mean of the P_{er50} from 10 simulations ($\overline{x}$) and the actual P_{er50} measured from ImageJ (μ); and their percentage of error (Equation 4-5), for each stone's type. The results reveal the accuracy of the aggregate´s generation model presented.

$$\%\ error = (1 - |R_{index}|) * 100 \qquad 4\text{-}5$$

Further, $\overline{x}$ and μ have been fitted with a straight line. The statistical index R-squared (R^2) of a linear regression graphs are plotted in Figure 4-9.

Table 4-2 – R_{index} and percentage of error values resultant from comparing the simulations with experiments

	G1	G2	RG	L1	L2	L3	L4	L5	GS	CG	S
$Min\ Feret$	0.997	0.989	0.964	0.978	1.037	1.017	1.024	0.969	0.999	0.969	1.027
% error	0.335	1.052	3.644	2.187	3.739	1.683	2.366	3.115	0.073	3.123	2.703
AR_{50}	0.997	0.989	0.964	0.978	1.037	1.017	1.024	0.969	0.999	0.969	1.027
% error	0.546	1.580	0.990	0.936	1.846	2.940	0.730	1.388	0.770	2.662	2.288
P_{50}	0.831	0.879	0.895	0.854	1.165	0.830	0.854	0.845	0.976	0.809	0.806
% error	16.90	12.11	10.46	14.61	14.18	16.95	14.57	15.54	2.38	19.11	19.40
A_{50}	0.887	0.860	0.971	1.155	1.194	1.113	1.194	1.205	1.036	1.235	1.245
% error	11.35	14.01	2.93	13.45	16.26	10.19	16.21	17.02	3.48	19.03	19.67
R_{50}	1.120	1.008	1.033	1.199	1.146	1.183	1.156	1.161	0.975	1.132	1.161
% error	10.74	0.78	6.02	16.62	12.77	15.47	13.48	13.90	2.51	11.70	13.90
C_{50}	1.051	1.043	1.035	0.929	0.949	0.954	0.957	1.008	0.983	1.095	1.143
% error	8.47	9.32	6.56	5.47	12.85	11.97	4.31	8.10	1.60	14.26	9.88

The analysis reveals an almost perfect adjustment for the shape´s factors; Minor Feret diameter (Min Feret) and Aspect Ratio (AR), whose indexes indicate less than 3% difference with the actual measurements. This is also reflected in their linear regression graph, with an R^2 bigger than 0.99 (see Figure 4-8.a and b).

From Figure 4-8.c and d it can be seen that the area and the perimeter have similar R^2 close to 0.95. For both shape's factors, the virtual data differ from the real data by around 10 and 15% for most of the aggregate's samples. Nevertheless, the samples: Crushed Glass (CG) and Slag (S), exceed these percentages; while the Round Gravel sample (RG) presents the lowest error (together with the Sphere Glass (SG) sample, as expected, due to the simplicity of its shape).

It is worth to mention that the stone type Crushed Glass (CG) and Slag (S) are the ones that present more complex shapes (more angularity), while with the RG presentsmore roundness shape (see Figure 4-8.).

Roundness and Circularity are the parameters with the lowest R^2 values, as can be observed in Figure 4-8.e and f. This could be due to the formulas of Roundness and

Sphericity depends on the area and the perimeter respectively, and therefore preserve the error from the other morphological features.

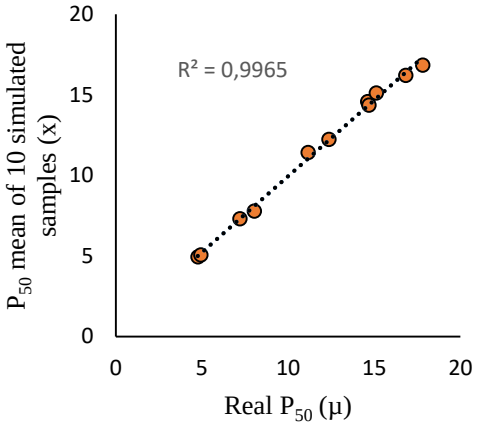
A bigger difference between the real and simulated shape´s parameters: area and perimeter, and therefore roundness and sphericity, than in the other shape´s factors could be due to:

(i) Errors induced from the selection of the variables involved in the creation of irregular particles, which were chosen by the trial-and-error method and can lead to lower angularity, and therefore less perimeter. This is confirmed with the Sphere Glass sample, which, in spite of being a primitive shape, does not show a perfect adjustment with the experimental data.

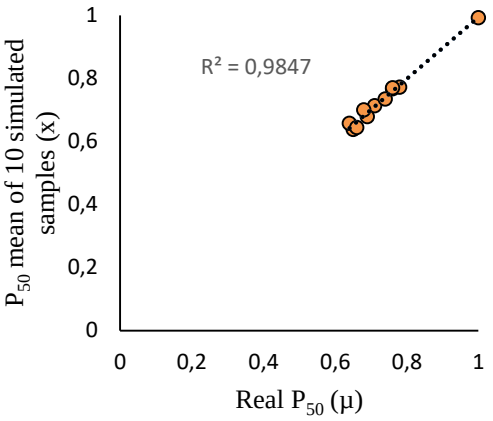
(ii) Errors induced from the measurement of the area and perimeters of the virtual particles, based to de technique described in Section 4.3.2.

(iii) Errors induced from the measurements of area and perimeter of the real particles by ImageJ, which may be influenced by shadowing and thresholding

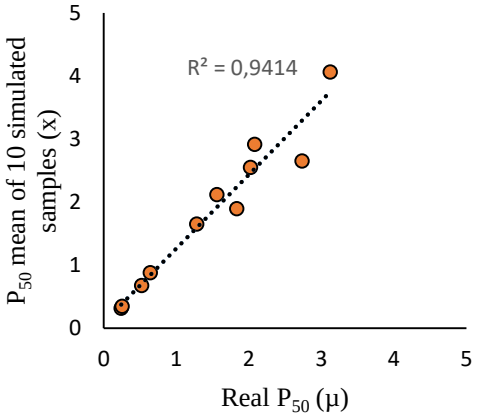
a) Minor Feret



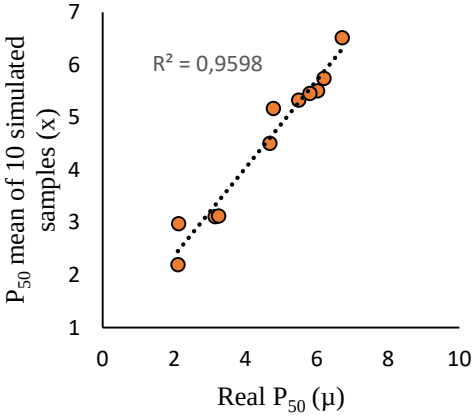
b) Aspect Ratio



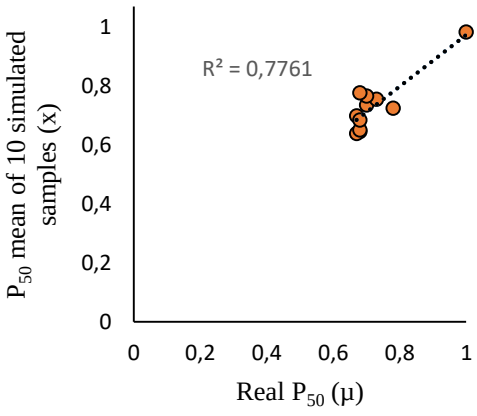
c) Area



d) Perimeter



e) Circularity



f) Roundness

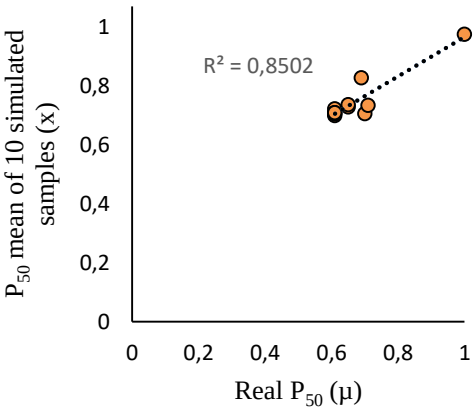


Figure 0-1 – R square of the different shape´s factor

Further, to ensure the precision of the virtual model the one sample student t-test has been employed, in order to test if the computed morphological factors are close to the experimental one. This information is shown in Appendix B.4.

The results of the t-tests reveal that they are no significant differences between the simulated and the expected values for most of the shape´s factors analysed, except for some stone´s types for the shapes factors: circularity and sphericity.

Besides, the cumulative frequency distribution curves of the shape´s features of all the particle´s samples were plotted, to compare the real vs virtual aggregates by simply representing one versus the other. Some examples are plotted in Figure 4-9, and the remainscan be found in Appendix B.5.

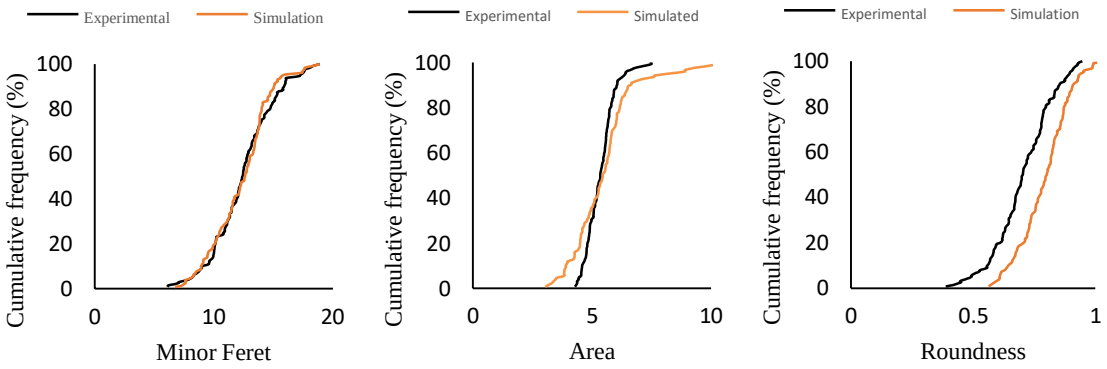


Figure 4-9 – Frequency distribution of minor Feret, area and roundness for the G2 aggregate sample

4.5 Conclusion

This Chapter shows a novel methodology to re-create commonly used aggregates in civil engineering. It is based on a on the use of: i) scalar functions to design free-form surfaces by mixing a spherical primitive and a Perlin noise; ii) the Marching Cube algorithm for generating the final triangular mesh for the stone´s visualization. The shape and size of the virtual aggregates are adjusted by rescaling and deforming the particles until they fall in the same statistical range of the real aggregates. The algorithm has been implemented in the NVIDIA PhysX physical engine as an alternative to DEM, with the objective of being able to use these aggregates in future research that will involve creating virtual granular assemblages.

The following conclusions have been obtained:

1. This methodology allows to re-create the morphologies, including concavities, of commonly used aggregates in civil engineering and generates shape´s indices very similar to those measured with the Image Analysis software ImageJ. This means that by the input of certain parameters, complete samples can be generated in a matter of seconds, which could be faster than other methodologies, such as imaging real aggregates.

2. It has been found that the methodology allows to re-create full gradations by using the Weibull distribution to decide the minor Feret and the aspect ratio of the virtual aggregates. This point, along with the previous one, refers to the fact that the algorithm created offers the possibility of changing particle size distribution and its shape in a fast way. This greatly extends the range, size, and type of particle systems that can be studied.

3. The area and perimeter, and therefore the circularity and roundness of the particles are controlled by three inputs that weight the spherical primitive and Perlin noise. In the current research, these inputs have been adjusted by trial and error, until the shape of the virtual shapes was found through visual inspection. In future research, an optimisation algorithm can be used to achieve an aggregate generation model more functional as well as to create the virtual aggregate in a more accurate way (Chapter 6).

4. In occasions, the methodology proposed in this Chapter produced aggregates that presented unnatural shapes, such a hole, due to the non-isotropic nature in three-dimensional space of the Marching Cube. This could be solved by implementing more advanced algorithms such as: the Marching tetrahedral. However, this topic is outside the scope of this research.

In summary, the proposed aggregate generation model in a PE can be useful to re-create realistically the virtual coarse and crushed aggregates that are normally used civil engineering, in a few seconds. This can then be used to simulate granular material at meso-scale level with specific applications such as: asphalt´s aggregate structures, ballast, or any other similar granular system, which is the topic of the next chapter.

CHAPTER 5. THE USE OF A PHYSIC ENGINE FOR THE SIMULATION OF A PARTICLE´S PACKING PROCESS

A three-dimensional particle packing model which has been developed by using the virtual aggregate generation model presented in the previous Chapter and the Unity3D platform is presented in this section.

This Chapter reported the simulation of several aggregate´s packing comprising single-size and various-size of grains. Their internal structures were quantified for a comparison with their corresponding actual samples, in order to demonstrate the viability of the proposed model. This chapter constitutes a preliminary demonstration of the feasibility of the PhysX Engine for the simulation of granular material at meso-scale level.

The methodology can be divided in three major procedures:

(i) First, a calibration of the Unity configuration is performed to optimize it and achieve a good balance between accuracy and efficiency, to allow the contact detection between complex shapes, including concavities, and the computation of a large number of interactions between those complex shapes.

(ii) Second, a pseudo code is implemented to calculate the volume of each aggregate in a sample to obtain their mass, by pre-setting the density of the material to be simulated. This serves to have the number of particles according to the weight of the desired sample. The validation of this operation has been done by simply comparing the number of stones of real particle´s samples with the simulated one.

(iii) Finally, several aggregates packing were simulated. The compaction process mimics the laboratory procedure described in Section 3.2.1, which include vibrations and a compaction under pressure operations. The validation has been done by comparing their fundamental packing characteristic´s parameters such as: porosity, tortuosity, connectivity, grains properties and void´s shape properties among others, with those obtained experimentally.

Results show that the properties from the packing experiments and simulations are highly comparable. This paper demonstrates the capacity of physics engines to simulate granular materials effectively, as well as quickly.

5.1 Introduction

In order to design civil engineering materials, such as asphalt, with optimised workability, durability and mechanical properties, it is of the foremost importance to understand how size distribution and shape of aggregates influence their porosity, stone interlock and amounts of asphalt binder or mortar required for an effective binding action [18][189].

Several restrictions and specifications about the shapes and grading of the aggregate exist to select the proper aggregates for the mixtures design, as it was described in Section 2.1. Nevertheless, there is no standard to determine how particle´s changes influence the overall material behaviour [1]. This is mainly because laboratory studies for the aggregate´s characterization and for the material behaviour are subjective, expensive and time consuming.

For an in-depth understanding of the influence of the size, and type of particle on the stone-based mixtures, several software packages and models for their simulation have been developed over the years; being the most widely used the Discrete Element Methods, as mentioned in Section 2.3.2. Nevertheless, most of the DEMs requires prohibitive consumer-grade hardware when entering complex shape; especially when those have concavities. For that reason researchers have recently started using Physics Engines for simulating granular material as an alternative to the classics DEMs [6][7][9], due to their computational efficiency when entering complex shape, as is demonstrated in [13], thanks to the contact resolution they use [5] (see Section 2.4). However, their accuracy is still disputed.

For that reason, the objective of this chapter is to develop a particles packing model using a Physics Engine, under the purpose to clarify if the representation of the structure they form is accurate enough, to encourage further investigation.

5.2 Experimental analysis of the real aggregates packing specimens

Several single-sized and various-sized packed coarse particles specimens in dry condition were manufactured and analyzed to evaluate the influence of the different size and shape on their packing characteristics, as presented in 3.2. The indices to measure the granular material used in this study were explained 2.3.3 and summarized in Table 3-3.

The samples were prepared according to the description in Section 3.2.1, which include a vibration process and a compaction by dropping a weight above the samples. The number of the particles of each sample were counted and their heights after compaction were recorded. These data are shown in Table 3-4 and they will be used to ensure that virtual samples have a consistent number of stones and dimensions.

After each real specimen were scanned, their internal structures were quantified and the results are shown in Table 3-5, along with the graphs of the air void content distribution with depth, which are plotted in Figure 3-7. Besides, the aggregate´s orientation and segregation of the various-size packing specimens were also studied as described in 3.2.2*;* to ensure that the PE can simulate the particle arrangement in a realistic way, since particle distribution and orientation can create anisotropy in a stone-based material which will influence the mechanical properties [32]. The measured vertical and radial segregation and the average orientation angle of the aggregates are shown in Table 3-7 and Table 3-8 respectively.

The outcome of this study serves to validate the packing model proposed.

5.3 Unity 3D game engine platform

Several Physics Engines which different capacities and license formats are available.A complete review can be seen in [190][191][192].The platform chosen for theaggregates and packing simulation is Unity3D which use the PhysX physics engine, currently developed by NVIDIA Corporation [193].

The requirements to choose a physics engine is based on the following reasons:

1) availability of free or economically licenced software.
2) good documentation.
3) Good balance between the numerical accuracy and the computational effort of the integrator solver. The NVIDIA's PhysX physics engine count with a dedicated hardware accelerator (a Physics Processing Unit (PPU)), which allow to improve the physics and add more objects in the simulations making it a prominent simulation tool [194] [195][196] [197][124].

4) Ability to detect collisions between complex shapes. Unity 3D is one of the few ones that can efficiently to assess collision´s detections between objects with complex shapes, thanks to the different options it provides regarding the collision detection techniques and mesh colliders; that must be adjusted depending on the simulation's objective. (Note that In Unity3D, a mesh collider refers to the mesh added to the geometry of an object to make it suitable for collision detection).

5) Scalability of collision detection. The PhysX engine is able to compute a large number of complex objects (64,000 individual rigid bodies) in a fast and accurate way [120].

Previous researchers have investigated the capabilities of different physics engines by performing several tests, concluding that PhysX is one of themost efficient, as can be seen in [190], [192], [195] and [198].

5.4 Parameter´s optimization and configuration of Unity 3D

Unity3D platform has various software libraries to re-create simulation such as rigid body-system, (which is the one suitable to simulate a particles packing process), in a physically accurate manner [120]. The rigid body system consists in a loop for the contact detection, the contact resolution, and the time integrator, which compute the motions of bodies based on the Newton-Euler equations for both: translational and rotational motions, as was described in Section 2.4. More detailed information about this process can be found at [199].

Nevertheless, it is necessary a generic setting optimization and parameter tweaking to get simulations as similar as possible to the real tests done in laboratories,

due the philosophy of prioritizing simulation speed over accuracy that the game´s engine has by default.

In this section the refinement focuses on 1) the implementation and optimization of a collision mesh that fits the visual model of the objects as closely as possible; and 2) the configuration of some parameters of the Game Physics and time integrator to achieve results that resemble the real-world physics, that would deliver for an appropriate balance between the precision of the physics performance with an acceptable time and consumer-grade hardware.

It is necessary to point out here that the simulation reported in this thesis has been performed using a Windows® laptop with an Intel Core i7-7700HQ/2,8GHz CPU and 16 GB memory.

There are many variables that can affect the accuracy and performance of the simulation. A complete guide can be found at [193]. The most significant parameters for our simulation can be divided into three main groups, which are described throughout the following points and summarised in Table 5-1:

1) The RigidBody parameters. The RigidBody is a component that exists in Unity3D which is added to the simulated objects to recreates their physical properties.

2) The Physics Setting parameters, which serves to adjust the physics global simulation.

3) The Time Manager, which allows the setting of several properties that controlthe timing of the physics calculation during the simulation.

It is worth mentioning another block of parameters that also affects the PE performance, the Quality section of the Unity3D platform. This section is mostly dedicated to the level of graphic quality (such as the textures and the shadows) and it comesat the expense of frame rate [200]. The level of quality can be selected, choosing between 6 different levels, from very low quality to ultra-quality. As our main objective is not to create a good visual quality but an interaction between objects as realistic as possible, it was decided to lower the quality level to the minimum one, to avoid any CPU memory overhead during the simulation.

5.4.1 The Rigid Body parameters

To achieve realistic particle movements and collisions, it is necessary to implement some material´s mechanics properties. The objects simulated must be defined in terms of shape, density, coefficient of friction and coefficient of restitution.

Material's properties

The quantification of the frictional properties of geological materials is still a topic that is being explored for better understanding the mechanisms that take place at the grain contact properties, which is beyond the limits of this research. Nevertheless, the selected parameters should have some consistency with the material being simulated. For convenience, all the simulated non-spherical grains have been set with the same coefficients of friction and restitution, based on previous research.

Despite the transition between moving and resting particles are controlled by the user, the static friction (μ_s) has been established based on experimental work performed for gravels found in [9]. This serves to take into account the roughness of the surface of the grains on the general behaviour of the granular flow during the

simulation. The coefficient of restitution, called “Bounciness” in Unity3D, was also extracted from the same work, which in turn was based on the report done for limestone of Robotham et al. (1995)[201].

The dynamic friction (μ_d) was set according to the results obtained by *K. Senetakis et al., (2017)* [202], where micromechanical sliding experiments were conducted on crushed limestone in dry conditions. On the other hand the friction between the particles and the wall of the container was extracted from [13]. It is worth mentioning that, in Unity3D, the friction of two colliding objects is combined and the final value is averaged [203].

Regarding the sphere glass sample´s properties, both: the static and the dynamic friction employed are based on the experimental work of Ishibashi et.al [204]; while the coefficient of restitution is taken from [205].

The density of the materials was set according to the laboratory data from the laboratory of the University of Nottingham (see Table 3-4).

The linear/angular drag, used mainly for fluid mechanics, is set to zero to reduce the computational expenses.

Collision mesh

The object’s collision meshes (also named *meshcollider*) that Unity 3D offers by default ranges from simplest primitive to convex polyhedra. Often, they do not match to the geometry of the objects (note that the virtual aggregate modeled for our simulation can have both: convex and non-convex geometry). Their implementation can lead to inter-particle penetration type problems since the collisions might be computed before true contact.

To address this problem and consider the concavities that the virtual objects might have, it is necessary to introduce a decomposition algorithm, which divides the concave geometry into a set of convex shapes [206]. This allows the collision mesh to fit the object's geometry and concave detections between objects.

For Unity 3D, various plugins which use different techniques exist for this purpose as: *the Collision Mesh Generator* plugin, which uses a NP-Hard approach technique [207]; the *Non Convex Mesh Collider* plugin, which exploits a divide and conquer strategy [139] or the *Concave Collider* plugin which utilizes a novel hierarchical segmentation approach [208].

The NP-hard approach technique has the inconvenience of creating a large number of vertices in the object's mesh, which is not convenient for the simulation performance. The divided and conquer strategy has the limitation of needing sophisticated analysis for the characterization of the 3D model, in addition of being too restrictive when adjusting the approved convex mesh, which can result in poor decomposition set. For these reasons, it has been concluded that the best option is the use of the *Concave Collider* plugin [206], which can adjust the 3D mesh with a smaller number of triangles in a fast way. A complete review of how these different techniques work is found in [208].

The definition of the different parameters that the *Concave Collider* plugin uses can be found in [209]. However, the information provided is limited and no details regarding the difference between the three algorithms that this plugin offers were found. Therefore, the method and the parameters selected has been chosen through trial-and-error tests.

This algorithm's configuration has been set with intermediate precision, as shown in Figure 5-1.b (where the *meshcollider* is represented as green lines), to reduce the computational effort.

This decision was taken after checking that the model configuration with medium precision mesh provided acceptable results (See Appendix C.1), while the implementation of a high precision mesh makes the software performance drop. Hence, a high precision mesh was impossible for samples with a larger number of stones (as S or L3) due to system failures due to high computational load.

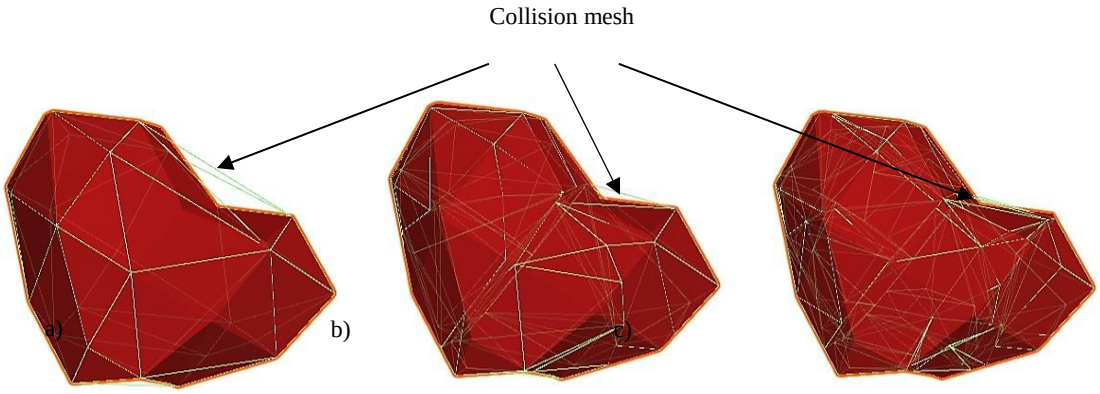


Figure 5-1 – Visual Particle with a) convex mesh collider; b) concave mesh collider with intermediate precision; c) concave mesh collider with high precision.

5.4.2 The Physics Setting parameters

The Physics setting section controls the global physics simulation of the game engine and certain adjustments are necessary so that it approaches the operation of the real world as closely as possible, within the limitations that will always be because the computation of the physics is limited.

ContactOffset

The ContactOffset controls the distance in which contact points (and therefore impulse forces) are generated between two potential colliding bodies [210]. In general, for the video games creation, where the time step of the constraint solver is usually large for a smooth real-time performance and the mesh collider are simplified, and therefore larger than the graphic´s shapes, this parameter has value larger than zero to see the graphic´s shapes to appear to be in contact when they are exactly touching. However, in spite of in our case the shapes of the collision meshes are adapted to the graphic shapes (thanks to the use of the Concave Collider plugin); the ContactOffset isset with a value slightly greater than zero (2mm), to increase the distance where contactpoints are calculated in order to: 1) solve the intermediate precision setting of the Concave collider plugin (see Figure 5-1.b) and 2) reduce as much as possible the particle´s overlapping, since being those in constant movement it is impossible to avoidit 100%.

Solver Iterator

The Solver Iterator solves physic´s constraints such as friction or coefficient of restitution; Unity3D incorporates the Non-linear Gauss-Seidel Solver [211]. This solver guarantees that constraints are respected by making a predefined number of iterations in each time step. The higher their values are, the greater the number of iterations, the more accurate of the resulting contacts or velocities after a *RigidBody* collides or bounces, at the expense of a major computational effort required [212][195].

The values of the Default Solver Iterations and the Solver Velocity Iteration that Unity3D have by default are 6 and 1 respectively. However, to obtain greater precision during the physics calculation, these values were increased based on previous research

[13], and trial and error tests, being the threshold value the one before the simulation becomes unstable due to the process overhead.

Collision detection system

As described in Section 2.4.2, the collision algorithm is divided into: the Broad phase level and the Narrow phase level.

Unity 3D uses the sweep and prune algorithm at the Broad phase level; whereas for the Narrow phase level Unity 3D offers several options [200]: 1) the *Legacy Contact Genreation*, which corresponds to the one named "a posteriori" or discrete collision detection and 2) the *Persistent Contact Manifold*, which corresponds to the “priori” collision or Continuous Collision Detection (CCD) technique, both briefly described in Section 2.4.2.

Several trials have been done with both of the options concluding that the CCD is the most suitable since check collisions in every timestep reducing like this considerably the object overlapping and passing through each other. This could be due to the simulation consisting of several small objects moving at a certain speed and that the use of a discrete collision detection can miss numerous collisions, leading to a particle’s interlock being less realistic [195]. The gravity is set to -9.8 (the default value).

5.4.3 The Time Manager parameters

The parameter fixed time step specifies how often the physics are calculated. Lowering this value increases the physics simulation's accuracy, avoiding the object's position error at the expense of lower Frames per Second (FPS), which affects smooth visualization.

For the video games performance, a minimum value between 50 and 70 FPS is accepted, however in the present work, the performance of the physics over a smooth visualization predominates, considering acceptable values between 6 and 10 FPS. In other words, with those values, the worst scenario will be a rate of 1 frame each 0.16 seconds.

The *Fixed Time step* controls the updating of the physics at a rate of:

$$Physics\ Update = \frac{1}{Fixed\ Time\ step} \qquad 5\text{-}1$$

Several tests have been carried out by lowering the pre-set Time Step by intervals of 0.002 seconds to see how it affects the FPS. As the number of rigid bodies also affects performance, the tests have been run on the sample which present the largest number of particles (Limestone L2 see Table 3-4). As an example, Figure 5-2 shows the variation of FPS when the time step value is lowered, being 0.002 the chosen value, equivalent to 500 updates of the physics per second, at FPS rate ranging from 4 to 7.

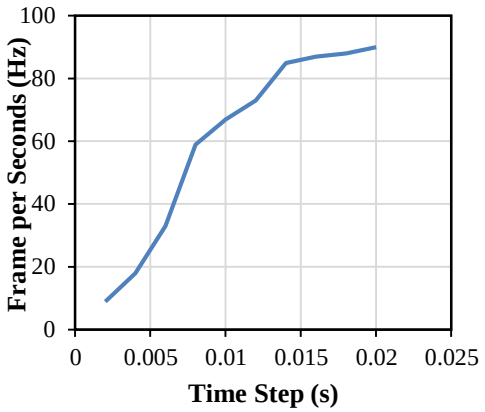


Figure 5-2 – Time step values VS Frame per Seconds for a gradation with 2300 particles

5.4.4 Summary of the Unity 3D Parameters configuration

The next table summarizes the values and the algorithm configuration of the variables optimized.

Table 5-1 – Physics' parameters configuration in Unity 3D

Rigid Body Parameters	Value/Algorithm	**Physics Setting Parameters**	Value/Algorithm
Collision Mesh shape	Concave Collider	Gravity (m/s^2)	-9.8
Mass (kg)	$\frac{V_s}{\rho_s}$ [a]	Contact Offset (m)	0.0002
		Default Solver Iterations	70
Static Friction, μ_s	0.793 [b]	Default Solver Velocity Iterations	30
Dynamic Friction, μ_d	0.109[c]	Contacts Generation	Persistent Contacts Manifold
Coefficient Of Restitution (Bounciness)	0.315[d]	**Time Setting Parameters**	
Collision Detection Type	Continuous Dynamic	Fixed Time Step (s)	0.002

[a] V_s is the virtual stone's volume & ρ_s is the material´s density.
[b] Refer to the non-spherical grain. For the SG sample μ_s was set equal to 0.243.
[c] Refer to the non-spherical stone–stone μ_d. For the SG sample μ_d was set to 0.216. The stone–container μ_d is set equal to 0.5.
[d] Refer to the non-spherical samples. For the SG sample the COR was set equal to 0.926

5.5 Particle´s packing simulation process

With the parameter's configuration described in the above section and the aggregate generation model presented in Chapter 4, the PE simulation was performed in Unity 3D to create three-dimensional virtual packing of particles samples with of prescribed size and shape distributions, and density. The packing simulations were made for the twelve different specimens presented in Section 3.2.1, which comprise nine one-sized packing specimens and three various-sized packing specimens.

The packing process can be divided into two main steps:

1) Pluviation. The packing simulation starts with aggregates being poured into acylindrical container with the same dimensions as the real one. The virtual

stones were dropped from the top of the container and settled down under gravity. The total number of particles in the simulation was defined by the total mass of the sample to be simulated and the material´s density, since the volume of each particle is measured by employing the c-sharp code, found in [213] (its mathematical explanation can be read in [214])

This serves to generate a number of particles according to the weight of the desired sample. This was validated by simply counting the number of stones of the real specimen (see Table 3-4) and comparing them with the simulated one. The results reveal a similar, but not identical, number of particles simulated (Table 5-2).

2) Compaction. The packing process mimics the laboratory procedure explained in Section 3.2.1. First, a vibration is applied during two minutes to the cylindrical container (Figure 5-3.a), with frequencies and amplitudes set for each of the three axes, found by trial-and-error tests (Appendix C.3). Second, a frictionless circular top weight, with the same mass and dimension as the real one, was dropped under the gravity from a height of 5cm (Figure 5-3.b) while the samples vibrated during two minutes (a video demonstration is linked in AppendixC.2). Finally, the height of the bottom of the compressor weight is adjusted to the actual height of the real specimens, and the vibration is kept during fourminutes more, for the rearrangement of the particles (Figure 5-3.c).

The final virtual particle assemblies were saved in a 3D obj format and exported to the free and open source 3D creation, Blender [215], for further analysis of the packed aggregates with the image processing software ImageJ (Section 3.2.2). This process is summarized in the Appendix C.4.

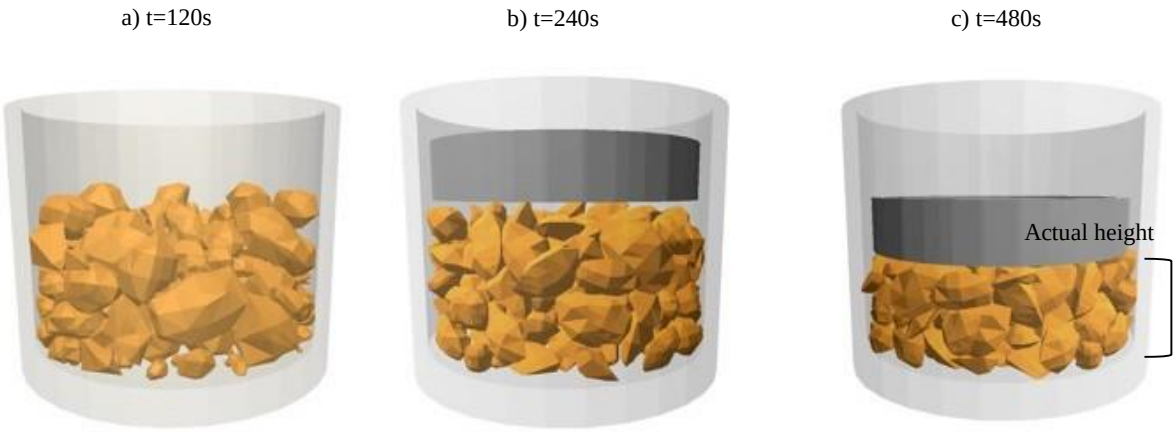


Figure 5-3 – Sequence of the virtual compaction process.

Note: a) vibration of the aggregates placed under gravity b) sample prior to compaction under surcharge weight and c) sample at end of compaction.

5.6 Results and discussion

All twelve aggregate packing specimens displayed in Section 3.2.1, have been virtually recreate and their internal structure quantified through the software ImageJ, using the parameters defined in Section 3.2.2.

5.6.1 Visual comparison and virtual mixtures' number of particles

Figure 5-4 compares visually a selection of the experimental and virtual specimens. There is a high level of realism in the simulated packings. Nevertheless, from the Figure 5-4. a and b (the Glass Sphere sample (GS)), it can be noticed that the experimental specimen shows more regular structure (crystallinity) than the simulated one.

The reasons could be several: 1) to the approximation of the inter-particles friction and Coefficient of Restitution, which controls the relative velocity and position before and after collision. This can be also extrapolated to non-spherical particles, in which their shape also influences the coefficient of restitution. 2) the compaction and

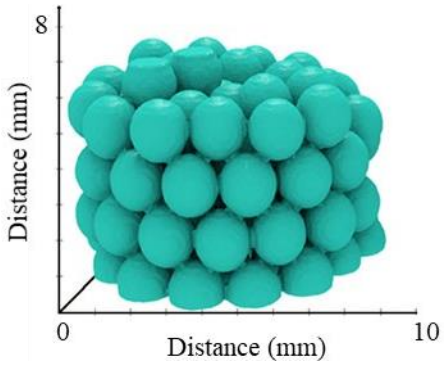
vibrating time might not be adjusted to match the real one adequately; 3) it might be necessary to improve the accuracy of the PE.

A comparison between the numbers of particles in the mixtures is shown in Table 5-2. Note that the mass simulated is the same for the experimental and virtual mixtures. There is a linear agreement between the experimental and numerical results, with R^2 0.99 for the mixtures' number of particles. Hence, aggregates were accurately measured and simulated, and aggregate packings can be accurately reproduced. All the particles' packed samples are plotted in Appendix C.5.1.

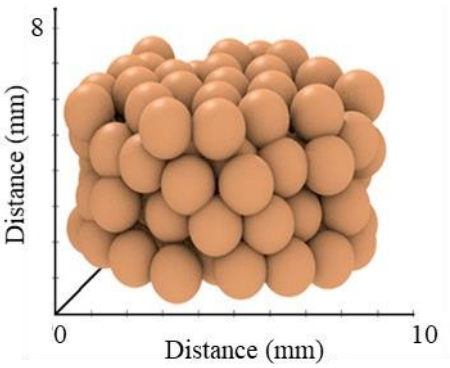
All the simulations carried out in this work had a duration from 10 minutes to 50 minutes, being the most conflictive the samples with a larger number of particles, where the FPS was close to 3 Hertz (making the simulation take longer compared to the other samples). However, the PE simulation can be considered very fast (taking into account that particles with concavities slow down the performance considerably), compared to the time that simulations made with classic DEMs last, as previously demonstrated other authors [13].

Table 5-2 – Comparison of the real and simulated number of particles

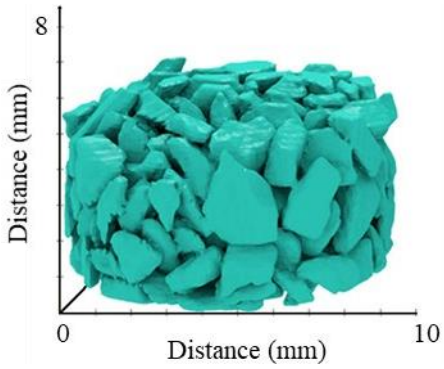
		G1	**G2**	**RG**	**L1**	**L2**	**L3**	**GS**	**CG**	**S**	**VS20**	**VS14**	**VS10**
Number of particles	*Experimental*	186	104	83	160	2305	905	102	1340	1330	789	507	581
	Simulated	194	110	78	166	2502	921	104	1298	1363	802	522	595


8
Distance (mm)
0
Distance (mm)
10

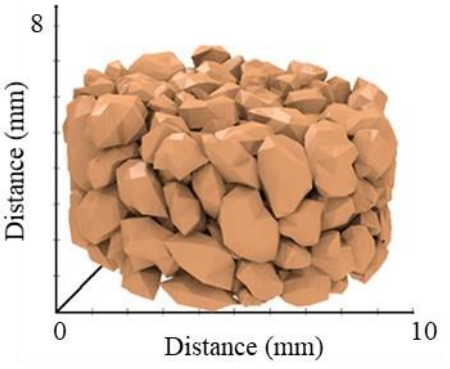
a)


8
Distance (mm)
0
Distance (mm)
10

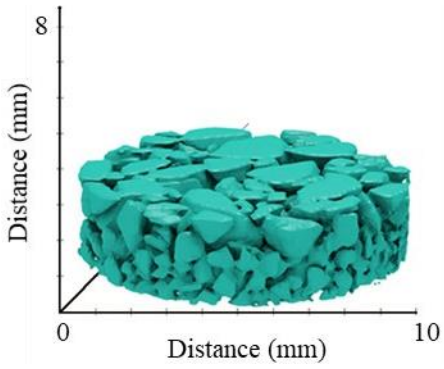
b)


8
Distance (mm)
0
Distance (mm)
10

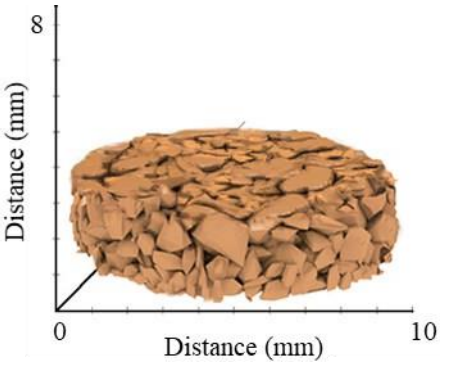
c)


8
Distance (mm)
0
Distance (mm)
10

d)

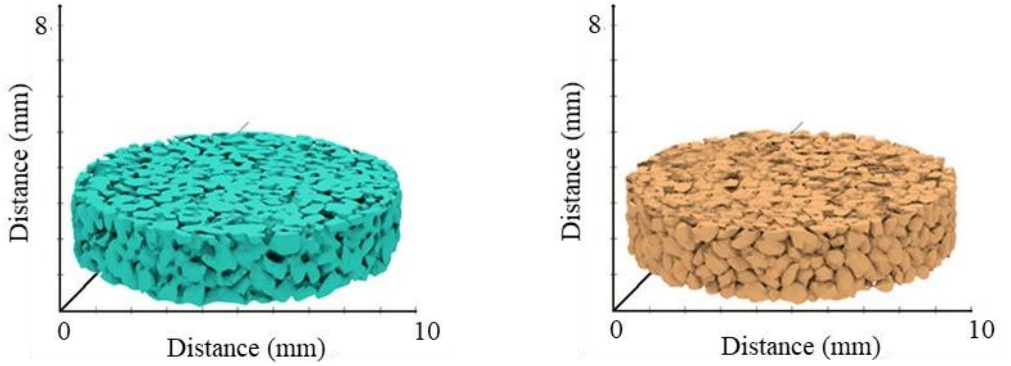

8
Distance (mm)
0
Distance (mm)
10

e)


8
Distance (mm)
0
Distance (mm)
10

f)

.



g) h)

Figure 5-4 – Comparison between real and simulated packed aggregates

Note: a) GS experimental; b) GS virtual; c) L1, experimental; d) L1, virtual; e) VS14, experimental; f) VS14, virtual; g) VS10, experimental; h) VS10, virtual.

5.6.2 Difference between the internal structures of the real and virtual particles packing

The results obtained from the analysis of the virtual specimens can be seen in Appendix C.5.2.

The packing model validation were asses through ratio between the experimental data and the actual one (R_{index}) and the percentage of error (% error); defined in Eq. 4-4 and Eq. 4-5 respectively (Section 4.4.2). Further, their Air Voids distribution curves were alsocompared, as well as the particle´s orientation and segregation of the various particles samples.

Table 5-3 – R values resultant from comparing the simulations with experiments

			G1	G2	RG	L1	L2	L3	GS	CG	S	VS20	VS14	VS10
Air Void Properties	Porosity (p)	R_{index}	1.07	1.05	1.13	1.07	1.13	1.06	1.02	1.09	1.09	1.08	1.08	1.04
		%error	7.29	5.06	13.48	6.75	13.33	5.51	1.65	8.89	9.05	7.54	8.26	4.07
	Average pore size ($\bar{\Phi}_v$)Φ_v)	R_{index}	1.09	1.07	1.17	1.09	1.12	1.11	1.07	1.10	1.11	1.08	1.10	1.10
		%error	9.38	6.80	16.75	8.86	11.76	10.67	6.53	9.89	11.24	7.74	10.37	9.59
	Maximum pore size ($\Phi_{v.max}$)	R_{index}	1.07	1.05	1.32	1.14	1.14	1.15	1.04	0.97	1.11	1.04	1.13	1.06
		%error	6.94	5.32	31.77	14.48	13.72	14.93	4.19	3.36	11.24	3.59	13.00	5.88
	Tortuosity (τ)	R_{index}	0.88	0.93	0.84	0.91	0.84	0.87	0.88	0.87	0.90	0.87	0.91	0.89
		%error	11.90	6.92	16.31	9.35	16.15	12.78	11.76	12.80	9.68	13.22	8.55	11.11
	Connectivity (z)	R_{index}	1.11	1.08	1.07	1.14	1.15	1.12	1.07	1.15	1.16	1.18	1.19	1.17
		%error	10.53	7.57	7.19	13.52	14.85	12.15	6.94	15.24	16.11	18.32	19.38	17.03
	Aspect Ratio (AR)	R_{index}	1.30	1.32	1.21	1.17	1.26	1.16	1.04	1.25	1.33	1.26	1.15	1.19
		%error	29.79	31.71	20.59	16.67	25.80	16.17	4.00	24.72	33.33	25.53	14.58	18.75
	Circularity	R_{index}	1.37	1.29	1.29	1.38	1.32	1.29	1.10	1.29	1.36	1.42	1.44	1.38
		%error	36.67	28.89	28.85	37.85	31.67	28.62	9.76	29.03	36.36	41.82	44.00	37.74
Aggregate Properties	Average aggregate size ($\bar{\Phi}_a$)	R_{index}	0.95	0.96	0.88	1.09	0.95	1.03	0.98	0.87	1.12	0.94	0.93	0.91
		%error	5.00	4.22	12.15	8.61	5.19	2.73	1.99	13.33	11.60	5.71	7.50	9.28
	Maximum aggregate size ($\Phi_{a\ max}$)	R_{index}	0.93	0.94	0.87	0.85	0.92	0.85	0.98	1.21	0.93	0.93	0.92	0.96
		%error	7.01	6.35	12.61	15.40	7.89	15.45	2.34	21.39	7.34	6.50	8.20	4.10
	Node density (n_d)	R_{index}	0.68	0.73	0.60	0.63	0.67	0.67	0.88	0.85	0.67	0.55	0.58	0.65
		%error	31.58	26.67	40.00	36.84	33.04	33.33	12.50	14.69	32.76	45.00	42.00	35.00
	Roundness	R_{index}	1.15	1.14	1.08	1.28	1.41	1.28	0.98	1.29	1.31	-	-	-
		%error	15.38	14.29	8.45	27.54	40.98	27.87	2.00	29.23	31.15			
	Circularity	R_{index}	1.03	0.98	1.10	1.09	1.06	1.10	1.14	1.09	1.14	-	-	-
		%error	2.74	2.00	9.86	8.96	6.41	10.45	14.29	8.82	14.49			
	Aspect Ratio	R_{index}	1.04	1.06	1.07	1.08	1.09	1.04	0.99	1.06	1.12	-	-	-
		%error	4.35	5.63	6.58	7.69	9.23	4.41	1.00	6.25	12.12			

From Table 5-3 it can be seen a percentage of error between 2% and 20% for the properties: porosity, connectivity, tortuosity, stone´s size and void´s size over all the samples. Porosity and pore size reach higher values in simulated samples (R_{index}>1), which is expected since virtual samples are "cleaner" without traces of dust or particle breakage. The connectivity of the pores is also more significant in the simulated

samples. However, we see that these have a lower tortuosity, which is also to be expected since the virtual stones do not have the same roughness that the real ones have.

The grains shape´s factor as well the void´s shape factor present error range 15-30%; (excepting for the SG sphere). Both: the pores and the grains of the virtual samples tend to be more rounded than those analyzed experimentally (R_{index}>1).

The particle node density (which represent the stone-stone contact as explained in Section 2.3.3), present the biggest difference between real and simulated values. This could be due to a bigger simplicity in the structures formed by virtual samples (rounder particles, the contrast between the sharpness of the virtual and experimental images, amount other factors).

5.6.3 Comparison between the real and virtual Air Void´s distribution curves

The goodness of fit of the computed to real void volume has been quantified using the Kolmogorov-Smirnov Test [216] [217], using the critical value for a confidence level at 95%($D_{crit,0.05}$). This value has been obtained with (Eq. 5-2). If $D_{crit,0.05}$, is smaller than the maximum absolute distance between the expected and observed distribution functions (D_s), we can conclude that the values from the experiments and simulations are equivalent. This test is based on the maximum difference between the empirical and the numerical cumulative distribution without any assumption about the distribution. The results are presented in Table 5-4.

$$D_{crit,0.05} = 1.36 \sqrt{\frac{1}{n_x} + \frac{1}{n_y}} \qquad 5\text{-}2$$

Where,

n_x and n_y are the population size of the samples being compared.

Table 5-4 – D_s and $D_{crit,0.05}$ of experimental and simulated vertical porosity curves

	G1	G2	RG	L1	L2	L3	GS	CG	S	VS20	VS14	VS10
D_s	0.012	0.013	0.040	0.024	0.016	0.008	0.020	0.008	0.009	0.035	0.023	0.028
$D_{crit,0.05}$	0.027	0.028	0.037	0.023	0.029	0.026	0.026	0.028	0.030	0.032	0.025	0.028

The D_s values are smaller $D_{crit,0.05}$, except for the samples RG, L1, and VS20. In these cases, this might occur due to difference between the real and virtual particles, or because the compaction levels are not identical. How to simulate different levels of compaction and how they influence the structure they form need to be assessed in future research. Even so, it can be confirmed that the virtual particles packing process has a very close AV distribution function than the experimental one.

An example of the Air Voids vertical distribution curves is plotted in Figure 5-5. All the AV distribution curves can be seen in Appendix C.5.3.

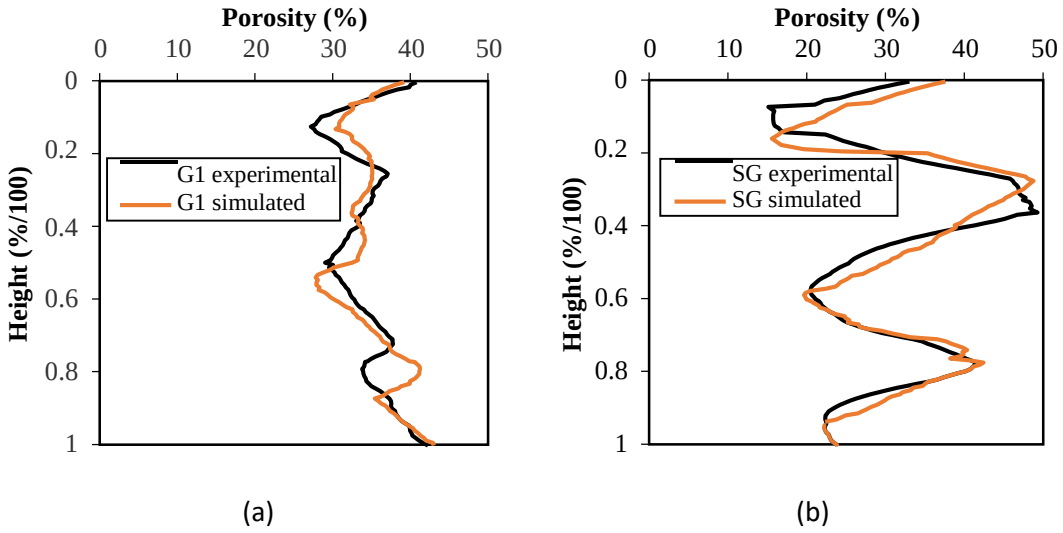


Figure 5-5 – Distribution of vertical porosity for (a) G1 and (b) GS mixtures

5.6.4 Comparison between the real and virtual particle´s segregation and orientation

Finally, the aggregate segregation and orientation of the virtual various-size samples have also been analyzed, by using the method described in Section 3.2.2; and the results are presented in Table 5-5 and Table 5-6, respectively.

Table 5-5 – S_L Values for vertical and radial segregations for the virtual specimens

Specimen by Nominal Size	Vertical Segregation			Radial Segregation		
	Average Minor Feret (mm)		S_L	Average Minor Feret (mm)		S_L
	Top	Bottom		Outer	Inner	
20mm	6.2	5.1	19.57	4.1	4.69	-9.58
14mm	4.78	4.02	15.91	4.67	5.1	-8.43
10mm	3.15	3.54	-7.02	3.51	3.55	-1.12

Table 5-6 – Average orientation angle, εº, for the virtual specimens

Specimen	Vertical direction			Radial direction	
	Top	Bottom	Throughout specimen	Outer	Inner
VS20	50.879	54.78	58.87	61.34	53.78
VS14	58.32	61.87	55.65	60.87	52.54
VS10	*55.95*	*56.15*	*55.34*	*56.91*	*55.23*

Table 5-5 shows that the virtual mixtures have approximately 40% and 17 % lower vertical and radial segregation than the experimental ones, which may occur due to the lack of fine aggregates in the virtual specimens.

Regarding the average angle's orientation of the grains, (Table 5-6), the $\bar{\varepsilon}$ value is close to 50º for all the specimens. However, this value is 10% higher in the virtual mixtures than in the experiments, which means that the virtual particles tend to be more oriented in the circumferential direction than the experimental ones. The reasons for this phenomenon could be related to the slight differences in the dimensions between the virtual and physical aggregates, or in differences on the vibration patterns. It is unclearhow relevant this difference would be to measure properties such as elasticity or hydraulic conductivity and further investigation are needed.

5.7 Conclusion

This chapter proposes a novel methodology to simulate 3D aggregate packing by using a physics engine that generates virtual stones with prescribed shape and size distribution and compacts them using vibration and vertical loading. 12 different mixtures have been reproduced computationally, whose internal structure characteristics at mesoscale level are similar to the experimental one; demonstrating thus the model's capability and flexibility.

However, the designed particle packing process simulation is in an early stage and its feasibility still need to be improved by fine tuning the parameters that control the physical properties of the granular material and the PE variables, in addition to being verified with further research, due to the number of variables involved.

The following conclusions can be drawn:

1. A Physics Engine can be used to simulate dry granular material with a high level of realism. The virtual 3D aggregates packing, and the experiments are comparable in voids size and shape properties, particle size shape properties, particle orientations, and segregation.

2. For an equivalent mass of materials, a linear relationship exists between the number of aggregates in the virtual and experimental mixture.

3. On average, the virtual mixtures were 7% more porous, their pores were 10% bigger, 12% less tortuous, with 13% more connectivity, 22% higher aspect ratio and, 32% higher circularity than the experimental mixtures.

4. The numerical method used allows creating virtual mixtures with a similar vertical distribution of pores as those measured experimentally using Computed Tomography scans.

5. The virtual mixtures show approximately 40 % and 17 % lower vertical and radial segregation than the experimental ones respectively. After compaction, the virtual aggregates are 10 % more oriented in the circumferential direction than the experimental ones.

6. The time required to do the simulations were in average around 30 minutes, which allow a rapid study of different granular materials, to improve their design, save laboratory time, cost, and better use of the natural resources.without the need of lengthy and costly laboratory work.

7. With further research, this could be a potential tool various fields dealing with stone-based materials to improve their design, save laboratory time, cost, and better use of natural resources.

The model presented in this paper shows the potential of a new stone assembly simulation approach in real-time, which could be a starting point for future simulation, in the study and optimisation of aggregates blend design at meso-scale level, strongly related with the stone-based mixtures' performance [87], in a faster way than other theoretical and experimental models available in the literature.

The next chapter will be based on the simulation of different asphalt gradation curves of the coarse aggregates and their packing to represent asphalt skeletons.

CHAPTER 6. POROSITY PREDICTION MODEL FOR POROUS ASPHALT MIXTURES THROUGH A PHYSICS ENGINE

Creating a simulation tool for aggregate selection in asphalt mix designs and the computation of their porosity is one of the major aims in pavement engineering. It is a challenging topic since asphalt mixtures contain a wide range of particle sizes and shapes.

In this chapter, a Physics Engine (PE) is used to generate and visualise the compaction of several virtual porous asphalt skeletons to predict their air void content based on their volumetric properties. The virtual mixtures have aggregates with realistic shapes and gradations. The virtual aggregates have been created using the aggregate generation model presented in Chapter 4. The feasibility of the PE to provide packed particles assemblies with similar characteristics to the real one was proved in Chapter 5.

The work described in this chapter is divided into two mains parts:

(i) First, the particle generation model presented in Chapter 4 has been optimised using a Differential Evolution algorithm to improve it and create a more computationally efficient tool, which prevents finding by trial and error the variables that define the aggregate properties (see Section 4.3.1).

(ii) Then, a virtual compaction process is developed to achieve the same level of compaction as a Superpave Gyratory Compactor. The goal is to create a simulation tool that can stop when the coarse aggregates are fully compacted by applying an adequate maximum pressure and accounting for penetration between particles. The prediction of the air voids content is based on the percentage of each range of aggregate sizes included in the sample, the

percentage of fine particles, the percentage of binder and the dimensions of the cylindrical samples.

6.1 Introduction

Mixture porosity has a significant influence on asphalt's performance. Nevertheless, its prediction is difficult due to the complex shapes of asphalt mix particles and how they fit together. Asphalt mixtures are normally designed using trial and error laboratory tests, which is time-consuming and requires significant human and material resources. As a result, it is very difficult to obtain a large amount of consistent experimental data to explore the relationship between the mixtures' porosities and their volumetric properties, gradation, and asphalt content. For this reason, their virtual simulation to predict their mechanical properties and behaviour is currently a popular topic in civil engineering. Some of the researches that predicted the porosity of granular materials are cited below.

In [218], the authors designed a virtual porosity prediction model through the use of Discrete Element Modelling (DEM) software and a Back Propagation Neural Network Algorithm (BPNN). First, using the software Particle Flow Code (PFC), the authors generated static compaction system to pack spherical coarse aggregates with randomly generated gradations. Then, they “trained” the BPNN to achieve realistic gradations and accurate porosity prediction. The results were validated by performing a laboratory test consisting of mixing steel balls in the same proportion as the virtual tests, with accurate results.

To extend this model to predict the porosity of granular material considering realistic particle shapes, the authors used a reduction factor on the predicted results, which was achieved by using virtual aggregates modelled with the clump template

technique [101]. Nevertheless, it was observed that the results obtained by the two models, i) the one using spherical particles plus a reduction factor to take into account the stone´s shapes, and ii) the one using the clump template techniques to model the aggregates, were quite different.

In [219], the authors used the open DEM source YADE to simulate the Superpave Gyratory Compaction of porous asphalt, investigating the influence of the number of gyrations, the gyration angle and the parameters of the contact model, also using spheres to model the coarse aggregate [219]. The model was validated by conducting laboratory compaction tests of Hot Mix Asphalt (HMA) using rounded gravel as aggregates. The results revealed that the air voids predicted by DEM simulation were lower than the actual ones, because spherical particles are easier to compact as there is no interlock between them if more angular particles would have been modelled.

In [220], the authors used spherical particles through a DEM software to estimate the Voids in Mineral Aggregates (VMA) of HMA. Firstly, they propose a new method to characterise the different gradation types; then they define a parameter named *percentage of voids change;* with which they link the aggregate gradation property to the VMA of the HMA. The VMA prediction method was verified with the VMA values of different mix designs from eastern Washington paving projects, without experimental testing. The particles shapes were indirectly considered by adjusting model parameters such as the contact friction angle between particles. Most of the DEM-derived values were close to the theoretical mix-design values.

Although the previous works represent an advance for the design of HMA, all of them have in common the use of spherical particles as virtual aggregates and their gradation as a major controlling parameter over their assembly structures. In contrast

the shape of the coarse aggregates is not taken into account or is taken into account indirectly through pre-defined parameters such as reduction factor of contact friction angles. Nevertheless, the shape effect of the aggregates assembly is non-negligible as they play an essential role in the mesoscopic structure of mixtures [87] [221].

For this reason, in this chapter an air void prediction model of porous asphalts is presented, by employing virtual aggregates with complex shapes and the use of a Physics' Engine. This engine can compute particles´ movement and their interaction in an accurate way, as demonstrated in the previous chapter.

Hence, this chapter presents an air void prediction model of porous asphalt by employing virtual aggregates with complex shapes and a Physics' Engine able to compute particle´s movement and their interaction in an accurate way, as demonstrated in the previous chapter.

6.2 Virtual aggregate generation model optimisation

The analysis to validate the aggregates generation model presented in Chapter 4 revealed that the shapes factors: area, perimeter, roundness and circularity of the virtual particle´s samples need to be more accurate, since they present a percentage of error of the 50th Percentiles (P_{er50}), which were obtained from the statistic measurements of the virtual samples (Equation 4-5), between 10 and 20% respect to the experimental data, in almost all the samples simulated. This was also confirmed through the R-squared analysis, used to compare the good of fit between the real data and the simulated one (see Section 4.4.2).

This is because those morphological features depend on a noise generation function, which is in charge to deform the particles until a certain shape is achieved. In

turn, this noise generation function is controlled by several variables set by trial-and-error tests until the stone type to be simulated matches visually with the Krumbein and Sloss chart classification for aggregates (see Section 4.4.1).

This process was subjective, complex and time-consuming. In addition, the accuracy of the results strongly depends on the complexity of the shape to be simulated (i.e., samples as Sphere Glass (SG) and round gravel (RG), are easier to classify and therefore to simulate).

This is because those morphological features depend on a noise generation function, which is in charge to deform the particles until a certain shape is achieved. In turn, this noise generation function is controlled by several variables set by trial-and-error tests until the stone type to be simulated matches visually with the Krumbein and Sloss chart classification for aggregates (see Section 4.4.1).

This process was subjective, complex, and time-consuming. In addition, the accuracy of the results strongly depends on the complexity of the shape to be simulated (i.e., samples as Sphere Glass (SG) and round gravel (RG) are easier to classify and therefore, to simulate).

This section discusses the implementation of a Differential Evolution algorithm (DE), with the purpose to make the aggregate generation model more functional and increasing the model accuracy by adjusting those inputs, which serve to create different particle's roundness, and which were found by trial and error, automatically.

The DE has shown good performance in addressing optimisation problems to generate realistic air void shape properties (roundness and circularity) for studying virtual asphalt pore networks, as can be seen in [154].

To reduce computational times, the DE optimisation will be used only to find the 50th percentiles of the area and the perimeter since the roundness and the circularity are parameters calculated by a formula that depends on the area and perimeter,respectively (Equations. 2-2 and 2-3). Thus, they will improve once the area and perimeter are optimised, as is shown in Figure 6-1.

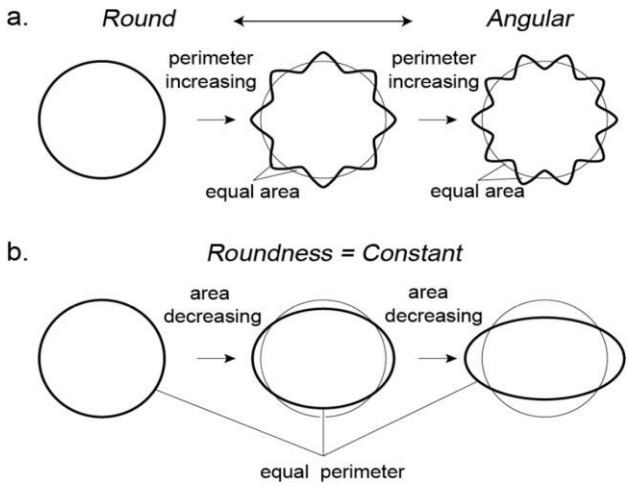


Figure 6-1 – How changes in the area and perimeter change particles angularity

Note: Image from: New parameter of roundness R circularity corrected by aspect ratio [81].

6.2.1 Differential Evolution Description

The Differential Evolution (DE) is a metaheuristic optimisation technique that has been widely used in different science fields, thanks to its effectiveness and the fact that it does not need previous information to solve problems. However, it can determine the quality of fit of any possible given solution.

As the actual problem can be considered as a low dimensional problem (only threevariables need to be optimized), the classic DE has been implemented to simplify the process. The different procedures that DE can use can be found in [222] [223]. In addition, it gives the possibility of introducing physical constraints on the variable to

be optimized, to improve its performance. The DE code adapted to the c-sharp language and implemented in the aggregate generation model can be found in [224].

Figure 6-2 shows flow chart about the DE optimisation process implemented in the virtual aggregate generation model. Once the input parameters are set, the DE consist of a series of mutations and a crossovers techniques from an initial population of individuals, to create new ones with better values, according to an objective function (also known as Fitness Function (FF)), which serves to compare how close are the results obtained to the one desired [222].

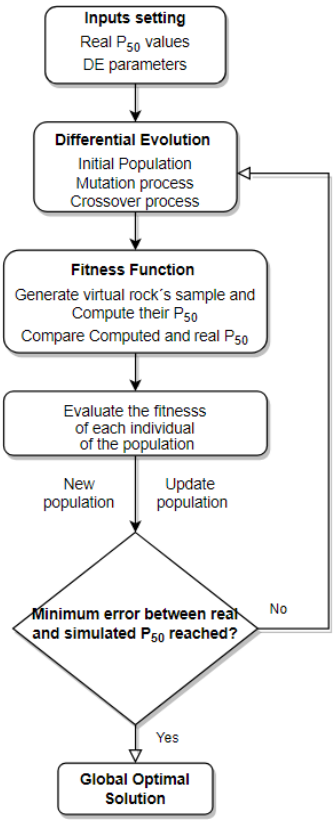


Figure 6-2 – Flow chart of the DE optimization process

A summary of how this methodology works is described below (the detailed method´s description and be found in Appendix D.2)

First, the users need to define the size of the initial population of individuals, which are numerical vectors consisting of a random combination within the lower and upper boundaries established for the different solutions of the parameters to be optimized (Table 6-1). This technique assumes a uniform probability distribution for all random decisions [223]. The size of the initial population is named from now *NP*.

Then, the evolution starts by applying a mutation strategy in each individual of the first population generated, named target individuals, to create mutant vectors.

Each mutation consists on selecting randomly three members of the population. Then, to one of the vectors is added the weighted difference between the two remaining individuals. The weighted difference is done through a mutation constant, *F*, named scaling factors, which control the amplification of the difference vector.

Then a child individual is created by crossing the mutant vector with another random vector selected from the population, based on a crossover rate, *Cr*. This process is known as Crossover. Each child individual from the population is a trial solution.

Finally, the trial vector and the current individual are compared through a Fitness Function $f(I)$, in charge to evaluate the solution created in each mutation and crossover, to know which individuals produce the fitter results, in order to choose the best for next generation of individuals.

In this research, the Fitness Function is defined as the percentage of the absolute errors of the computed P_{er50} and the experimental data (Equation 6-1).

These stages are repeated in subsequent iteration, until $f(I)$ reach a minimum error defined by the user. In each iteration several virtual rocks are generated, and their morphological information measured.

$$(I_i) = \left|1 - \frac{P_{er50\ virtual}}{P_{er50\ real}}\right| * 100\ ; \quad f_i(I) \leq error \qquad 6\text{-}1$$

During the evaluation, several rocks of the type analyzed are created and their morphological information: area and perimeter, are measured following the procedure explained in Chapter.4 (Section.4.3.2).

The DE optimization can be defined mathematically as minimization problem, since these stages are repeated in subsequent iteration, until (I_i) reach a minimum error defined by the user.

The following section presents a summary of the inputs necessary for the DE process and the chosen values, according to the analysis carried out.

6.2.2 Inputs parameters setting

As described in the previous section, the DE is controlled for three main variables which are: the population size (*NP*), the scaling factor (*F*), and the mutation rate (*Cr*). Their fitness value depends on several factors: the problem dimensions (the number of parameters to be optimized) and the type of problem to be solved.

Other parameters to take into account are: the number of generations allowed (*G*), which is an stop criterion to avoid stagnation when the algorithm does not find the correct direction to find the optimal solution [225]; the upper and lower boundary constraints of the parameters to be optimised; the maximum error allowed; and in our case, the number of virtual rocks to be created (*r*) for the calculation of their $P_{er50\ virtual}$.

In this study, the parameters to be optimised are three (low dimension problem): the Sphericity Control (*Sc*), the Intensity Deformation (*Idef*) and the First Random

Value (*f*), which are those which control the noise function, and which are defined in Section 4.3.1. To prevent the generation of inconsistent solutions and to ensure a fine-tuning on the adjustment of the 50th percentiles, those variables have been bounded by lower and upper limits. The parameters were also rounded to a fixed number of decimal (Table 6-1). These boundaries were experimentally determined in Chapter 4 as was shown in Figure 4-7.

Table 6-1 – Boundaries Constraints of the Parameters to optimise.

Parameter	Range	Rounding	Unit
Sphericity Control (S_c)	0-3	10^{-15}	-
$Intensity\ Deformation$ (I_{def})	0-1	10^{-15}	-
First Random Value (f)	0-3	10^{-15}	-

An accurate choice of the DE parameters is important to achieve a fast convergence toward the optimal solution, avoid a slow optimization process and enable an agile simulation tool. The fine-tuning of these parameters have been done by trial and error, by testing different variable combination and evaluating the performance's approach.

For an easier selection of the DE´s parameters, the search was simplified by fixing the variables: *NP*, *G* and *r*. The maximum error was set to 10% (Table 6-2).

According to previous research, the *NP* should range between 5 and 10 times the dimension of the problem [226]. Therefore, to ensure that the initial population have enough mutually different vectors and avoid a large population size, (which can affect the ability of the algorithm to converge to the optimal solution), a value of 20 was thought appropriate [225].

The maximum iteration allowed (*G*) was set to 100, to prevent the algorithm from running an excessive time.

Regarding the number of the virtual rocks to be generated (r), an analysis of 100 stones is considered enough for their P_{50} evaluation, based on preliminary parametric analysis results carried out throughout this work.

The remaining DE parameters: *F* and *Cr*, were obtained by trial and error by running several tests until the global minimum error was achieved. To ensure a correct direction search of the algorithm, the initial *F* and *Cr* values were chosen according to previous research, as is explained below.

In [227], the authors conclude that for small population size (20-40), a high crossover (*Cr*) rate is preferred. In [228] the authors concluded that a small scaling factor *F* leads most robust search in DE with small *NP*, since high mutation rates normally led to random search [229]. Therefore, they proposed a value of $F=1/n$ as an effective choice, being *n* the dimension of the problem.

Considering the above mentioned, the first test was conducted using a 0.33 scaling factor and a 1 crossover rate, to see if a quick convergence was possible.

The following tests were run by combining these two parameters in various values, using increments of 0.01 for *F*, and increment of -0.01 for *Cr*; as their optimal values are normally found between [0.4,1.0] and [0.8,1.0] respectively [230] [231].

The final setting *Cr* and *F* values to minimise the error are plotted in Table 6-2.

Table 6-2 – DE parameters setting

Population Size (NP)	20
Maximum number of Iteration (G)	100
Error (%)	10
Number of virtual rock (r)	100
Scaling Factor (F)	0.42
Crossover Rate (CR)	0.87

6.3 Experimental analysis

In Section 3.3 was presented the experimental work associated with the current chapter. A summary is provided below.

Eight cores of porous asphalt mixtures with different aggregates blends were designed for this study have been compacted using a Superpave Gyratory Compactor (SGC) under the same conditions of: pressure, number of gyrations and gyration angle.

The asphalt mixtures were manufactured by blending in different proportions the limestone coarse aggregate: L2, L3, L4 and L5 (Table 3-9), whose sizes and morphological properties are shown in Table 3-2.

The sieve No. 4 was used to sort the aggregates into coarse and fine groups (named as dust from now). The sample´s weight, their maximum density as well as their target density and target porosity are plotted in Table 3-10.

The analysis of the compacted specimens consisted of recording the bulk density parameter (after the number of gyrations established is achieved), which is back-calculated by the SGC, which bases its estimates on: the distance between the base of the top and bottom plates (the final height of the samples), which were placed in the mould previous the compaction process, the mass of the samples and the dimensions of the mould. Then, the air void content was calculated in accordance with BS EN 12697 8:2018 (see Section 3.3.3). Besides, the porosity resulting from the SGC data were

compared with the porosity obtained through laboratory test using the sealed samples techniques on the same six cylindrical cores for each mixture´s design, to ensure that the results obtained by SGC are correct. The results of both analyses can be seen in Table 3-10.

The outcome of this analysis serves to validate the proposed PE asphalt's skeleton compaction model, by comparing the porosity resulting from the SGC with the virtual approach.

6.4 Virtual production of asphalt solid skeletons

The same Physics Engine (the PhysX engine through the Unity 3D platform) used for the particle packing model presented in the previous chapter, is employed in this study to generate the virtual asphalts skeletons compaction model. The main goal is to predict the air void content of the mixtures, based on their volumetric properties.

A uniaxial dynamic compression load plus a vibration applied to the centroid of the rocks was chosen to mimic the composition of SGC, as it was found to be the most suitable to achieve the desired compaction effect without losing computational efficiency.

The first compaction simulation attempt considered the angle of gyration and the rotational movement of the SGC in the virtual model approach. However, the collision detections between the stones and the top and bottom compressor plates, and between the stones and the container were not robust enough. Many contacts were lost, and many particles passed through the container walls and platens, as can be seen at the video link disposed in Appendix D.1. This phenomenon was accentuated as the number of

particles with complex shapes increased, so that the computational performance decreased significantly.

For these reasons, it was decided to simplify the model to one providing a uniaxial dynamic compressive load under the premise that the final result – the maximum compaction of the coarse aggregates – could be achieved in this way.

The virtual asphalt mixture´s skeletons specimens were made by mixing the same proportion of coarse aggregates as those used in the experiment (see Table 3-9), by generating the same limestone aggregates type through the methodology as described in Chapter 4 and its optimization, presented in this chapter. The particle combination was done based on the volumetric percentage of the entire sample yet with only the coarse aggregate fractions being simulated.

The virtual stones samples are random irregular particles that follow a specific gradation and shape´s properties, according to the size range and the stone´s type to be simulated (Sections 4.3.1 and 4.3.2). Pictures of the real and the virtual stones used in this work can be found in Appendix D.3.

The inputs necessary for the generation of the asphalt skeletons of the different samples are shown in Table 6-3. The mass of the aggregates was calculated by multiplying the volume of each aggregate generated by the density of the materials, (Table 3-4) and summing all the stone masses, as explained in Section 5.5. The density of bitumen was assumed 1.03 g/cm3.

The virtual aggregates were placed in a cylindrical container with the same diameter as the real mould used for the SGC (10cm), and were compacted using two cylindrical pistons, one at each extreme of the cylinder (Figure 6-3).

The physics parameters configuration in Unity 3D were keep the same as those detailed in Table 5-2, except the gravity, which was set to zero to avoid particles segregation. In this work the particles are frictionless.

For each specimen, the air prediction model follows the next steps:

i. Implement the input variables in the virtual compaction model for the generation of asphalt skeletons and calculate their air void content, based on their volumetric properties, once compacted. The different input variables and the calculation implemented in the algorithm are presented in Section 6.4.1.

ii. The particles were generated without gravity, to avoid segregation, and they are floating randomly in the cylindrical area. The cylindrical container was modelled with a height of 60cm to have enough space to allow the stones to mix (Figure 6-3.a). For the stones to be subject to vibration, a force of 2 N with random direction is applied to their centroids for 0.01 s. This was repeated every 0.5s. This force was selected based on experience as it was observed that it is enough to simulate aggregate mixture's movement.

iii. The two pistons used to compress the sample moved vertically, towards the centre of the container, in two stages: 1) first, the two pistons moved at a constant speed of 0.01m / s, until the samples reach 35% air void content, to allow rearrangement of the particles (note that all experimental samples have a lower air void content value than 35%). 2) The pistons then moved under pressure that progressively increased by a fixed amount of 116 kPa/min, for a gradual compaction, until it reaches the maximum force, set at of 350 kPa (Figure 6-3. a and b), as explained below. The maximum force was found through trial-and-error testing and the details are explained in Section 6.4.2.

Due to the nature of the physics engine, overlapping stones caused jitter (high frequency disturbances) [232] between the virtual aggregates, which made them vibrate, contributing to their compaction by re-accommodating them.

iv. The air void content of the mixtures was then calculated, based on the cylindrical volume and the particle volumes. For that, the positions of the compressive pistons were recorded during the simulation to know the air void content in each moment.

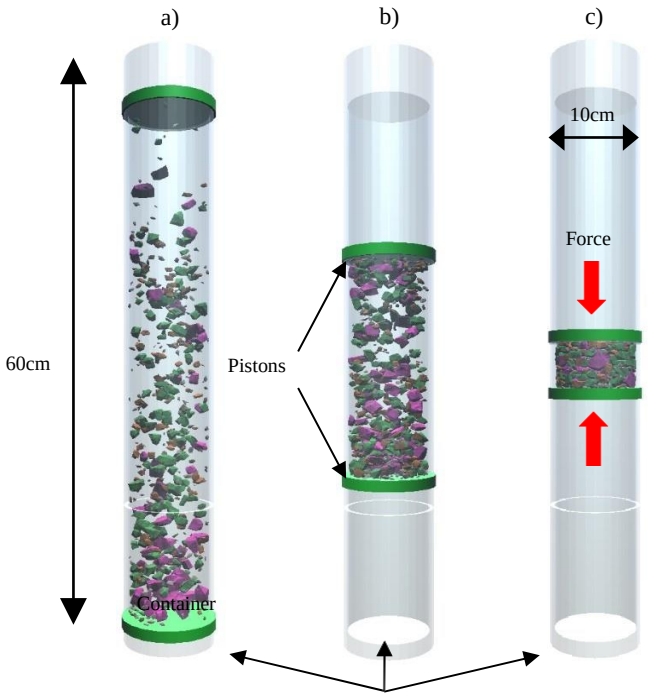


Figure 6-3 – Sequence of the virtual asphalt skeleton compaction process

Note: a) aggregates placed without gravity; b) pistons moving at a constant velocity and c) pistons moving at a progressive force until reaching the maximum pressure.

6.4.1 Input variables and porosity calculation

The inputs necessary for the generation of the different virtual specimens consist of the percentage of each coarse size fraction, the percentage of binder content and the total mass of the samples (see Table 6-3). These data are the same as those used in the

experimental work (see Table 3-9). The same type of aggregates and bitumen were used, and therefore their densities were kept constants for all the specimens.

Due to the limitation of computer processing, it is impossible to generate all the particles that an asphalt mixture can have without the system becoming unstable. For that reason, in this work, only the coarse aggregate fraction will be simulated under the hypothesis that, as explained in Section 3.3.1, porous asphalt is predominantly composed of coarse aggregates, being the main role of the fine to fill the void between the larger particles without producing dilation of the coarse aggregate skeleton.

Table 6-3 –Input to simulate the different porous asphalt mixtures

	Mix 1	Mix 2	Mix 3	Mix 4	Mix 5	Mix 6	Mix 7	Mix 8
	Mass (%)							
20 (mm)	0	0	0	0	0	0	20	45
14 (mm)	0	0	0	0	60	90	20	5
10 (mm)	0	0	75	91	15	0	15	10
6 (mm)	67	85	7	0	5	5	15	25
Dust (4mm)	33	15	18	9	20	5	30	15
Binder Content (%)	6	5.3	5.5	4.3	5	4	4.7	5
Total mass (gr)	1044	926	1052	941	1055	885	1119	939

The calculations to obtain the simulated air void content (AV) of the porous asphalt are based on the volumetric percentage of each range size plus the bitumen/dust mastic in the entire cylindrical sample. Hence, the final prediction formula is:

$$AV = 100\left(\pi\left(\frac{D}{2}\right)^2 H - V_{Coarse\ aggregates} - V_{Dust} - V_{bitumen}\right) \quad 6\text{-}2$$

Where,

D is the diameter of the cylinder (10cm);

H is the is the distance between the pistons at a given time step. When the maximum force is applied, the final H represents the height of the mixtures;

$V_{Coarse\ aggregates}$ is the sum of the volumes of all the coarse aggregates fraction; and

V_{Dust} and $V_{bitumen}$ are the volumes of the fine fraction and the binder content respectively in use in the equivalent real mix.

6.4.2 Determination of the maximum virtual force

Since all the virtual mixtures are compacted under the same pressure condition (as it has been done with the actual one, as detailed in Section 3.3.2); to determine the suitable maximum force of the virtual compaction to achieve the same air void content as the ones reached by SGC, several trial-and-error tests were carried out on sample Mix 7 (see Table 6-3), since is the one that experimentally presented the lowest air void content and the highest locking point (see Table 3-10 and Figure 3-9 respectively).

Various maximum compaction forces were used for the testing by considering the following factors:

- A porosity close to the real one (±2% of error).
- The stability of the virtual aggregate structure.

In this study, the structure of the virtual aggregates is considered stable when their number of contacts reach a relatively anchored value, that is, there are nosignificant increases in contacts between particles, but rather that this changes due to the constant movement of the particles.

As explained in Chapter 5, every virtual aggregate has a mesh collider attached, a component with the same shape as the virtual aggregates, which allow the physics computation when the particles interact. In this study, the meshes of two virtual aggregates have collided when the distance between their meshes is 0.002mm (see Section 5.4.2). When that happens, the software records a contact. Counting the number of existing contacts in a simulation is possible thanks to the algorithm implemented by default in the Unity platform called: *OnCollisionEnter* [233], through its function: *contactCount* [234].

- Ensuring that there is no significant overlap between particles.

Due to a large number of simulated particles and their constant movement during the compaction simulations, it is impossible to avoid 100% penetration betweenparticles since the physics algorithm might not be able to process all the contact pointsbetween particles at each time step, producing thus penetration between particles. Nonetheless, this is not usually appreciated since PE solves the overlapping by calculating a reaction forces or correcting velocities that serve to separate the two objects. Furthermore, if the compaction pressure is too big, it forces the particles to penetrate each other, producing higher degree of overlapping, which might affect the whole particles structure, leading unrealistic asphalt skeletons.

Therefore, the particle's overlapping was calculated using the default script in Unity 3D called *ComputePenetration* [235]. This script computes the distance by which meshes must be displaced to separate two that have become overlapped. More detail can be found at [127]. During the compaction process, the sum of the overlapping distances was calculated. The degree of overlapping allows during the simulation was

chosen based on the impact of the particle's interpenetration on the virtual granular assembly generated.

6.5 Results

6.5.1 Validation of the DE optimisation for the aggregate generation model

For the validation of the DE optimisation implemented in the aggregate generation model, the same statistic tests as those used in Chapter. 4 have been employed (Section 4.4.2). Those are:

1) the ratio, R_{index}, (Eq. 4-4) (the ratio between the mean of the 50th percentile (P_{er50}) from 10 simulations ($\bar{x}$) of each shape´s factor analysed in this study, P_{er50} (μ)); and their percentage of error (Eq 4-5).

2) the statistical index R-squared (R^2) of a linear regression between $\bar{x}$ and μ (Figure 6-4).

The results shown in Table 6-4 reveals that, with the implementation of the DE algorithm, the proposed aggregate generation model performance is statistically more accurate, since the percentage of error has been considerably reduced for most of the cases, compared to the percentage of errors obtained before the implementation of the DE (see Table 4-2).

The R square values plotted in Figure 6-4, also shows a better fit between the experimental and simulated data, although the shape´s factor circularity still has a R^2 value below 0.9 (Figure 6-4.c).

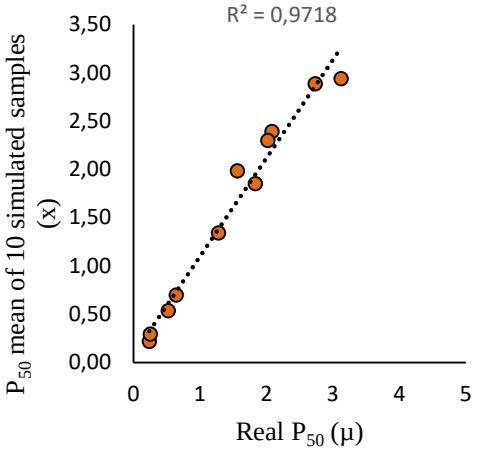
Further, this improvement was also corroborated through 1) a comparison between the cumulative frequency distribution before and after the model optimisation of the

different morphologic features studied in this chapter, as is shown in Appendix D.4.1; and 2) a student t-test of the shape´s factors: circularity and roundness, which is presented in Appendix D.4.2.

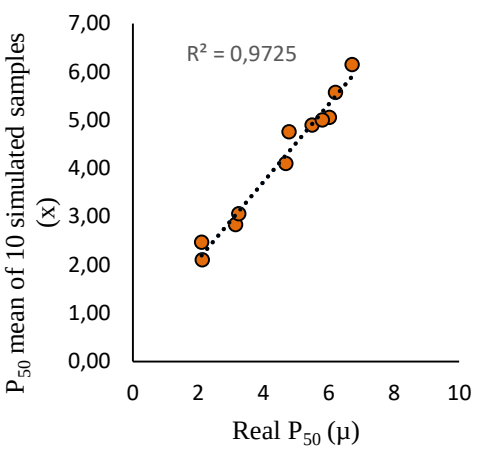
Table 6-4 –R_{index} and percentage of error values resultant after the DE optimization

	G1	**G2**	**RG**	**L1**	**L2**	**L3**	**L4**	**L5**	**GS**	**CG**	**S**
P_{50}	0.838	0.891	0.915	0.860	0.989	0.901	0.874	0.898	0.993	0.941	1.172
% error	16.16	10.91	8.45	13.95	1.128	9.947	12.55	10.25	0.733	5.935	14.6
A_{50}	1.149	1.271	1.057	1.138	0.909	1.028	1.047	0.942	1.012	1.093	1.16
% error	12.78	20.31	5.34	12.05	9.15	2.78	4.48	5.78	1.18	8.46	14.48
R_{50}	0.998	0.994	1.009	1.072	1.042	1.057	1.055	1.062	0.977	1.097	1.082
% error	0.203	0.643	0.874	6.77	4.013	5.382	5.294	5.821	2.486	8.836	7.39
C_{50}	1.060	1.054	1.038	0.950	0.949	0.951	0.946	1.006	0.981	0.973	0.945
% error	5.64	5.03	3.68	5.04	4.85	5.41	3.32	0.86	1.78	2.67	5.91

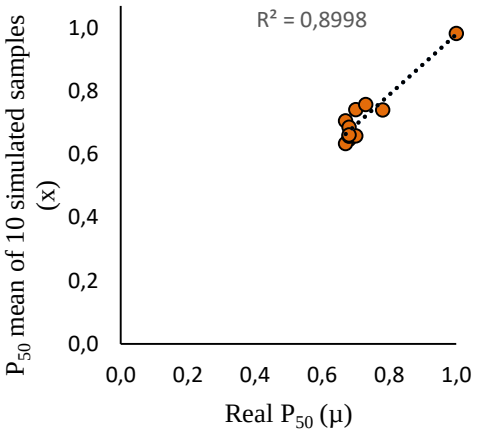
a) Area



b) Perimeter



c) Circularity



d) Roundness

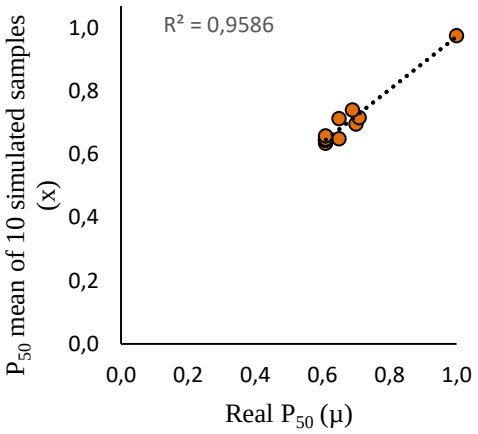


Figure 0-1 –R square of the different shape´s factor after the optimization

6.5.2 Maximum virtual compaction force

To choose an adequate maximum virtual compaction force, the overlapping between particles (mm), air void content (%) and the number of contacts were recorded for the different maximum pressure values. The initial pressure was 50kPa, and this was increased in intervals of 50kPa to achieve a void content close to that obtained in the real sample. The overlapping and the number of contacts values refer to those obtained at the end of the compression test.

A statistical regression analysis was performed using the Minitab Statistical Software to study how the virtual maximum force applied during the compaction process influences in those variables mentioned above. The complete analysis can be seen in Appendix D.5.

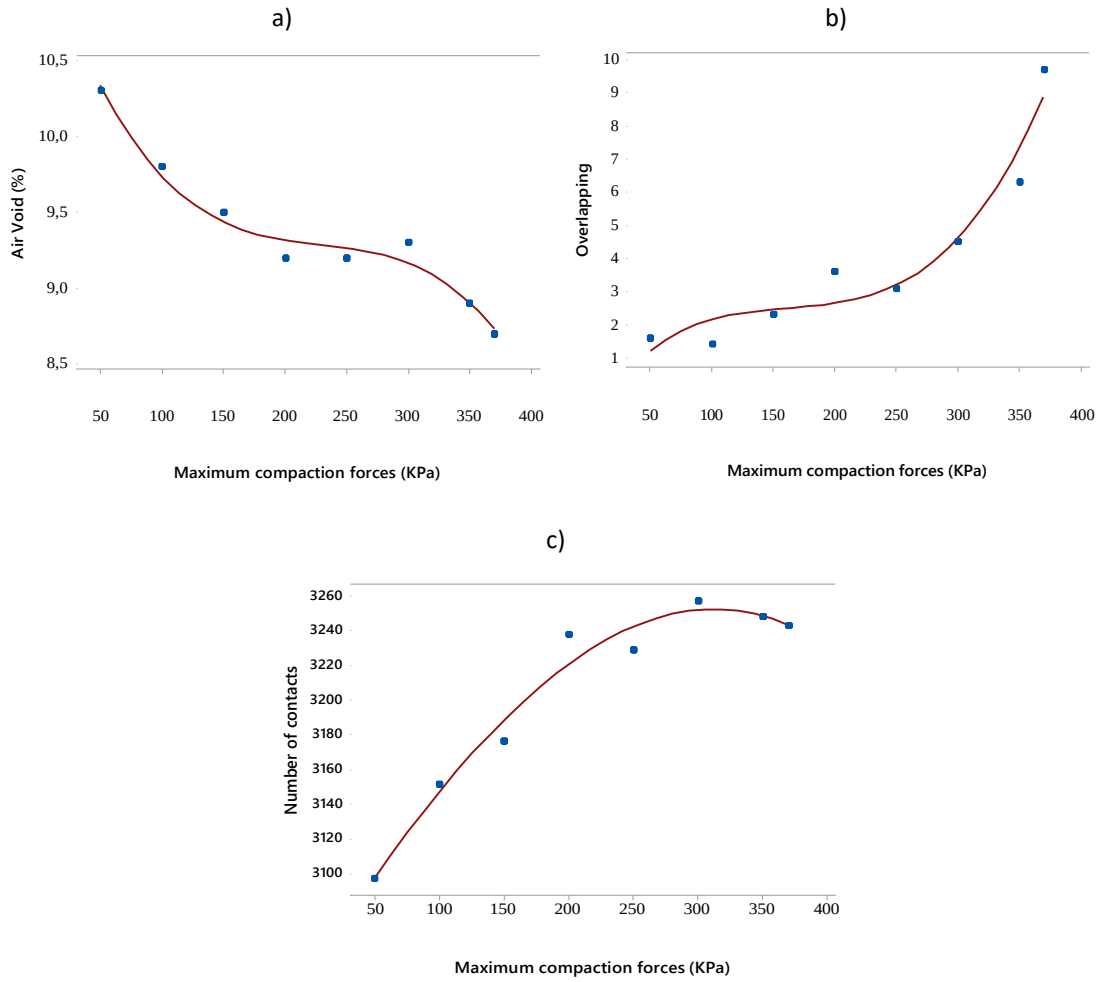


Figure 6-5 –Regression graphs of a) the simulated air void, b) the overlapping,and c) the number of contacts as a function of the virtual compaction force.

Figure 6-5 shows the different polynomial regression graphics of: a) the air void, b) the number of contacts and c) the overlapping as a function of the virtual maximum force applied during the compaction simulation.

It can be observed that the porosity as a function of the maximum force is close to a cubic regression. This could be a clue for future research to study the differences between the Superpave Gyratory Compactor compacting curve, which normally follow a logarithmic curve [272], and the virtual compacting curve, since this topic falls outside the research line of this thesis.

At 50kPa the void content reached is around 10.5%, which is still far from the experimental one (7.52%, see Table 6-6). Between 50kPa and 200kPa, the air void content gradually decreases from 10.5% to 9.2% approximately. Then, between 200kPa and 300kPa the air void content remains more or less constant (staying at ~9.2%).

This might be because the aggregates cannot be further compacted without interparticle penetration. Some cycles are needed by the system to reach equilibrium since after the compressive force increases particles overlap greatly and the physics algorithms attempt to correct it by stronger repulsive force (which prevent of increasing the compaction density). Therefore, when increasing the maximum compactionforce to 350kPa, it can be seen that the air voids content achieves values lower than 9%,although at the expense of a much higher overlapping, as described later in this section.

Figure 6-5.b shows how the particle's interpenetration variation as a function of maximum virtual pressure is also close to a cubic regression model. Between 50kPa and 100kPa, the overlapping between particles does not vary appreciably, having a value close to 1.5mm. From 100kPa, the overlapping grows exponentially to almost 4mm when the maximum virtual force is 200kPa. However, this is not appreciated in

the final aggregate assembly generated. At 250kPa, the overlapping remains very similar as with 200kPa. This may be due to i) that the PE is generating numerous impulses in the opposite direction to the movement of the pistons used for compaction, in order to avoid the overlapping, which agrees with the phenomenon explained in the previous paragraph about the fact that the porosity remains constant for the maximum forces of 200kPa and 300kPa; ii) particles in constant motion causes the measurement of their overlapping to produce ambiguous results difficult to study with certainty.

With forces bigger than 250kPa, the overlapping grows exponentially again. The overlapping becomes very noticeable when they exceed the 9mm beyond 350kPa (see Figure 6-6). Therefore, the samples cannot be further compacted without the structure being affected by overlapping.

Regarding the number of contacts, Figure 6-5.c shows that the number of contacts grows constantly for maximum forces between 50kPa and 200kPa. For forces bigger than 200kPa, very ambiguous results appear, again, this could be due to the constant movement of the particles that make it very difficult to obtain an accurate measurement. Nonetheless, beyond 300kPa it seems that the contact number has stable values between 3240 y 3260, which indicated that the pressure at which the aggregate interlock is stable should be between 300kPa and 400kPa.

Therefore, by considering the effect of the overlapping in the structure, a number of contacts more or less constant, and a porosity value with a ±2% of error from the actual one; a maximum force of 350kPa was chosen to be implemented in the simulated asphalt skeleton compaction.

It must be kept in mind that exist several elements which could affect the accuracy of the software's outcomes such as: the speed of compaction or the number of stones in

the mixture (notice that, even though the trial tests was done on the same sample, due to the stochastic nature of the aggregate generation algorithm, the number of particles can change each time). The influence of these elements needs to be investigated in future research.

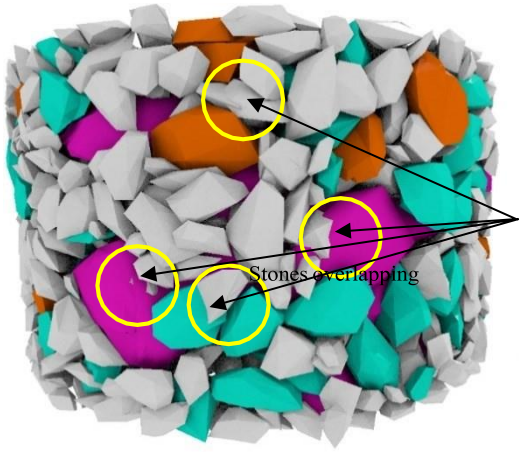


Figure 6-6 – Perceptible penetration between particles in Mix 7 specimen compacted at 370kPa

6.5.3 Porosity prediction model validation. Comparison between experimental and virtual data

6.5.3.1 Repeatability of the asphalt skeleton simulation

To evaluate the repeatability of the proposed and to eliminate the impact of the contingency, each mixture was simulated several times, since for each set of the simulated mixtures, the stones and their positions were different due to the stochastic component of the aggregate generation algorithm (Chapter 4). The inputs and the PE configuration were kept constant during all the tests.

Due to the time the simulation needs, only three compactions of each type of sample were made, since the number of particles in each specimen ranged from 1800 to 6200 and the simulation time varied from 20 minutes up to 4 hours.

Table 6-5 shows the average of three simulations of the computed height (*H*) and porosity (*AV*) for the eight different simulated asphalts skeleton. Notice that one of the mixtures (Mix 1), could not be simulated, the reason could be a failure in the system due to the large number of particles involved, as will be explained in the next section

From Table 6-5 it is observed that the data computed from the replicate testing are very close, since the variance in the computed height is practically zero in all de samples examined, and the biggest variance on the computed porosity is 0.37. Therefore, the ability to generate similar results for multiple preparations of the same sample can be ensured (repeatability).

Therefore, it can be concluded that the simulation tool has less level of error than those shown from the laboratory experiment (see Section 3.3.4).

Table 6-5 –Results of three simulation tests of the computed height (H) and porosity (AV)

Mixtures		Units	Test number				
			1	*2*	*3*	*Mean Value*	*Variance*
Mix 1	*H*	(cm)	-	-	-	-	-
	AV	(%)	-	-	-	-	-
Mix 2	*H*	(cm)	6.02	6.12	6.08	6.07	0.003
	AV	(%)	20.77	21.05	20.93	20.91	0.02
Mix 3	*H*	(cm)	6.31	6.29	6.29	6.29	0
	AV	(%)	14.6	14.48	14.53	14.51	0.004
Mix 4	*H*	(cm)	6.33	6.27	6.32	6.3	0.001
	AV	(%)	26.02	24.97	25.51	25.5	0.276
Mix 5	*H*	(cm)	6.19	6.28	6.23	6.23	0.002
	AV	(%)	13.4	14.6	13.8	13.93	0.373
Mix 6	*H*	(cm)	6.34	6.28	6.19	6.27	0.006
	AV	(%)	30.4	29.9	29.62	29.97	0.156
Mix 7	*H*	(cm)	6.21	6.18	6.19	6.19	0
	AV	(%)	9	8.72	9.3	9	0.084
Mix 8	*H*	(cm)	6.15	6.12	6.18	6.15	0.001
	AV	(%)	22.4	22.12	22.8	22.44	0.117

6.5.3.2 Accuracy of the air void predictions

The predicted porosity values of the tested specimens were plotted against the experimental one, as shown in Figure 6-7. The predicted air void contents are the average of 3 simulations, plotted in Table 6-5. From this figure it can be observed that the points are located surrounding the 1:1 slope line, which indicate a good level of accuracy of the air void prediction model. Nonetheless, despite numerous attempts the algorithm could no compute the mixture Mix 1 due to system crashes. This would be surely due to the large number of stones with complex shapes simulated. Figure 6-7 also shows a lower dispersion in asphalt mixtures with lower porosity.

Therefore, in order to analyse the influence of the variables: number of particles (P_N), volume of mastic, (M_V), and the particle's interpenetration (*overlapping*) on the predicted porosity, those values were computed as plotted in Table 6-6. Besides, the

error of each predicted value (*AV error*), the mean values of the height of the virtual samples ($\overline{H}$) and the mean value of air void content computed ($\overline{AV}$), as well as the mean values of the height of the actual mixtures ($\overline{H}_{exp}$) and the mean of the experimental air void content ($\overline{AV}_{exp}$), are also presented in the same table for the ease of the reader.

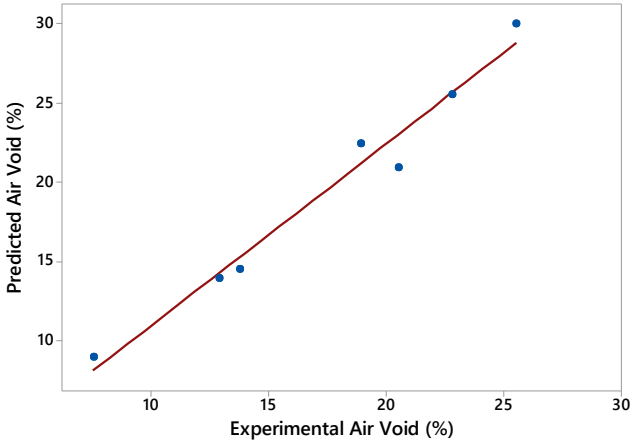


Figure 6-7 – Correlation between predicted and simulated porosity

Table 6-6 –Results of the air void prediction model

Mixtures	$\overline{H}$	$\overline{H}$	$\overline{H}_{exp}$[1] (cm)	$\overline{AV}_{exp}$[2] (%)	*AV error (%)*	P_N[3]	Mv[4] (%)	*Overlapping (mm)*
Mix 1	-	-	6.17	12.4	-	4546	-	-
Mix 2	20.91	6.07	5.96	20.48	0.43	4134	20.21	-
Mix 3	14.51	6.29	6.22	13.75	0.76	3549	24.74	1.4
Mix 4	25.5	6.3	6.1	22.78	2.72	3343	13.87	4.73
Mix 5	13.93	6.23	6.19	12.86	1.07	1987	27.54	5.41
Mix 6	29.97	6.27	5.93	25.5	4.47	2071	15.38	5.87
Mix 7	9	6.19	6.07	7.52	1.48	2264	34.74	6.3
Mix 8	22.44	6.15	5.91	18.9	3.5	2686	19.55	6.8

[1] Mean value of the height of the actual samples compacted with SGC
[2] Mean value of the porosity of the actual samples compacted with SGC
[3] Number of particles simulated
[4] Volume of mastic in the mixture

The relationship between the errors of the model and those independent variables was studied through multiple regressions analysis using Minitab 18 statistical software.

The analysis revealed that the independent variable of overlapping is not helpful explaining the variance of error in the virtual porosity prediction model, while the variable volume of mastic and number of particles have statistical significance on the error. The complete multiple regression analysis can be seen in Appendix D.6.

Figure 6-8 shows that the error in the predicted porosity decreased linearly for larger values of both: volume of mastic and number of particles, being the regression equation expressed as:

$$AV\ error = 9.57 - 0.1665\ volme\ of\ mastic - 0.001323\ number\ of\ particles \quad \text{6-3}$$

A larger number of stones means smaller stone size, which makes the virtual mixtures easier to compact due to the particles can find a stable position between the gaps in an easier way than bigger stone's size, since those tend to overlap until they find their equilibrium position in the assembly.

This phenomenon can be seen if compare Mix 2 and Mix 6 which have a maximum aggregate size of 6mm and 14mm respectively. Mix 2, having the highest number of particles, presents a porosity practically equal to the real one; while Mix 6 presents the highest error (see Table 6-6), since it contains the highest proportion of coarser stones (90% of 14mm), as shown in Table 6-3.

Nonetheless it should point out that Mix 2 also present greater height than the actual one, indicating that the volume of stones generated by the physics engine is

bigger than in the experiment. This could be due to the stochastic nature of the aggregate generation system, or due to some flaw in the Differential Evolution optimisation presented in this work.

On the other hand, when the volume of mastic in the mixtures increase, and the number of particles of two mixtures is similar, means that the virtual particles have higher chance to find a stable position making their compaction easier. An example can be seen if comparing Mix 3 and Mix 4. Both have a similar number of virtual stones, although Mix 3 has about twice the volume of mastic and therefore presents a smaller error.

The same phenomenon occurs when comparing Mix 5 and Mix 6; and Mix 7 and Mix 8 respectively. Both pairs of mixtures contain a similar number of particles but, Mix 5 and Mix 7 have almost twice the volume of mastic than Mix 6 and Mix 8, which lead to aa porosity closer to the actual one.

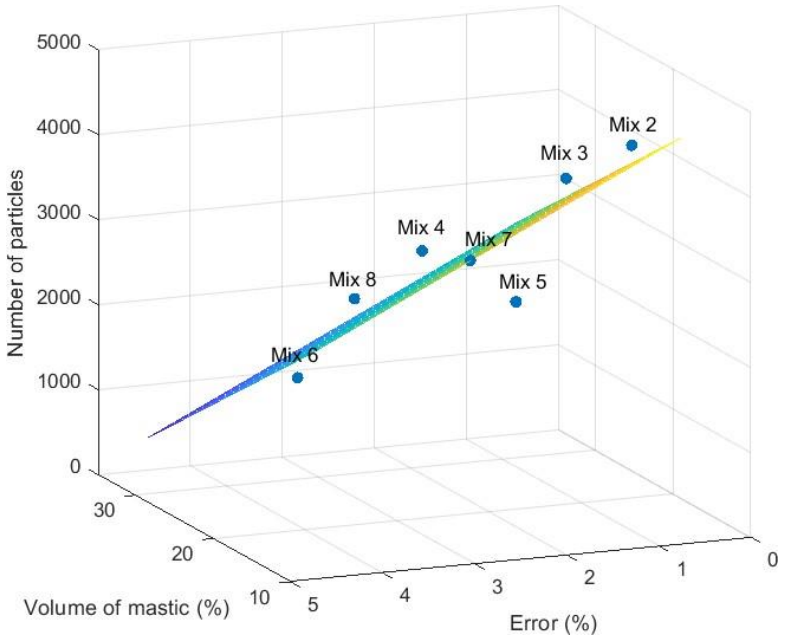


Figure 6-8 –Multiple linear regression analysis response surface of the air voids errors with the number of aggregates and volume of mastic

The R-square and the standard deviation obtained in the regression analysis were 79.82% and 0.84 respectively, Therefore, considering the parameters of goodness of fit the Equation 6-3 represents well the error of the asphalt skeleton simulation under the studied conditions.

Nevertheless, there is still about a 20% of the AV error variation which remains unexplained. This could be partly due to the overlapping between particles, partly due to the stochastic nature of the aggregate generation model; or it could be also due to the angle of rotation was not taken into account. Besides, it could also be attributed to other factors that are still unknown, due to the number of parameters involved in the Physic Engine configuration, as mentioned in Chapter 5.

Besides, to quantify the correlation between the parameters analysed, a Pearson correlation test has been carried out (Appendix D.6.3), showing that the overlapping has a strong inverse correlation with the number of particles. An explanation could be that smaller number of particles means larger stones, which in turn have more contact surfaces. This could cause larger penetration between aggregates when applying the same force.

Besides, the computation of the penetration of the sample: Mix 2, could not be carried out due to the large number of simulated particles, which produces abrupt variations in the overlapping calculation. However, in general, from Table 6-6 it seems that all the virtual mixtures present lower overlapping values than the maximum reasonable value (9mm), found in Section 6.5.2.

The virtual specimens also present a visually acceptable degree of overlapping as can be seen from Figure 6-11, which illustrates the asphalt skeleton structures simulated in this way.

Furthermore, the experimental asphalt core (Mix 3) was scanned using X-ray CT to obtain its air void distribution in vertical direction, to compare it with its equivalent virtual sample thus, providing a preliminary confirmation of the effectiveness of this simulation approach.

The curve corresponding to the real mixture in Figure 6-9 has been obtained by removing the mastic by using a segmentation process through the software ImageJ, following the procedure described in [171]. The virtual specimen was also analysed through the ImageJ, after using the free license software Blender was used to create "virtual tomography" as described in Appendix C.4.

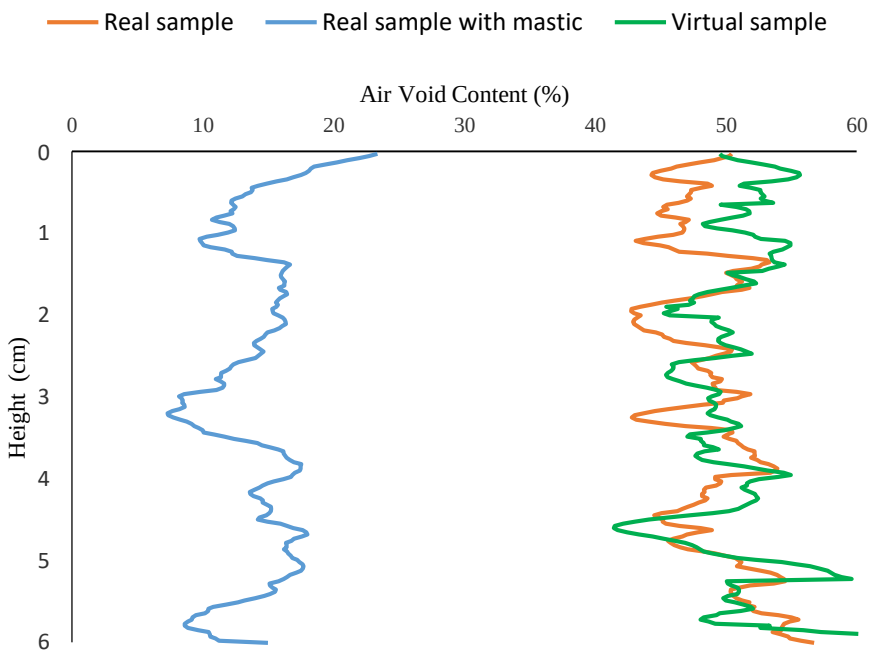


Figure 6-9 –Comparison of the vertical air void distribution between the experimental and simulated Mix 3

From Figure 6-9, it can be observed that the virtual and real air voids contents are numerically very similar: the average porosity in the real mixture was 48.65%, whereas the virtual mixture it was 50.38%. Nonetheless, the virtual curve presents larger "peaks" than the real one; a phenomenon that becomes more pronounced towards the bottom of the sample. This could be due to no simulation of the dust, which might lead to a virtual stone assembly less homogeneous than the actual one. Besides, a segregation of the finer particles towards the bottom of the real sample might occur, causing thus fewer air voids content in this area.

Figure 6-10 shows a comparison between two slices obtained from the above-mentioned asphalt cores. It can be noticed that the virtual mixture (right), presents stone shapes and particle's arrangements similar to the actual one, showing a high level of realism of the simulation approach. Nonetheless, further research is needed to confirm it.

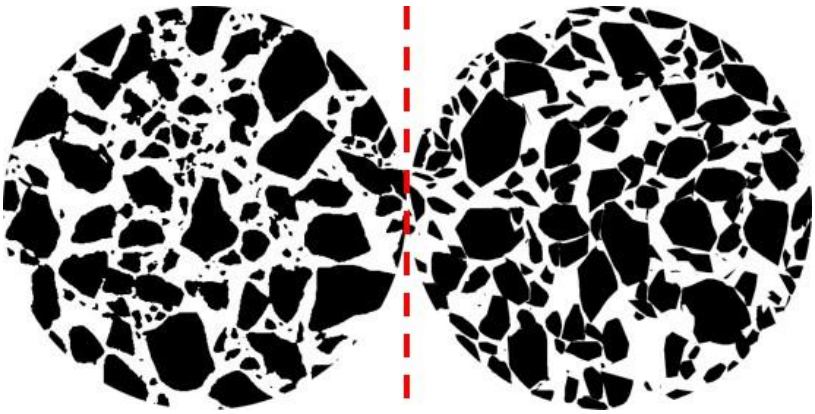


Figure 6-10 – (Left) real asphalt sample of Mix 3 (~13.75% air voids). (Right)real asphalt sample of Mix 3 (~14.5% air voids)

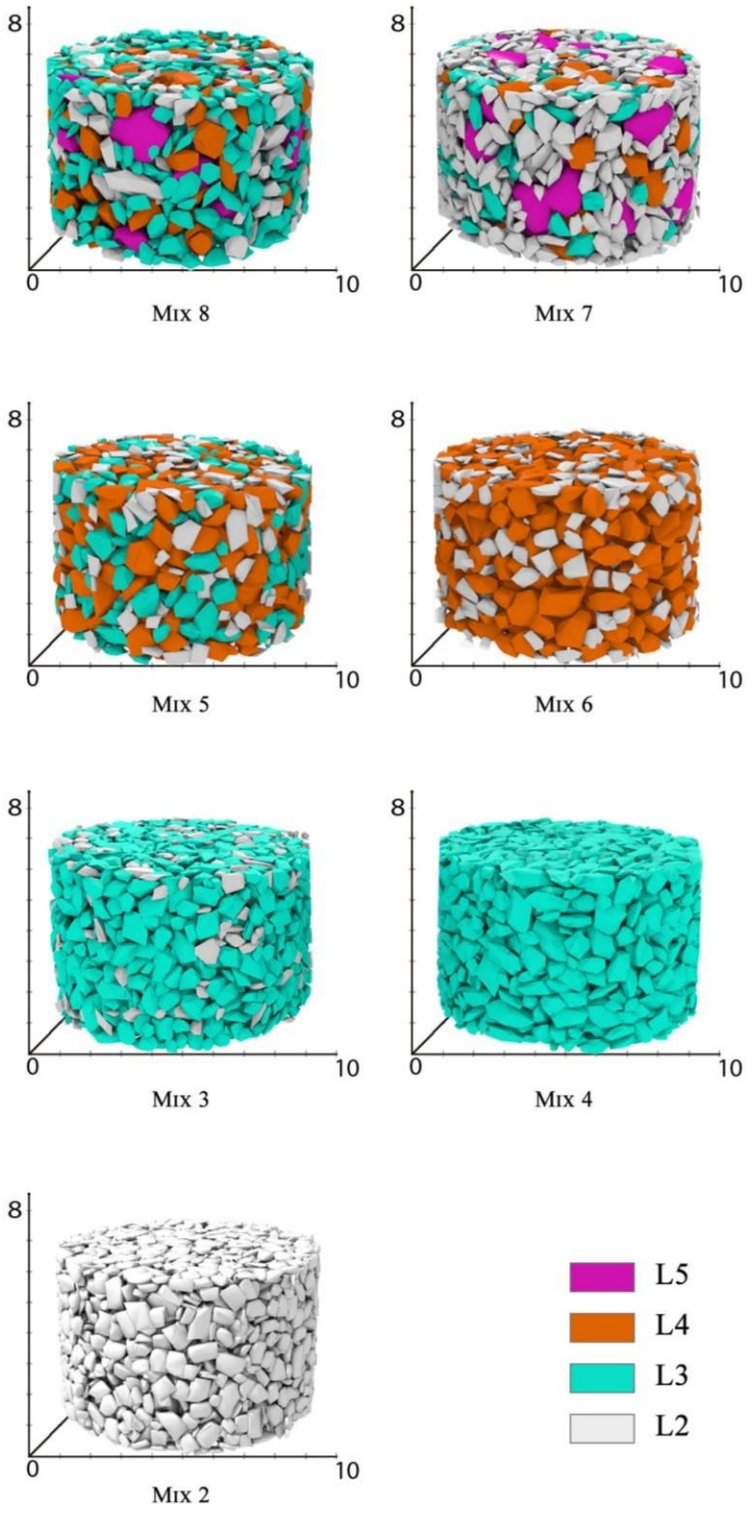


Figure 6-11 –Virtual asphalt simulated

6.5.3.3 Current Limitations

The virtual air void prediction approach tends to give greater height and porosity values, especially in samples with larger contents of coarser aggregates. The multiple regression model suggests that the variables: volume of mastics and number of particles have a big influence on the errors of the model. Nevertheless, there is still about a 20% of the porosity error variation which remains unexplained.

This can be attributed to several factors including:

1. Failures in the virtual aggregate generation model. Although the methodology proposed allows re-creation of full gradations by using the Weibull distribution to decide the Minor Feret and the Aspect Ratio values of the virtual aggregates, removal of extreme stones that have been generated may be necessary, due to the very low probability that they appear in real samples, especially in the fractions of aggregates that include a wide range of particles sizes. Otherwise, the simulation of asphalt skeletons could contain one or two excessively large stones that prevent the correct compaction. Put another way, using a simple statistical distribution as the basis of stone generation may be too simplistic.

   Examples are drawn in Figures B-3 and B-4 (Appendix B.4). In general, it was found that the main difference corresponds to the extremes of the distribution. This can induce errors in the representations of aggregates, such as aggregates that are too elongated, or bigger than the real one.

2. Over compaction of the actual samples. Samples with larger maximum aggregate size seems to be more affected by over-compaction (since they present a porosity value lower than the established target stated (Table 3-9)).

This could be due to the breakage of larger stones which might increase the density of the samples. This stone breakage is not reflected in the simulation and might result in underestimations of the air voids content, in samples with coarser aggregates.

3. Virtual compaction model failures. The overlapping between particles as well as the number of contacts are two "fictitious" concepts which were introduced in the analysis of the virtual asphalt skeleton compaction in order to choose an adequate maximum virtual pressure. A more detailed study about the dependence of the virtual force and the particle inter-penetration and their reorganisation through the jitter or (self-vibration due to the repulsive forces) is needed. Perhaps, higher forces and lower speed of compaction are necessary for the samples with larger stones size, to allow a better particle's reorganisation.

4. The binding effect of the mastic. The binder influence on the packing of the coarse aggregate during the compaction process [232], and its effect is not considered by the model approach presented in this study. Some Physics engines provide the possibility to compute fluids dynamic simulations of fluid-particle systems [38], which can be used in the future to model complete mixtures simulation.

5. The angle of gyration effect. It has also to be taken into account that the degree of compaction increasewith the increase of the angle of gyration, and in this virtual approach the angle of gyration has not been reproduced.

On the other hand, failure in the system appeared during the compaction process of one of the virtual stone mixtures, probably due to the interaction between many complex-shaped particles, making its simulation impossible to perform. It would be

convenient for future research to reduce the size of the simulated samples (reducing their mass and the diameter of the container), to achieve faster and more stable simulations. For this, it would be convenient to study how the size of the mould influences the compaction pressure and the final porosity both: virtually and experimentally.

Another solution could be the implementation of a parallel simulation technique on multi-core processors, since they contribute to increased realism and precision of results, as well as the number of particles to be simulated, without the need for prohibitively expensive hardware [109][238]. This technique could be also be used to undertake the implementation of the actual SGC motion in the proposed approach, in order to study its effect on the orientation and segregation of the particles, increasing thus, the realism and the precision of the results.

6.6 Conclusion

The study reported here provides a preliminary work to build a porosity prediction model of porous asphalts by using a Physics Engine that generates virtual stones with prescribed shape and size distributions by employing a DE optimisation technique. The desired level of compaction was achieved through a uniaxial dynamic compressive loading without excessively complex, nor lengthy, computational resources.

The following conclusions can be drawn:

1. The Differential Evolution employed to optimize the aggregates generation model presented in Chapter 4, can be used to automatically assess the morphological features of the virtual particles in a more

straightforward way and with more accurate results; resulting like this in a more computationally efficient tool, since, instead of the aggregates catalogue with different level of angularity (Figure 4-6), the user can directly enter the morphological data of the different stone´s types to be simulated.

2. Further research should apply the optimisation in the remains shape's factors to minimize the error between the simulated and the experimental morphological features. Also, better knowledge about the control variable for DE it might be necessary for a better practical application.

3. The maximum pressure of the virtual compaction model and the overlapping between particles need further investigation and clarification. Further research could potentially create a correlation parameter between these two variables.Based on the results of the predicted air void content, it can be concluded that, in most cases, the porosities predicted by the physics engine model are very close to the real ones. However, the virtual air void prediction approach tends to give greater height and porosity values, especially in samples with larger contents of coarser aggregates.

4. A regression analysis found a strong relationship between the variables: volume of the mastic on the mixtures and their number of particles which explained 80% of the model failures.

5. Mixtures with higher number of particles are easier to compact as they can find a stable position easier than mixtures with bigger particles.

6. Mixture with more percentage of mastic is easier to compact as they have

higher chance to find an stable position.

It should be pointed out that this model is a prototype and the generation of virtual mixes with a wide range of gradations and types of aggregates, delivering higher number of results, is necessary in order to verify the general applicability of the proposed air void prediction model.

Nevertheless, based on the results it can be concluded that, with further research, the physics engine compaction model could become a useful tool to study the relationship between the volume percentage of each particle size group and the porosity of the mixtures, in this way saving time and material resources needed for laboratory tests.

CHAPTER 7. CONCLUSION AND RECOMMENDATION FOR FUTURE RESEARCH

At the beginning of the project, the main goal of the present work was to build a software able to simulate a wide range of different asphalts Concretes, including the dust and the binder, in order to predict their air void content in an accurate and fast way.

A considerable amount of work was needed to acquire the knowledge and basic skills in several different areas such as, images processing, programming, physics, statistics, before being able to obtain the first realistic virtual aggregate samples and their corresponding assemblies. Additionally, technological limitations, lack of information of how the algorithm of the PhysX engine (and its corresponding plugin) work, and unstable development environments have been a constant during the whole process, especially when trying to incorporate the finest particles (dust), or the binder by incorporating the Unity 3D platform obi fluid plugin [239], into the simulation.

Due to these drawbacks which were impossible to overcome without spending a considerable amount of time and money, it was decided that the initial project was to ambitious and therefore it was limited to simulation of the coarse aggregates in dry condition with the final goal of predicting porous asphalt porosity.

In this chapter, the implications of the results obtained in this study are discussed to see how the research aims, presented in Chapter 1, have been met and how some outstanding questions about the use of the Physics Engine for the simulation and study of granular materials have been answered.

This chapter describes the overall conclusion of the present thesis, and discusses some specific “key” considerations for the improvement of the actual work, to encourage future research to develop further investigation about this topic.

7.1 Introduction

This thesis is based on creating a simulation tool of granular material at mesoscale level, which in the future could serve to reduce cost and time consuming experiments to identify the best aggregates blend design, which is closely related to the real performance of a stone-based mixture.

The PhysX PE, through the free license platform Unity3D, has been chosen due to its advantage, described in Section 5.3., for the different works presented in this thesis, which are:

1) Real time virtual aggregate generation, through a semi stochastic algorithm which models random irregular particle shapes (including concavities), with realistic size and shape. Chapter 4.

2) To compute virtually packed stone assemblies in dry condition, to generate aggregate structures resembling actual materials both visually and numerically. Chapter 5.

3) To compute virtual porous asphalt aggregate skeletons, in order to predict the air void content of the mixtures, based on volumetric calculation. Chapter 6.

In the next section, a summary of the findings of this thesis is drawn, alongside recommendations for their improvement for further investigations.

7.2 Summary of findings and recommendations

7.2.1 3D aggregate generator model with realistic shapes

The aggregate generation model presented in Chapter 4 and its optimization through a Differential Evolution algorithm (DE), presented in Chapter 6 (Section 6.2), shows a novel methodology to re-create different types of particle with a wide range of sizes and shapes (including concavities), able to generates shape indices very similar to those measured on real particle samples, with the Image Analysis software ImageJ (Section 3.1).

This model, built in the Unity 3D platform, allows virtually complete particles samples with realistic shapes to be generated in a matter of seconds, without the need to examine real stone samples (apart of the traditional sieve analysis to know their size distribution), based on a mesh shape library based on the Krumbein and Sloss chart classification for coarse sediments. Inherent to the shape library is the method developed to compute the particles size distribution of the samples, by implementing a semi stochastic algorithm which generate stones samples in a Weibull probabilistic manner since it was found that the real samples typically followed this distribution.

This can lead to the creation of different virtual stone libraries in a more streamlined way than by imaging real specimens, which in addition are very costly in both economical and computational terms, when it comes to simulate a full coarse aggregate gradation of stone-base mixtures, since they incorporate a great amount of data.

Drawbacks and recommendations

- Although the statistical results were accurate, the virtual stones appear to have less complex shape than the real ones. This is due to the fact that properties, such as the surface textures, or the 3D characterization of the stones, were not taken into account. Further investigation is needed in order to implement these shortcomings and thus, improve the proposed model.

- The approach particle modeling through the voxel assembly approach, the Marching Cubes (MC) (see Appendix B.1), presents certain operating errors which leads to the generation of unrealistic particles (Section 4.3.1), due to the intrinsic nature of the MC. This could be solved by implementing more advanced algorithms such as: the Marching tetrahedral [240], or implementing another type of triangulation for the virtual stone mesh generation, able to avoids the generation of unnatural shapes, as that proposed in [241].

This methodology is still in its infancy and certain refinements are necessary. Some ideas could be to employ the Differential Evolution optimization in all the shape factors taken into account for the virtual aggregates generation (see Section 6.2); and/or or the use of more sophisticated computing techniques, such as machine learning [242][243], which could be able to recognize and classify different aggregates morphologies using pictures of stones through a pixel-based classification [244], in order to find the correct search direction of the DE, thus achieving a correct balance between rapid convergence to the global minimum error and virtual stones being as realistic as possible.

7.2.2 The use of the physics engine for the simulation of the particles packing process

In order to clarify if the use of a PE is adequate for conducting granular material simulation at meso-scale level, in Chapter 5 the different virtual stone types used in this work were packed using vibration and a vertical loading, so as to mimic the experimental work described in Section 3.2.1, by using the PhysX engine in the Unity 3D platform.

The results obtained by comparing the virtual aggregate´s packing structure with the actual one, revealed that the PE can generate stone assemblies with realistic features at meso-scale level, such as: porosity, voids size and shape properties, particle size shape properties, particles orientations, and segregation, among others (Section 5.5). Further, the time required for the simulations were, on average, around 30 minutes, which makes this modelling approach a possible flexible and fast tool for the study of different types of granular material, such as asphalt, ballast or concrete.

Drawbacks and recommendations

- The experimental specimens showed a more regular structure (crystallinity) than the simulated one, and some virtual structure descriptor values differed by 30 and 40% from the actual one, in some cases.

- The virtual vibration data and the time of vibration, to which the samples were subjected was set by trial and error. Moreover, the height of the top weight, used to compact the virtual specimens, was monitored and the simulation was stopped manually once it had achieved the same height as the experimental stone packed samples (Section 5.4). It is recommended that future research should study in detail the fluctuation of the settlement of the

virtual packing, with different configurations of amplitude, frequency and time of vibration, and to see how the magnitude differs from the laboratory results, in order to create a simulation tool capable of stopping the compaction process automatically.

- The physical properties of the different computed materials, such as friction and coefficient of restitution, were the same in all cases, and were based on previous investigation. It would be convenient for future research to carry out laboratory procedures to determine the real properties of each material (i.e. pluviation test to obtain the angle of friction of the different granular materials [9]); in order to achieve a more realistic particle interlock behaviour and take into account the effect of the surface roughness of the different grains.

The designed particle packing process simulation is at an early stage and its feasibility still needs to be improved by fine tuning the parameters that control the physical properties of the granular material and the PE variables, in addition to being verified with further experiments, due to the large number of variables involved.

7.2.3 Porosity prediction model for porous asphalt mixtures through a physics engine

In Chapter 6, an air void prediction model approach for porous asphalts was presented, by employing the virtual aggregate generation with complex shapes described in Chapter 4, and using the same Physics Engine as the one employed for the particle assemblies shown in Chapter 5 (the PhysX engine in Unity3D platform), where its ability to compute the particle movements and their interaction in an accurate way was demonstrated.

The comparisons made between the porosity resulting from the experimental work and the virtual approach were very close in most of the cases (Section 6.5.2). However, the virtual model tends to give higher height and porosity values, especially in samples with larger content of coarser aggregates.

Drawbacks and recommendations

- The angle of gyration and the rotational movement of the SGC were not taken into account, to prevent stones from falling through the container (Appendix D.1); this is a phenomenon that could occur due to the large number of particles simulated, which makes the computational performance of the PE decrease and therefore, causes many contact points between objects to be lost when the movement of the container includes rotation. Further research could be undertaken to implement the actual SGC motion in the proposed approach. For this, in addition to investigating the improvement of the configuration of the PEs to achieve a collision system as precisely as possible, parallel simulation techniques on multi-core processors could be employed [120], since these techniques allow improvement in the PE performance, by increasing the realism and precision of results, as well as the amount of particles e simulated, without prohibitive consumer-grade hardware [109][236].

- The virtual pressure used to compact the asphalt skeleton was kept the same for all the samples and it was found by trial-and-error by considering the mixture porosity, the number of contacts and the overlapping between particles. A more in depth investigation about the relationship between those variables, in addition to taking into account the influence of the speed of

compaction or the number of stones in the prediction porosity model outcome, needs to be carried out in future research.

- Additionally, further investigation could be based on the study of the relation between the virtual force required to compact an asphalt concrete and the different inputs involved such as the number of aggregates and the amount of mastic, to extend this prototype to all types of asphalt and not just porous asphalt [245].

- One of the mixtures could not be simulated, probably due to the interaction between many stones with complex shapes, which caused a system collapse. It is recommended that future research reduce the size of the simulated samples (reducing the diameter of the cylindrical container). For this, it would be necessary to study how the size of the mould influences the compaction pressure and the final porosity both: virtually and experimentally. Another solution might be the employment of a parallel simulation technique on multi-core processors, as mentioned previously in this section.

7.3 Final conclusion

Despite the capability of the PE to recreate different stone shapes and their assemblies quite realistically, this work is just a preliminary study; deeper investigation is necessary to understand its correct operation, in order to create an appropriate tool for engineering purposes.

The designed particle packing process simulation is at an early stage and its feasibility still needs to be improved by fine tuning of: 1) the parameters that control

the physical properties of the granular material and 2) the Physics Engine variables and configuration.

Furthermore, the PE offers different ways to calculate collision detection and collision resolution, this being a very complex subject. Also, since the PhysX engine is a closed source, it is not possible to know in detail of how it performs the different tasks, working only with the information provided by Nvidia Corporation. As a consequence, the validation of the developed model has been exploratory, based on trial and error tests in most of the cases, and evaluating their effect on the final results obtained, without taking into account the approximations of the internal calculations that the PE uses to simulate real physics. However, for these particular DEM applications, it can be concluded that the PE can provide reliable results.

CHAPTER 8. POSSIBLE FUTURE APPLICATIONS

In this chapter, a brief description of the possible applications that this modelling approach could have in the future, after overcoming the drawbacks explained in the previous chapter, is presented. Nonetheless, it should be pointed out that their evolution and applicability are conditioned by future technological advances.

8.1 Aggregate´s generator model with realistic shapes

The aggregate generation methodology described in this work allows full gradations to be re-created by using the Weibull distribution to decide the minor Feret and the aspect ratio of the virtual particles, as well as to create different types of stone morphology in a realistic way, including concavities, of commonly used aggregates in civil engineering, by the input of certain parameters.

This offers possibility of changing particle size distribution and its shape in a fast way, thus, being able to create a stone sample library which greatly extends the range, size, and type of particle systems that can be studied, thereby overcoming the limitation of other techniques, such as scanning real stone samples, which are time-consuming, expensive, and therefore not available to everyone.

8.2 The use of a physic engine for the simulation of a particle´s packing process

The stone assembly approach presented in Chapter 5 allows a rapid study of a wide variety of particle assemblies with different stone sizes and shapes. In the future the vibration values could be modified to recreate other compaction effects (i.e. field compaction).

With further development, it could become a potential tool for an in-depth study of how the properties of aggregates affect the structure they form (i.e. how the structure changes with the angularity of the grains [246], or the simulation of indirect methods of measurement of void content, such as the Bailey method, for the gradation selection of mixtures [247]).

This could be useful for various fields dealing with stone-based materials to improve mixture design, saving laboratory time, cost, and better using natural resources, without the need for lengthy and costly laboratory work.

8.3 Porosity prediction model for porous asphalt mixtures

The air void prediction model for porous asphalt shows the potential of a new simulation approach that can be used in the study and optimisation of stone-based material design, providing a starting point for future developments.

This could lead to a useful tool to analyse: i) the relationship between the volume percentage of each particle size group and the porosity of the mixture in a faster way than the empirical method; ii) to do quantitative studies of the distribution of the stone-stone contacts (coordination number) within hot mix asphalt, to achieve a proper load transfer chain in the aggregate structure. In this way, the structural properties of the particle packing could be studied in a more effective way than previous research, which used spherical particles as in [244], through the incorporation of complex shapes in the simulation; iii) the exporting of particle samples or particles assemblies/arrangements into a Finite Element Method software (FEM) or Discrete Element Methods, for the purpose of solving problems involving granular media deformations, which require a high level of numerical accuracy (i.e. simulating the dynamic modulus of mixtures); as has been done by previous research [6].

One the other hand, Physics Engines have software libraries able to compute fluid dynamic simulation, and are able to simulate fluid-particle systems [234], which could be used for example, to implement the effect of the mastic during compaction of the coarse aggregate.

REFERENCES

[1] H. Huang, “Discrete element modeling of railroad ballast using imaging based aggregate morphology characterization,” University of Illinois at, 2010.

[2] R. Isola, “Packing of Granular Materials,” University of Nottingham, 2008.

[3] D. Zhao, E. G. Nezami, Y. M. A. Hashash, and J. Ghaboussi, “Three-dimensional discrete element simulation for granular materials,” *Eng. Comput.*, vol. 23, no. 7, pp. 749–770, 2006, doi: 10.1108/02644400610689884.

[4] J. E. Lane, P. T. Metzger, and R. A. Wilkinson, *A Review of Discrete Element Method (DEM) Particle Shapes and Size Distributions for Lunar Soil*, no. December. 2010.

[5] M. Pytlos, M. Gilbert, and C. C. Smith, “Modelling granular soil behaviour using a physics engine,” *Géotechnique Lett.*, vol. 5, no. 4, pp. 243–249, 2015, doi: 10.1680/jgele.15.00067.

[6] P. Cao, F. Jin, D. Feng, C. Zhou, and W. Hu, “Prediction on dynamic modulus of asphalt concrete with random aggregate modeling methods and virtual physics engine,” *Constr. Build. Mater.*, vol. 125, pp. 987–997, 2016, doi: 10.1016/j.conbuildmat.2016.08.121.

[7] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, “The quickhull algorithm for convex hulls,” *ACM Trans. Math. Softw.*, vol. 22, no. 4, pp. 469–483, 1996, doi: 10.1145/235815.235821.

[8] S. Paltani, “Monte Carlo Methods,” *Stat. Course Astrophys. 2010–2011*, pp. 2010–2011, 2011, doi: 10.1002/9781118445112.stat03876.pub2.

[9] E. Izadi and A. Bezuijen, “Simulation of granular soil behaviour using the Bullet physics library,” *Geomech. from Micro to Macro*, pp. 1565–1570, 2013, [Online]. Available: http://www.crcpress.com/product/isbn/9781138027077.

[10] E. G. Kalaitzis, “Performance analysis of a morphological voronoi tessellation algorithm,” pp. 201–202, 1994.

[11] F. Zhu and J. Zhao, “Modeling continuous grain crushing in granular media: A hybrid peridynamics and physics engine approach,” *Comput. Methods Appl.*

Mech. Eng., vol. 348, pp. 334–355, 2019, doi: 10.1016/j.cma.2019.01.017.

[12] E. Izadi and A. Bezuijen, "Simulating direct shear tests with the Bullet physics library: A validation study," *PLoS One*, vol. 13, no. 4, pp. 1–28, 2018, doi: 10.1371/journal.pone.0195073.

[13] H. He, J. Zheng, Q. Sun, and Z. Li, "Simulation of realistic particles with bullet physics engine," *E3S Web Conf.*, vol. 92, pp. 1–5, 2019, doi: 10.1051/e3sconf/20199214004.

[14] British Standard Institute (BSI), "BS EN 13043:2013, Aggregates for bituminous mixtures and surface treatments for roads , airfields and other trafficked areas," no. November, 2013.

[15] A. C. E-701, "Aggregates for concrete," *Concr.*, vol. 32, no. 5, pp. 12–14, 1998, doi: 10.4324/9780203478981.ch16.

[16] The Concrete Countertop institute, "The role of aggregate in concrete," 2011. http://www.concretecountertopinstitute.com/blog/2011/03/the-role-of-aggregate-in-concrete-countertop-mix-formulas/.

[17] H. P. Sooriyaarachchi and E. D. L. Lasintha, "Influence of Fine Aggregate Types on the Performance of Self - Compacting Concrete," *Eng. J. Inst. Eng. Sri Lanka*, vol. 49, no. 1, p. 9, 2016, doi: 10.4038/engineer.v49i1.6914.

[18] F. T. S. Aragão, A. R. G. Pazos, L. M. G. da Motta, Y. R. Kim, and L. A. H. do Nascimento, "Effects of morphological characteristics of aggregate particles on the mechanical behavior of bituminous paving mixtures," *Constr. Build. Mater.*, vol. 123, no. October, pp. 444–453, 2016, doi: 10.1016/j.conbuildmat.2016.07.013.

[19] H. Liu, P. Hao, and J. Xu, "Effects of Nominal Maximum Aggregate Size on the Performance of Stone Matrix Asphalt," *Appl. Sci.*, vol. 7, no. 2, p. 126, 2017, doi: 10.3390/app7020126.

[20] X. Bian, H. Huang, E. Tutumluer, and Y. Gao, "'Critical particle size' and ballast gradation studied by Discrete Element Modeling," *Transp. Geotech.*, vol. 6, no. January, pp. 38–44, 2016, doi: 10.1016/j.trgeo.2016.01.002.

[21] Pavement Interactive, "Gradation and Size." https://pavementinteractive.org/reference-desk/materials/aggregate/gradation-and-size/#:~:text=Nominal maximum size.,.%2C 1996%7B%7B1%7D%7D).

[22] B. D. Prowell, “Permeability characteristics of coarse- graded superpave mixes,” *Natl. Cent. Asph. Technol.*, vol. 277, no. January, 2002.

[23] A. Setiawan, L. B. Suparma, and A. T. Mulyono, “The Effect of Aggregate Gradation on Workability of Asphalt Concrete,” no. August, 2016, doi: 10.21817/ijet/2016/v8i4/160804042.

[24] O. VO and A. EE, “Effects of Aggregate Gradation on the Properties of Concrete Made From Granite Chippings,” *J. Steel Struct. Constr.*, vol. 2, no. 1, pp. 16–20, 2016, doi: 10.4172/2472-0437.1000109.

[25] B. Huang, X. Chen, X. Shu, E. Masad, and E. Mahmoud, “Effects of coarse aggregate angularity and asphalt binder on laboratory-measured permanent deformation properties of HMA,” *Int. J. Pavement Eng.*, vol. 10, no. 1, pp. 19–28, 2009, doi: 10.1080/10298430802068915.

[26] R. Polat, M. M. Yadollahi, A. E. Sagsoz, and S. Arasan, “The Correlation between Aggregate Shape and Compressive Strength of Concrete: Digital Image Processing Approach,” *Int. J. Struct. Civ. Engg. Res.ijscer.com*, vol. 2, no. 3, pp. 63–80, 2013.

[27] Y. Sun, “Effect of particle angularity and size distribution on the deformation and degradation of ballast under cyclic loading,” pp. 1–237, 2017, [Online]. Available: http://ro.uow.edu.au/theses1/9.

[28] C. F. Mora and A. K. H. Kwan, “Sphericity, shape factor, and convexity measurement of coarse aggregate for concrete using digital image processing,” *Cem. Concr. Res.*, vol. 30, no. 3, pp. 351–358, 2000, doi: 10.1016/S0008-8846(99)00259-8.

[29] M. Hachani Ibrahim, “Experimental study and comparison between the use of natural and cruhed,” *Energy Procedia*, vol. 119, 2017.

[30] Building Technology, “Classification of Aggregates as per Size and Shape - Coarse and Fine Aggregates,” 2017. https://theconstructor.org/building/classification-of-aggregates-size-shape/12339/.

[31] Highway Engineering Government of Indiana., “Properties Considered in Mix Design.” http://www.in.gov/indot/files/Chapter_04(4).pdf.

[32] T. M. Massaro, “Characterization of asphalt concrete using anisotropic damage viscoelastic-viscoplastic model,” *J. Chem. Inf. Model.*, vol. 12 Suppl 1, no. 9, pp.

1–29, 2005, doi: 10.1007/978-1-4614-7990-1.

[33] Australian Standards, “Aggregate for Railway Ballast The requirements of AS 2758.7,” 2009.

[34] M. Z. H. Mahmud, H. Yaacob, R. P. Jaya, and N. A. Hassan, “Laboratory investigation on the effects of flaky aggregates on dynamic creep and resilient modulus of asphalt mixtures,” *J. Teknol.*, vol. 70, no. 4, pp. 107–110, 2014, doi: 10.11113/jt.v70.3498.

[35] ASTM D 4791, *Flat Particles, Elongated Particles, or Flat and Elongated Particles in Coarse Aggregate*, vol. 04. 1999.

[36] I. Kamaruddin, A. Soelistijo, and M. Napiah, “A comparison of the engineering properties of gap-graded and well-graded road bituminous mixtures,” *WIT Trans. Built Environ.*, vol. 111, pp. 275–287, 2010, doi: 10.2495/UT100251.

[37] C. Engineering, S. Grade, S. V. P. College, and M. P. Bhopal, “Effect of Shape and Size of Aggregate on Permeability,” vol. 2386, no. Part I, 2012.

[38] S. Tafesse, J. M. Robison Fernlund, W. Sun, and F. Bergholm, “Evaluation of image analysis methods used for quantification of particle angularity,” *Sedimentology*, vol. 60, no. 4, pp. 1100–1110, 2013, doi: 10.1111/j.1365-3091.2012.01367.x.

[39] ASTM, “Standard practice for description and identification of soils (visual manual procedure),” vol. i, no. D 2488-09a, p. 11, 2009, doi: 10.1520/D2488-09A.2.

[40] “Standard Test Method for Index of Aggregate Particle Shape and Texture 1,” vol. 00, no. Reapproved 2006, pp. 3–6, 2011, doi: 10.1520/D3398-00R06.2.

[41] I. Karaman, “Characterization of Aggregate Resistance To Degradation in Stone Matrix Asphalt Mixtures,” no. December, 2005, [Online]. Available: http://oaktrust.library.tamu.edu/bitstream/handle/1969.1/3342/etd-tamu-2005C-CVEN-Gatchal.pdf?sequence=1.

[42] Y. Descantes, Y. Fosse, and G. Ehret, “Automated measurement of aggregate properties: Part 1 - Crushed and broken surfaces in coarse aggregate particles,” *Mater. Struct. Constr.*, vol. 39, no. 285, pp. 3–12, 2006, doi: 10.1617/s11527-005-9001-z.

[43] S. S. Jamkar and C. B. K. Rao, “Index of Aggregate Particle Shape and Texture of coarse aggregate as a parameter for concrete mix proportioning,” *Cem. Concr. Res.*, vol. 34, no. 11, pp. 2021–2027, 2004, doi: 10.1016/j.cemconres.2004.03.010.

[44] ASTM D 5821-13, “Standard Test Method for determining the percentage of fractured particles in coarse aggregate,” *Annu. B. Am. Soc. Test. Mater. ASTM Stand.*, no. 4, pp. 1–6, 2014, doi: 10.1520/D5821-13.2.

[45] B. E. Standard, “BSI Standards Publication Tests for geometrical properties of aggregates Part 3 : Determination of particle shape — Flakiness index,” *Br. Eur. Stand.*, 2012.

[46] E. Masad and T. Fletcher., “Aggregate imaging system (AIMS): basics and application,” *Texas Transp. Inst.*, vol. Report 5-1, no. 2, 2005.

[47] D. Su and W. M. Yan, “Quantification of angularity of general-shape particles by using Fourier series and a gradient-based approach,” *Constr. Build. Mater.*, vol. 161, pp. 547–554, 2018, doi: 10.1016/j.conbuildmat.2017.12.004.

[48] W. Dahmen, “Wavelet methods for PDEs - Some recent developments,” *J. Comput. Appl. Math.*, vol. 128, no. 1–2, pp. 133–185, 2001, doi: 10.1016/S0377-0427(00)00511-2.

[49] and D. B. Leslie Gates, Eyad Masad, Roger Pyle, “Aggregate Image Measurement System (AIMS 2): Final Report,” vol. 2, no. January, pp. 1652–1664, 2011.

[50] M. Moaveni, S. Wang, J. Hart, E. Tutumluer, and N. Ahuja, “Evaluation of aggregate size and shape by means of segmentation techniques and aggregate image processing algorithms,” *Transp. Res. Rec.*, no. 2335, pp. 50–59, 2013, doi: 10.3141/2335-06.

[51] W. K. Pratt, “Image Segmentation,” *Comput. Vis.*, pp. 579–622, 2000, doi: 10.1002/9780470097434.ch17.

[52] T. M. Al Rousan, “Characterization of aggregate shape properties using a computer automated system,” no. December, p. 211, 2004, [Online]. Available: http://txspace.di.tamu.edu/bitstream/handle/1969.1/1485/etd-tamu-2004C-ENGR-AL.pdf?sequence=1.

[53] J. M. R. Fernlund, “Image analysis method for determining 3-D shape of coarse aggregate,” *Cem. Concr. Res.*, vol. 35, no. 8, pp. 1629–1637, 2005, doi:

10.1016/j.cemconres.2004.11.017.

[54] T. Solomon, “A new image analysis technique to quantify particle angularity,” Virginia Tech, College of Engineering, 2002.

[55] G. A. Swift, “Characterization of coarse aggregate angularity using digital image processing,” University of Missouri-ROLLA In, 2007.

[56] Norbert H. Maerz and Steven Michael Lusher, “Measurement of flat and elongation of coarse aggregate using digital image processing,” *Transp. Res. Board 80th Annu. Meet.*, no. January, pp. 2–14, 2001.

[57] E. N. C. H. R. P. Masad, *Test Methods for Characterizing Aggregate Shape, Texture, and Angularity*. 2007.

[58] Retsh Technology, “Camsizer,” 2018. https://www.retsch-technology.es.

[59] T. R. HORIBA, “CAMSIZER XT 3-D Printing,” *Technology*, 2017, [Online]. Available: https://www.slideshare.net/HORIBA/understanding-your-particle-size-analyzer-results.

[60] R. Technology, “Particle Analyzer CAMSIZER ®,” *Horiba Sci.*, pp. 1–16, 2012, [Online]. Available: https://static.horiba.com/fileadmin/Horiba/Products/Scientific/Particle_Characterization/CAMSIZER/CAMSIZER_P4_Brochure.pdf.

[61] F. Lanaro and P. Tolppanen, “3D characterization of coarse aggregates,” *Eng. Geol.*, vol. 65, no. 1, pp. 17–30, 2002, doi: 10.1016/S0013-7952(01)00133-8.

[62] C. L. Lin and J. D. Miller, “3D characterization and analysis of particle shape using X-ray microtomography (XMT),” *Powder Technol.*, vol. 154, no. 1, pp. 61–69, 2005, doi: 10.1016/j.powtec.2005.04.031.

[63] B. Zhao and J. Wang, “3D quantitative shape analysis on form, roundness, and compactness with μCT,” *Powder Technol.*, vol. 291, pp. 262–275, 2016, doi: 10.1016/j.powtec.2015.12.029.

[64] X. Yang, S. Chen, and Z. You, “3D Voxel-Based Approach to Quantify Aggregate Angularity and Surface Texture,” *J. Mater. Civ. Eng.*, vol. 29, no. 7, p. 04017031, 2017, doi: 10.1061/(ASCE)MT.1943-5533.0001872.

[65] H. Kim, C. Haas, A. R.- Proc., 17th Int. Symp. on, and undefined 2000, "'Innovative system for scanning construction aggregates using laser profiling," *Researchgate.Net*, no. February, 2014, [Online]. Available: https://www.researchgate.net/profile/Craig_Browne/publication/228902355_'Innovative_system_for_scanning_construction_aggregates_using_laser_profiling/links/02e7e52f3b1898f4ec000000.pdf.

[66] Y. Liu and H. L. Xing, "Surface mesh generation of large-scale digital rock images in 3D," *Procedia Computer Science*, vol. 18. pp. 1208–1216, 2013, doi: 10.1016/j.procs.2013.05.287.

[67] T. S. Newman and H. Yi, "A survey of the marching cubes algorithm," *Comput. Graph.*, vol. 30, no. 5, pp. 854–879, 2006, doi: 10.1016/j.cag.2006.07.021.

[68] Q. Du, V. Faber, and M. Gunzburger, "Centroidal Voronoi tessellations: Applications and algorithms," *SIAM Rev.*, vol. 41, no. 4, pp. 637–676, 1999, doi: 10.1137/S0036144599352836.

[69] C. Jin, X. Yang, Z. You, and K. Liu, "Aggregate Shape Characterization Using Virtual Measurement of Three-Dimensional Solid Models Constructed from X-Ray CT Images of Aggregates," *J. Mater. Civ. Eng.*, vol. 30, no. 3, p. 04018026, 2018, doi: 10.1061/(asce)mt.1943-5533.0002210.

[70] E. J. Garboczi, G. S. Cheok, and W. C. Stone, "Using LADAR to characterize the 3-D shape of aggregates: Preliminary results," *Cem. Concr. Res.*, vol. 36, no. 6, pp. 1072–1075, 2006, doi: 10.1016/j.cemconres.2006.03.017.

[71] J. Schindelin, "Fiji: an open-source platform for biological-image analysis"," *Nature methods*, 2012. https://imagej.net/ImageJ.

[72] H. G. Merkus, "Particle Size Measurements: Fundamentals, Practice, Quality," Pijnacker, 2009, pp. 30–34.

[73] P. Smart, "Particle Shape Statistics Revised," *Glas. Univ.*, 2013, [Online]. Available: http://userweb.eng.gla.ac.uk/peter.smart/ima/13C11.pdf.

[74] B. W. Analyze, "Introduction to Molecular Biology Techniques Workshop 15C Image Analysis Part I : Image basics , densitometry and particle analysis."

[75] E. Profitis, D. Kapatos, E. Chatzitheodoridis, D. Xirouchakis, and C. Loupasakis, "Digital methods for measuring grain size parameter of Aggregate – Binder Mixtures," vol. XLVII, no. 3, pp. 1834–1843, 2013.

[76] F. N. Altuhafi and M. R. Coop, “Changes to particle characteristics associated with the compression of sands,” *Geotechnique*, vol. 61, no. 6, pp. 459–471, 2011, doi: 10.1680/geot.9.P.114.

[77] D. Psimadas, P. Georgoulias, V. Valotassiou, and G. Loudos, “Molecular Nanomedicine Towards Cancer :,” *J. Pharm. Sci.*, vol. 101, no. 7, pp. 2271–2280, 2012, doi: 10.1002/jps.

[78] Sympatec GmbH, “Particle Shape Analysis,” 2017. https://www.sympatec.com/en/particle-measurement/glossary/particle-shape/.

[79] Retsch Technology, “CAMSIZER® Characteristics,” p. 8, 2009.

[80] S. Ribeiro and C. Bonetti, “Variabilidade morfométrica de sedimentos arenosos: revisão de métodos e uso do software ImageJ na diferenciação de ambientes deposicionais na Ilha de Santa Catarina e região continental, SC, Brasil,” *Gravel*, vol. 11, no. 1, pp. 37–47, 2013.

[81] Y. Takashimizu and M. Iiyoshi, “New parameter of roundness R: circularity corrected by aspect ratio,” *Prog. Earth Planet. Sci.*, vol. 3, no. 1, 2016, doi: 10.1186/s40645-015-0078-x.

[82] J.-C. Pinoli, “Mathematical Foundations of Image Processing and Analysis, Volumen 2,” in *Volume 2*, .

[83] T. Ferreira and W. S. Rasband, “ImageJ User Guide IJ 1.46r,” *IJ 1.46r*, p. 185, 2012, doi: 10.1038/nmeth.2019.

[84] J. Rodriguez and T. Edeskär, “Case of study on particle shape and friction angle on tailings,” *J. Adv. Sci. Eng. Res.*, vol. 3, no. 4, pp. 373–387, 2013, doi: 10.1016/j.eurpolymj.2004.03.025.

[85] D. Frank and et al, *Concrete Repair, Rehabilitation and Retrofitting IV*. Leipzig, Germany, 2015.

[86] U. Ulusoy, “Quantifying of particle shape differences of differently milled barite using a novel technique: Dynamic image analysis,” *Materialia*, vol. 8, no. August, 2019, doi: 10.1016/j.mtla.2019.100434.

[87] Y. Miao, X. Liu, Y. Hou, J. Li, J. Wu, and L. Wang, “Packing characteristics of aggregate with consideration of particle size and morphology,” *Appl. Sci.*, vol. 9, no. 5, 2019, doi: 10.3390/app9050869.

[88] S. Katuwal, T. Norgaard, P. Moldrup, M. Lamandé, D. Wildenschild, and L. W. de Jonge, “Linking air and water transport in intact soils to macropore characteristics inferred from X-ray computed tomography,” *Geoderma*, vol. 237–238, pp. 9–20, 2015, doi: 10.1016/j.geoderma.2014.08.006.

[89] M. A. A. Garcia, “Aboufoul , M . and Garcia , Alvaro (2017) Factors affecting hydraulic conductivity of asphalt mixture . Materials and,” *Mater. Struct.*, vol. 50, pp. 1–16, 2017, doi: 10.1617/s11527-016-0982-6.

[90] A. Nemati Hayati, M. M. Ahmadi, and S. Mohammadi, “How particle shape affects the flow through granular materials,” *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.*, vol. 85, no. 3, pp. 2–6, 2012, doi: 10.1103/PhysRevE.85.036310.

[91] A. Tatomir, “From of Flow in Fractured Porous Media Sediment dynamics in the floodplain,” *Inst. für Wasser- und Umweltsystemmodellierung der Univ. Stuttgart*, no. July, 2015.

[92] M. M. Ahmadi, S. Mohammadi, and A. N. Hayati, “Analytical derivation of tortuosity and permeability of monosized spheres: A volume averaging approach,” *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.*, vol. 83, no. 2, 2011, doi: 10.1103/PhysRevE.83.026312.

[93] R. P. Zou and A. B. Yu, “Evaluation of the packing characteristics of mono-sized non-spherical particles,” *Powder Technol.*, vol. 88, no. 1, pp. 71–79, 1996, doi: 10.1016/0032-5910(96)03106-3.

[94] W. Sobieski and S. Lipiński, “The influence of particle size distribution on parameters characterizing the spatial structure of porous beds,” *Granul. Matter*, vol. 21, no. 2, pp. 1–15, 2019, doi: 10.1007/s10035-019-0866-x.

[95] B. Soltanbeigi, “Multi-scale analysis of the influence of particle shape on the mechanical response of granular materials,” The University of Edinburgh, 2019.

[96] B. Soltanbeigi, A. Podlozhnyuk, J. Y. Ooi, C. Kloss, and S. A. Papanicolopulos, “Comparison of multi-sphere and superquadric particle representation for modelling shearing and flow characteristics of granular assemblies,” *EPJ Web Conf.*, vol. 140, no. June, 2017, doi: 10.1051/epjconf/201714006015.

[97] J. A. Epaarachchi, *The effect of viscoelasticity on fatigue behaviour of polymer matrix composites*. Woodhead Publishing Limited, 2010.

[98] D. Bao, C. Jiaqi, H. Wang, H. Liu, and J. Zhang, “Characterizing Skeleton Structure and Stacking Properties of Continuous and Gap Graded Aggregate

Mixtures," *Adv. Civ. Eng.*, vol. 2019, 2019, doi: 10.1155/2019/4903602.

[99] S. Shen and H. Yu, "Characterize packing of aggregate particles for paving materials : Particle size impact," *Constr. Build. Mater.*, vol. 25, no. 3, pp. 1362–1368, 2011, doi: 10.1016/j.conbuildmat.2010.09.008.

[100] X. Ding, T. Ma, and W. Gao, "Morphological characterization and mechanical analysis for coarse aggregate skeleton of asphalt mixture based on discrete-element modeling," *Constr. Build. Mater.*, vol. 154, pp. 1048–1061, 2017, doi: 10.1016/j.conbuildmat.2017.08.008.

[101] Y. Liu, X. Zhou, Z. You, S. Yao, F. Gong, and H. Wang, "Discrete element modeling of realistic particle shapes in stone-based mixtures through MATLAB-based imaging process," *Constr. Build. Mater.*, vol. 143, pp. 169–178, 2017, doi: 10.1016/j.conbuildmat.2017.03.037.

[102] X. Zhou, Y. Liu, and Z. You, "Discrete Element Modeling for Sieve Analysis with Image-based Realistic Aggregates," no. Ictim, pp. 916–923, 2016.

[103] J. Eliáš, "Simulation of railway ballast using crushable polyhedral particles," *Powder Technol.*, vol. 264, pp. 458–465, 2014, doi: 10.1016/j.powtec.2014.05.052.

[104] S. . Lee, "Developments in large scale discrete element simulations with polyhedral particles," 2014.

[105] A. Chiarelli, A. R. Dawson, and A. García, "Computational Packing of Aggregates for the Study Of Virtual Asphalt Samples," *ISAP 2016 Symp.*, no. July, pp. 1–12, 2016.

[106] S. M. Longshaw, M. J. Turner, E. Finch, and R. Gawthorpe, "Analysing the use of Real-time Physics Engines for Scientific Simulation: Exploring the Theoretical and Practical Benefits for Discrete Element Modelling," *Acme2010, 18Th ...*, no. March, pp. 18–21, 2010, doi: 10.13140/2.1.3212.7048.

[107] W. Gerstle, N. Sau, and S. Silling, "Peridynamic modeling of concrete structures," *Nucl. Eng. Des.*, vol. 237, no. 12–13, pp. 1250–1258, 2007, doi: 10.1016/j.nucengdes.2006.10.002.

[108] T. Washizawa and Y. Nakahara, "Parallel Computing of Discrete Element Method on GPU," *arXiv Prepr. arXiv1301.1714*, 2013, [Online]. Available: http://arxiv.org/abs/1301.1714.

[109] A. Asaduzzaman and H. Y. Lee, “GPU computing to improve game engine performance,” *J. Eng. Technol. Sci.*, vol. 46 B, no. 2, pp. 226–243, 2014, doi: 10.5614/j.eng.technol.sci.2014.46.2.8.

[110] M. A. Al Ibrahim, A. Kerimov, T. Mukerji, and G. Mavko, “Particula: A simulator tool for computational rock physics of granular media,” *Geophysics*, vol. 84, no. 3, pp. F85–F95, 2019, doi: 10.1190/geo2018-0481.1.

[111] A. Lagae *et al.*, “A survey of procedural noise functions,” *Comput. Graph. Forum*, vol. 29, no. 8, pp. 2579–2600, 2010, doi: 10.1111/j.1467-8659.2010.01827.x.

[112] B. Ghanbarian, A. G. Hunt, R. P. Ewing, and M. Sahimi, “Tortuosity in Porous Media: A Critical Review,” *Soil Sci. Soc. Am. J.*, vol. 77, no. 5, pp. 1461–1477, 2013, doi: 10.2136/sssaj2012.0435.

[113] H. J. Vogel, “A numerical experiment on pore size, pore connectivity, water retention, permeability, and solute transport using network models,” *Eur. J. Soil Sci.*, vol. 51, no. 1, pp. 99–105, 2000, doi: 10.1046/j.1365-2389.2000.00275.x.

[114] M. Larsbo, J. Koestel, and N. Jarvis, “Relations between macropore network characteristics and the degree of preferential solute transport,” *Hydrol. Earth Syst. Sci.*, vol. 18, no. 12, pp. 5255–5269, 2014, doi: 10.5194/hess-18-5255-2014.

[115] L. Luo, H. Lin, and S. Li, “Quantification of 3-D soil macropore networks in different soil types and land uses using computed tomography,” *J. Hydrol.*, vol. 393, no. 1–2, pp. 53–64, 2010, doi: 10.1016/j.jhydrol.2010.03.031.

[116] K. Hormann, V. Baranau, D. Hlushkou, A. Höltzel, and U. Tallarek, “Topological analysis of non-granular, disordered porous media: Determination of pore connectivity, pore coordination, and geometric tortuosity in physically reconstructed silica monoliths,” *New J. Chem.*, vol. 40, no. 5, pp. 4187–4199, 2016, doi: 10.1039/c5nj02814k.

[117] R. Arganda-carreras, Ignacio Fernandez-Gonzalez and A. Munoz-Barrutia, “Analyze Skeleton,” *BoneJ: free and extensible bone image analysis in Imagej*, 2007. https://imagej.net/AnalyzeSkeleton.

[118] I. Arganda-carreras, R. Ferna, and C. Ortiz-de-solorzano, “3D Reconstruction of Histological Sections : Application to Mammary Gland Tissue,” vol. 1029, no. March, pp. 1019–1029, 2010, doi: 10.1002/jemt.20829.

[119] gitHub Free macro, “ND ImageJ plugin for image analysis of porous materials : User Manual,” 2014. https://github.com/sedmorteza/ND/blob/master/User Manual.pdf.

[120] S. M. Longshaw, M. J. Turner, E. Finch, and R. Gawthorpe, “Physics Engine Based Parallelised Discrete Element Model,” *Int. Conf. Part. Methods (Particles 2009)*, no. NOVEMBER, pp. 1–4, 2009.

[121] S. Hedman, “Smooth and non-smooth approaches to simulation of granular matter,” p. 58, 2011, [Online]. Available: http://urn.kb.se/resolve?urn=urn:nbn:se:umu:diva-50584.

[122] S. M. Longshaw, M. J. Turner, E. Finch, and R. Gawthorpe, “Analysing the use of Real-time Physics Engines for Scientific Simulation: Exploring the Theoretical and Practical Benefits for Discrete Element Modelling,” *ACME2010, Proc. 18th Annu. Conf. Assoc. Comput. Mech. Eng.*, no. March, pp. 199–202, 2010, doi: 10.13140/2.1.3212.7048.

[123] K. W. Lim, K. Krabbenhoft, and J. E. Andrade, “A contact dynamics approach to the Granular Element Method,” *Comput. Methods Appl. Mech. Eng.*, vol. 268, pp. 557–573, 2014, doi: 10.1016/j.cma.2013.10.004.

[124] I. R. Ballantyne and S. T. Field, “Collision Overload: Reducing the Impact in Real-time Physics,” 2007.

[125] P. P. . Jacques Moreau, Jean, *NonSmooth mechanics and applications*. 1988.

[126] J. L. Seung and Y. M. A. Hashash, “iDEM: An impulse-based discrete element method for fast granular dynamics,” *Int. J. Numer. Methods Eng. 104(2)*, 2015, [Online]. Available: https://www.researchgate.net/publication/273958063_iDEM_An_impulse-based_discrete_element_method_for_fast_granular_dynamics.

[127] C. Fares and Y. Hamam, “Collision detection for rigid bodies: A state of the art review,” *Graph. 2005 - Int. Conf. Comput. Graph. Vision, Proc.*, 2005.

[128] B. Mirtich, “Impulse-based Dynamic Simulation of Rigid Body Systems,” *Comput. Sci.*, p. 246, 1996.

[129] M. Lin and S. Gottschalk, “Collision detection between geometric models: A survey,” *Proc. IMA Conf. Math. Surfaces*, no. June 1998, pp. 1–20, 1998, doi: 10.1.1.45.3926.

[130] M. Brandl, “Continuous Collision Detection for Wires with Adaptive Resolution,” 2017.

[131] Y. Shen and X. Sun, “Research and improvement of collision detection based on oriented bounding box in physics engine,” *2011 IEEE 3rd Int. Conf. Commun. Softw. Networks, ICCSN 2011*, pp. 73–76, 2011, doi: 10.1109/ICCSN.2011.6014221.

[132] E. Christer, *Real-Time Collision Detection*. 2004.

[133] C. Liang and X. Liu, “The Research of Collision Detection Algorithm Based on Separating axis Theorem,” vol. 2, no. 10, pp. 110–114, 2015.

[134] G. van den Bergen, *Collision Detection in Interactive 3D Environments*. 2003.

[135] M. Tamis and G. Maggiore, “Constraint based physics solver,” vol. 2015, pp. 1–31, 2015.

[136] U. U. Manual, “Unity User Manual (2018 . 3 beta) Working in Unity,” 2018.

[137] H. Xu and J. Barbic, “6-DoF Haptic Rendering Using Continuous Collision Detection between Points and Signed Distance Fields,” *IEEE Trans. Haptics*, vol. 10, no. 2, pp. 151–161, 2017, doi: 10.1109/TOH.2016.2613872.

[138] H. Wang, “Defending continuous collision detection against errors,” *ACM Trans. Graph.*, vol. 33, no. 4, 2014, doi: 10.1145/2601097.2601114.

[139] L. N. M. A. Jyh-Ming, “Approximate convex decomposition of polygons,” *Comput. Geom. 35*, pp. 100–123, 2006.

[140] B. Kenwright, “A Beginners Guide to Dual-Quaternions,” 2012, [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.407.9073&rep=rep1&type=pdf#page=15.

[141] E. M. Moghaddam, E. A. Foumeny, A. I. Stankiewicz, and J. T. Padding, “Rigid Body Dynamics Algorithm for Modeling Random Packing Structures of Nonspherical and Nonconvex Pellets,” *Ind. Eng. Chem. Res.*, vol. 57, no. 44, pp. 14988–15007, 2018, doi: 10.1021/acs.iecr.8b03915.

[142] Y. Qian, E. Tutumluer, Y. M. A. Hashash, J. Ghaboussi, and D. D. Davis,

"Ballast settlement ramp to mitigate differential settlement in a bridge transition zone," *Transp. Res. Rec.*, vol. 2476, no. October, pp. 45–52, 2015, doi: 10.3141/2476-07.

[143] T. D. Gatzke and C. M. Grimm, "Estimating curvature on triangular meshes," *Int. J. Shape Model.*, vol. 12, no. 1, pp. 1–28, 2006, doi: 10.1142/S0218654306000810.

[144] H. Wadell, "Volume , Shape , and Roundness of Quartz," *J. Geol.*, vol. 43, no. 3, pp. 250–280, 1035.

[145] H. Huang, "Discrete element modeling of railroad ballast using imaging based aggregate morphology characterization," no. April, p. 222, 2010.

[146] S. Redon *et al.*, "Fast Continuous Collision Detection between Rigid Bodies," 2009.

[147] G. H. A. J. J. Kumara, K. Hayano, and K. Ogiwara, "Image analysis techniques on evaluation of particle size distribution of gravel," *Int. J. GEOMATE*, vol. 3, no. 1, pp. 290–297, 2012, doi: 10.21660/2012.5.1261.

[148] C. Shanthi, R. Kingsley Porpatham, and N. Pappa, "Image analysis for particle size distribution," *Int. J. Eng. Technol.*, vol. 6, no. 3, pp. 1340–1345, 2014.

[149] N. Xuan *et al.*, "Introduction to Particle Size Analysis," *Refractories*, vol. 22, no. 4, pp. 354–358, 2013, [Online]. Available: http://dx.doi.org/10.1016/j.jart.2016.07.001.

[150] E. Olson, "Particle Shape Factors and Their Use in Image Analysis – Part 1 : Theory," *J. GXP Compliance*, vol. 15, no. 3, pp. 85–96, 2011, [Online]. Available: http://www.ivtnetwork.com/article/particle-shape-factors-and-their-use-image-analysis?part-1-theory.

[151] L. Esmaeelnejad, F. Siavashi, J. Seyedmohammadi, and M. Shabanpour, "The best mathematical models describing particle size distribution of soils," *Model. Earth Syst. Environ.*, vol. 2, no. 4, pp. 1–11, 2016, doi: 10.1007/s40808-016-0220-9.

[152] T. M. Zobeck *et al.*, "Aggregate-Mean Diameter and Wind-Erodible Soil Predictions Using Dry Aggregate-Size Distributions," *Soil Sci. Soc. Am. J.*, vol. 67, no. 2, p. 425, 2014, doi: 10.2136/sssaj2003.4250.

[153] Indian Highways, “Specification for Bituminous Macadam,” no. August, pp. 1–13, 2007.

[154] M. Aboufoul, A. Chiarelli, I. Triguero, and A. Garcia, “Virtual porous materials to predict the air void topology and hydraulic conductivity of asphalt roads,” *Powder Technol.*, vol. 352, pp. 294–304, 2019, doi: 10.1016/j.powtec.2019.04.072.

[155] M. Doube, “Volume Fraction,” 2004. http://bonej.org/volumefraction.

[156] S. S. Doube M, Kłosowski MM, Arganda-Carreras I, Cordeliéres F, Dougherty RP, Jackson J, Schmid B, Hutchinson JR, “BoneJ: free and extensible bone image analysis in ImageJ,” *Bone 47:1076-9. doi: 10.1016/j.bone.2010.08.023*, 2010. .

[157] R. Arganda-carreras, Ignacio Fernandez-Gonzalez, “Skeletonize3D,” *2007*. https://imagej.net/Skeletonize3D.

[158] J. Brocher, “BioVoxxel Toolbox,” 2018. https://imagej.net/BioVoxxel_Toolbox.

[159] “Particle orientation meassurment,” 2019. https://github.com/peterjlee/asc-ImageJ-MinimumDistances/blob/master/Add_Min_and_Max-Ref-Dist-to-Centroids_to_Results.ijm.

[160] A. Torkkeli, “Practical Methods for Measuring the Tortuosity of Porous Materials from Binary or Gray-Tone Tomographic Reconstructions,” *VTT Publ.*, vol. 55, no. 504, pp. 3–194, 2003, doi: 10.1002/aic.

[161] N. A. Hassan, G. D. Airey, R. Khan, and A. C. Collop, “Nondestructive characterisation of the effect of asphalt mixture compaction on aggregate orientation and segregation using x-ray computed tomography,” *Int. J. Pavement Res. Technol.*, vol. 5, no. 2, pp. 84–92, 2012, doi: 10.6135/ijprt.org.tw/2012.5(2).84.

[162] F. U. Rehman, “The Study of Aggregate Segregation and Asphalt Concrete Permeability from Microstructural Analysis,” *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 7, no. 8, pp. 1121–1134, 2019, doi: 10.22214/ijraset.2019.8157.

[163] M. M. Roozbahani, R. Borela, and J. D. Frost, “Pore size distribution in granular material microstructure,” *Materials (Basel).*, vol. 10, no. 11, pp. 1–21, 2017, doi: 10.3390/ma10111237.

[164] M. S. Sumanasooriya, O. Deo, and N. Neithalath, “Particle packing-based material design methodology for pervious concretes,” *ACI Mater. J.*, vol. 109, no. 2, pp. 205–214, 2012, doi: 10.14359/51683707.

[165] R. R. Septiawan, S. Viridi, and Suprijadi, “The effect of particle size ratio on porosity of a particles deposition process,” *Key Eng. Mater.*, vol. 675–676, no. June, pp. 647–650, 2016, doi: 10.4028/www.scientific.net/KEM.675-676.647.

[166] M. do C. F. Teixeira Freire, José Ana Maria da Silveira, *Transport Phenomena In Particulate Systems*. 2012.

[167] A. B. Yu and N. Standish, “Characterisation of non-spherical particles from their packing behaviour,” *Powder Technol.*, vol. 74, no. 3, pp. 205–213, 1993, doi: 10.1016/0032-5910(93)85029-9.

[168] W. Sobieski and S. Lipiński, “The analysis of the relations between porosity and tortuosity in granular beds,” *Tech. Sci.*, vol. 1, no. 20, pp. 75–85, 2017, doi: 10.31648/ts.2912.

[169] H. T. Hamad, D. W. Abbood, and A. S. Mustafa, “Strength Tortuosity - Porosity Relation in Locally Types of Porous Media (Experimental Model),” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 454, no. 1, 2018, doi: 10.1088/1757-899X/454/1/012093.

[170] T. Cao *et al.*, “Model for predicting the tortuosity of transport paths in cement-based materials,” *Materials (Basel).*, vol. 12, no. 21, pp. 1–19, 2019, doi: 10.3390/ma12213623.

[171] N. A. Hassan, M. Z. H. Mahmud, and R. P. Jaya, “Air void characterisation in porous asphalt using x-ray computed tomography,” *Adv. Mater. Res.*, vol. 911, no. January 2014, pp. 443–448, 2014, doi: 10.4028/www.scientific.net/AMR.911.443.

[172] P. Lovreglio, S. Das, K. A. Buist, E. A. J. F. Peters, L. Pel, and J. A. M. Kuipers, “Experimental and numerical investigation of structure and hydrodynamics in packed beds of spherical particles,” *AIChE J.*, vol. 64, no. 5, pp. 1896–1907, 2018, doi: 10.1002/aic.16127.

[173] F. Chen, D. Jelagin, and M. N. Partl, “Vibration-induced aggregate segregation in asphalt mixtures,” *Mater. Struct. Constr.*, vol. 53, no. 2, pp. 1–14, 2020, doi: 10.1617/s11527-020-01459-y.

[174] J. G. Cabrera and M. O. Hamzah, *Aggregate grading design for porous asphalt*,

L. 1994.

[175] British Standard Institute (BSI), *Bituminous mixtures - Test methods for hot mix asphalt - Part 5: Determination of the maximum density*, vol. 3, no. 1. 2009.

[176] J. Norambuena-Contreras, Q. Liu, L. Zhang, S. Wu, E. Yalcin, and A. Garcia, "Influence of encapsulated sunflower oil on the mechanical and self-healing properties of dense-graded asphalt mixtures," *Mater. Struct. Constr.*, vol. 52, no. 4, pp. 1–13, 2019, doi: 10.1617/s11527-019-1376-3.

[177] British Standard Institute (BSI), *BS EN 12697-31, Test Methods for Hot Mix Asphalt: Specimen Prepared by Gyratory Compactor.* 2007.

[178] P. Polaczyk, B. Huang, X. Shu, and H. Gong, "Investigation into Locking Point of Asphalt Mixtures Utilizing Superpave and Marshall Compactors," *J. Mater. Civ. Eng.*, vol. 31, no. 9, p. 04019188, 2019, doi: 10.1061/(asce)mt.1943-5533.0002839.

[179] The British Standards Institution 2012, "Bituminous mixtures - Test methods for hot mix asphalt - Part 6: Determination of bulk density of bituminous specimens," no. september, p. 149, 2005.

[180] Y. Du, P. Liu, J. Tian, J. Zhang, and Y. Zheng, "Preliminary investigation of the feasibility of using a Superpave Gyratory compactor to design cement-treated aggregate mixture," *Appl. Sci.*, vol. 8, no. 6, 2018, doi: 10.3390/app8060946.

[181] T. Stephen, "Improving and augmenting the ANM model for three-dimensional virtual concrete," Boise State University, 2018.

[182] J. M. R. Fernlund, R. W. Zimmerman, and D. Kragic, "Influence of volume/mass on grain-size curves and conversion of image-analysis size to sieve size," *Eng. Geol.*, vol. 90, no. 3–4, pp. 124–137, 2007, doi: 10.1016/j.enggeo.2006.12.007.

[183] C.-Y. Wu, *Discrete Element Modelling of Particulate Media*. 2012.

[184] M. Kanzler, M. Rautenhaus, and R. Westermann, "A Voxel-Based Rendering Pipeline for Large 3D Line Sets," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 7, pp. 2378–2391, 2019, doi: 10.1109/TVCG.2018.2834372.

[185] A. R. Fuller, H. Krishnan, K. Mahrous, B. Hamann, and K. I. Joy, "Real-time procedural volumetric fire," *Proc. - I3D 2007, ACM SIGGRAPH Symp. Interact. 3D Graph. Games*, no. January, pp. 175–180, 2007, doi:

10.1145/1230100.1230131.

[186] Unity Technologies, “Physics.Raycast.” https://docs.unity3d.com/ScriptReference/Physics.Raycast.html.

[187] Math.NET Numerics, “MathNet.Numerics.Distributions.” https://numerics.mathdotnet.com/api/MathNet.Numerics.Distributions/index.htm.

[188] D. Wang, X. Ding, T. Ma, W. Zhang, and D. Zhang, “Algorithm for virtual aggregates’ reconstitution based on image processing and discrete-element modeling,” *Appl. Sci.*, vol. 8, no. 5, pp. 1–16, 2018, doi: 10.3390/app8050738.

[189] O. VO and A. EE, “Effects of Aggregate Gradation on the Properties of Concrete Made From Granite Chippings,” *J. Steel Struct. Constr.*, vol. 2, no. 1, 2016, doi: 10.4172/2472-0437.1000109.

[190] T. Hamano, M. Onosato, and F. Tanaka, “Performance comparison of physics engines to accelerate house-collapsing simulations,” *SSRR 2016 - Int. Symp. Safety, Secur. Rescue Robot.*, pp. 358–363, 2016, doi: 10.1109/SSRR.2016.7784327.

[191] A. Seugling and M. Rolin, “Evaluation of physics engines and implementation of a physics module in a 3d-authoring tool,” *Umea Univ.*, p. 89, 2006, [Online]. Available: http://www8.cs.umu.se/education/examina/Rapporter/SeuglingRolin.pdf.

[192] A. Boeing and T. Bräunl, “Evaluation of real-time physics simulation systems,” *Proc. 5th Int. Conf. Comput. Graph. Interact. Tech. Aust. Southeast Asia - Graph. ’07*, vol. 1, no. 212, p. 281, 2007, doi: 10.1145/1321261.1321312.

[193] Developer.nvidia.com, “NVIDIA PhysX SDK 3.4.0 Documentation,” *https://docs.nvidia.com/gameworks/content/gameworkslibrary/physx/guide/Manual/Index.html*, 2017. .

[194] J. C. Linford, “Parallelized Rigid Body Dynamics,” 2014.

[195] José Mário Escudeiro de Aguiar, “Phenomenological Numerical Modelling of a Mineral Jig using Parallel Processing .,” 2015.

[196] H. Niu, Y. Gao, and Z. Hou, “Application research of PhysX engine in virtual environment,” *ICALIP 2010 - 2010 Int. Conf. Audio, Lang. Image Process.*

Proc., pp. 587–591, 2010, doi: 10.1109/ICALIP.2010.5684497.

[197] S. C. Mondesire *et al.*, "Physics Engine Benchmarking in Three-Dimensional Virtual World Simulation Physics Engine Benchmarking in Three-Dimensional Virtual World Simulation," *MODSIM World 2016*, no. 26, pp. 1–8, 2016.

[198] J. Hummel, R. Wolff, and T. Stein, "An Evaluation of Open Source Physics Engines for Use in Virtual Reality Assembly Simulations," *Adv. Vis. ...*, vol. 7432, no. August, pp. 346–357, 2012, doi: 10.1007/978-3-642-33191-6.

[199] J. Bender, K. Erleben, and J. Trinkle, "Interactive simulation of rigid body dynamics in computer graphics," *Comput. Graph. Forum*, vol. 33, no. 1, pp. 246–270, 2014, doi: 10.1111/cgf.12272.

[200] Unity Technologies, "Unity 3D manual," *2018.1-002E. Built: 2018-06-11.* https://docs.unity3d.com/Manual/class-Rigidbody.html.

[201] M. Robotham, H. Wang, and G. Walton, "Earth Retaining StructuresAssessment of risk from rockfall from active and abandoned quarry slopes," *Bridg. Eng. Handb.*, vol. 64, no. 12, pp. 299–330, 1995, doi: 10.1201/b15621-15.

[202] K. Senetakis, C. S. Sandeep, and M. C. Todisco, "Dynamic inter-particle friction of crushed limestone surfaces," *Tribol. Int.*, vol. 111, no. January, pp. 1–8, 2017, doi: 10.1016/j.triboint.2017.02.036.

[203] Unity Technologies, "Physic Material," *Unity User Manual*. https://docs.unity3d.com/Manual/class-PhysicMaterial.html.

[204] I. Ishibashi, C. Perry, and T. Agarwal, "Experimental determinations of contact friction for spherical glass particles," *Soils Found.*, vol. 34, no. 534, pp. 364–370, 1994, [Online]. Available: http://www.mendeley.com/research/geology-volcanic-history-eruptive-style-yakedake-volcano-group-central-japan/.

[205] H. Tang, R. Song, Y. Dong, and X. Song, "Measurement of restitution and friction coefficients for granular particles and discrete element simulation for the tests of glass beads," *Materials (Basel).*, vol. 12, no. 19, 2019, doi: 10.3390/ma12193170.

[206] Unity Technologies, "Concave Collider." https://assetstore.unity.com/packages/tools/physics/concave-collider-4596.

[207] B. Chazelle, D. P. Dobkin, N. Shouraboura, and A. Tal, "Strategies for

polyhedral surface decomposition: An experimental study," *Comput. Geom. Theory Appl.*, vol. 7, no. 5–6, pp. 327–342, 1997, doi: 10.1016/S0925-7721(96)00024-7.

[208] K. Mamou and F. Ghorbel, "A simple and efficient approach for 3D mesh approximate convex decomposition," *Proc. - Int. Conf. Image Process. ICIP*, no. May 2014, pp. 3501–3504, 2009, doi: 10.1109/ICIP.2009.5414068.

[209] Ultimate Game Tools, "Concave Collider Help," 2012. http://www.ultimategametools.com/products/concave_collider/help.

[210] NVIDIA PhysX SDK, "Advanced Collision Detection." https://docs.nvidia.com/gameworks/content/gameworkslibrary/physx/guide/Manual/AdvancedCollisionDetection.html.

[211] J. M. Gutiérrez, Á. A. Magreñán, and J. L. Varona, "The 'gauss-Seidelization' of iterative methods for solving nonlinear equations in the complex plane," *Appl. Math. Comput.*, vol. 218, no. 6, pp. 2467–2479, 2011, doi: 10.1016/j.amc.2011.07.061.

[212] Unity Technologies, "Unity Manual. Physics," 2019. https://docs.unity3d.com/Manual/class-PhysicsManager.html.

[213] IT1352, "How to calculate the volume of a 3D mesh object the surface of which is made up triangles," 2013. https://en.it1352.com/article/35b7091f1b5b4191bd7b82db7fc15de6.html.

[214] C. Zhang and C. Tsuhan, "Efficient feature extraction for 2d/3d objects in mesh representation," *Virtual Real.*, pp. 1–4, [Online]. Available: http://chenlab.ece.cornell.edu/Publication/Cha/icip01_Cha.pdf.

[215] blender.org, "Blender." https://www.blender.org/.

[216] S. Massa, "Kolmogorov Smirnov Test & Power of Tests," no. February, p. 22, 2016, [Online]. Available: http://www.stats.ox.ac.uk/~massa/Lecture 13.pdf.

[217] Real Statistics 2020, "Kolmogorov-Smirnov Test for Normality," 2019. http://www.real-statistics.com/tests-normality-and-symmetry/statistical-tests-normality-symmetry/kolmogorov-smirnov-test/.

[218] Y. Liu, M. Li, P. Su, B. Ma, and Z. You, "Porosity prediction of granular materials through discrete element method and back propagation Neural

Network algorithm," *Appl. Sci.*, vol. 10, no. 5, 2020, doi: 10.3390/app10051693.

[219] J. Chen, B. Huang, F. Chen, and X. Shu, "Application of discrete element method to Superpave gyratory compaction," *Road Mater. Pavement Des.*, vol. 13, no. 3, pp. 480–500, 2012, doi: 10.1080/14680629.2012.694160.

[220] S. Shen and H. Yu, "Analysis of Aggregate Gradation and Packing for Easy Estimation of Hot-Mix-Asphalt Voids in Mineral Aggregate," *J. Mater. Civ. Eng.*, vol. 23, no. 5, pp. 664–672, 2011, doi: 10.1061/(asce)mt.1943-5533.0000224.

[221] J. Li, J. Zhang, G. Qian, J. Zheng, and Y. Zhang, "Three-Dimensional Simulation of Aggregate and Asphalt Mixture Using Parameterized Shape and Size Gradation," *J. Mater. Civ. Eng.*, vol. 31, no. 3, p. 04019004, 2019, doi: 10.1061/(asce)mt.1943-5533.0002623.

[222] T. Eltaeib and A. Mahmood, "Differential evolution: A survey and analysis," *Appl. Sci.*, vol. 8, no. 10, 2018, doi: 10.3390/app8101945.

[223] F. Neri and V. Tirronen, *Recent advances in differential evolution: A survey and experimental analysis*, vol. 33, no. 1–2. 2010.

[224] Mbalchanowski, "Differential-Evolution," 2018. https://github.com/mbalchanowski/Differential-Evolution.

[225] A. W. Mohamed, "Solving large-scale global optimization problems using enhanced adaptive differential evolution algorithm," *Complex Intell. Syst.*, vol. 3, no. 4, pp. 205–231, 2017, doi: 10.1007/s40747-017-0041-0.

[226] P. Civicioglu and E. Besdok, *A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms*, vol. 39, no. 4. 2013.

[227] J. J. Grefenstette, "Optimization of Control Parameters for Genetic Algorithms," *IEEE Trans. Syst. Man Cybern.*, vol. 16, no. 1, pp. 122–128, 1986, doi: 10.1109/TSMC.1986.289288.

[228] H. Mühlenbein and D. Schlierkamp-Voosen, "Optimal Interaction of Mutation and Crossover in the Breeder Genetic Algorithm," *Proc. Fifth Int. Conf. Genet. Algorithms*, no. January 1993, p. 648, 1993.

[229] A. Hassanat, K. Almohammadi, E. Alkafaween, E. Abunawas, A. Hammouri,

and V. B. S. Prasath, “Choosing mutation and crossover ratios for genetic algorithms-a review with a new dynamic approach,” *Inf.*, vol. 10, no. 12, 2019, doi: 10.3390/info10120390.

[230] R. Storn and K. Price, “Differential evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces,” *Tech. Rep.*, no. TR-95-012, pp. 1–12, 1995, [Online]. Available: ftp://ftp.icsi.berkeley.edu/pub/techreports/1995/tr-95-012.pdf.

[231] S. K. Goudos, K. B. Baltzis, K. Antoniadis, Z. D. Zaharis, and C. S. Hilas, “A comparative study of common and self-adaptive Differential Evolution strategies on numerical benchmark problems,” *Procedia Comput. Sci.*, vol. 3, pp. 83–88, 2011, doi: 10.1016/j.procs.2010.12.015.

[232] A. Roennau, F. Sutter, G. Heppner, J. Oberlaender, and R. Dillmann, “Evaluation of physics engines for robotic simulations with a special focus on the dynamics of walking robots,” *2013 16th Int. Conf. Adv. Robot. ICAR 2013*, 2013, doi: 10.1109/ICAR.2013.6766527.

[233] U. Technologies, “OnCollisionEnter.” https://docs.unity3d.com/ScriptReference/Collider.OnCollisionEnter.html.

[234] Unity Technologies, “contactCount.” https://docs.unity3d.com/ScriptReference/Collision-contactCount.html.

[235] Unity Technologies, “Compute Penetration.” https://docs.unity3d.com/ScriptReference/Physics.ComputePenetration.html.

[236] R. Tonge, F. Benevolenski, and A. Voroshilov, “Mass splitting for jitter-free parallel rigid body simulation,” *ACM Trans. Graph.*, vol. 31, no. 4, 2012, doi: 10.1145/2185520.2185601.

[237] H. Yu and S. Shen, “Impact of aggregate packing on dynamic modulus of hot mix asphalt mixtures using three-dimensional discrete element method,” *Constr. Build. Mater.*, vol. 26, no. 1, pp. 302–309, 2012, doi: 10.1016/j.conbuildmat.2011.06.025.

[238] J. H. Falkenby, “Physically-based fluid-particle system using DirectCompute for use in real-time games,” 2014.

[239] D. Muliyati, D. Sumardani, S. Siswoyo, F. Bakri, E. Handoko, and H. Permana, “The effectiveness of breakwater shape: Fluid particle behavior simulation,” *J. Phys. Conf. Ser.*, vol. 1869, no. 1, 2021, doi: 10.1088/1742-6596/1869/1/012186.

[240] A.Hilton and J.Illingworth, “Marching Triangles: Delaunay Implicit Surface Triangulation,” *Clim. Chang. 2013 Phys. Sci. Basis*, vol. 1, no. September 2013, pp. 1–36, 2013, [Online]. Available: http://eprints.utas.edu.au/4774/.

[241] A. Garcia-Hernandez, S. Michot-Roberto, S. Dopazo-Hilario, A. Chiarelli, and A. Dawson, “Creation of realistic virtual aggregate avatars,” *Powder Technol.*, vol. 378, pp. 760–771, 2021, doi: 10.1016/j.powtec.2020.10.036.

[242] A. Zhang *et al.*, “Automated Pixel-Level Pavement Crack Detection on 3D Asphalt Surfaces with a Recurrent Neural Network,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 34, no. 3. pp. 213–229, 2019, doi: 10.1111/mice.12409.

[243] M. Zhang, W. Cheng, and Y. Wang, “Multiple-Fault Classification for Hot-Mix Asphalt Production by Machine Learning,” *J. Constr. Eng. Manag.*, vol. 144, no. 5, p. 04018024, 2018, doi: 10.1061/(asce)co.1943-7862.0001470.

[244] Z. Lai and Q. Chen, “Reconstructing granular particles from X-ray computed tomography using the TWS machine learning tool and the level set method,” *Acta Geotech.*, vol. 14, no. 1, pp. 1–18, 2019, doi: 10.1007/s11440-018-0759-x.

[245] A. Garcia Hernandez, L. Wan, and S. Dopazo Hilario, “In-silico manufacturing of asphalt concrete,” *Powder Technol.*, pp. 1–2, 2021.

[246] M. R. Rahimi Z, “Quantification of Coarse Aggregate Angularity by A Newly Developed Auto Grader Machine.,” *Spec. Issue Curr World Env. 2016;11(Special Issue 2(2016).. doi* , 2016, [Online]. Available: http://dx.doi.org/10.12944/CWE.11.Special-Issue1.07.

[247] L. Miranda, B. Rubio, R. Jiménez, L. Enríquez, and J. Loma, “Método De Dosificación Bailey. Aplicación a Las Mezclas Tipo Stone Mastic Asphalt (Sma),” *Asefma*, pp. 1–265, 2012.

[248] N. Shashidhar and K. Gopalakrishnan, “computer modeling and particle packing simulations Evaluating the aggregate structure in hot-mix asphalt using three-dimensional computer modeling and particle packing simulations,” no. March 2014, 2011, doi: 10.1139/l06-046.

[249] A. Delaby, L. Espinosa, C. Lépolard, C. Capo, and J. Mège, “3D reconstruction of granulomas from transmitted light images implemented for long-time microscope applications,” *J. Immunol. Methods*, vol. 360, no. 1–2, pp. 10–19, 2010, doi: 10.1016/j.jim.2010.06.008.

[250] Minitab 18 support, “Distribution ID plot for Distribution ID Plot (Arbitrary Censoring),” 2017. https://support.minitab.com/en-us/minitab/18/help-and-how-to/modeling-statistics/reliability/how-to/distribution-id-plot-arbitrary-censoring/interpret-the-results/distribution-id-plot/.

[251] H. Rinne, *The Weibull Distributation: A Handbook*. 2008.

[252] Z. Lu, L. Dong, and J. Zhou, “Nonlinear Least Squares Estimation for Parameters of Mixed Weibull Distributions by Using Particle Swarm Optimization,” *IEEE Access*, vol. 7, no. May, pp. 60545–60554, 2019, doi: 10.1109/ACCESS.2019.2915279.

[253] J. M. R. Fernlund, “The effect of particle form on sieve analysis: a test by image analysis,” *Eng. Geol.*, vol. 50, no. 1–2, pp. 111–124, 1998, doi: 10.1016/S0013-7952(98)00004-0.

[254] C. F. Mora, A. K. H. Kwan, and H. . Chan, “Particle size distribution analysis of coarse aggregate using digital image processing,” *Cem. Concr. Res.*, vol. 28, no. October, p. 226765, 1998.

[255] J. Congote, A. Moreno, I. Barandiaran, J. Barandiaran, J. Posada, and O. Ruiz, “Marching cubes in an unsigned distance field for surface reconstruction from unorganized point sets,” *GRAPP 2010 - Proc. Int. Conf. Comput. Graph. Theory Appl.*, pp. 143–147, 2010, doi: 10.5220/0002846901430147.

[256] A. Opalach and S. C. Maddock, “An overview of implicit surfaces,” *Introd. to Model. Animat. using implicit surfaces*, no. January, p. 1, 1995.

[257] L. Wang, “Algorithms and Criteria for Volumetric Centroidal Voronoi Tessellations,” l’École Doctorale Mathématiques, Sciences et technologies de l’information, Informatique Algorithmes, 2018.

[258] R. Shu, C. Zhou, and M. S. Kankanhalli, “Adaptive marching cubes,” *Vis. Comput.*, vol. 11, no. 4, pp. 202–217, 1995, doi: 10.1007/BF01901516.

[259] G. M. Nielson, “On marching cubes,” *Vis. Comput. graph*, vol. 9, pp. 283–297, 2003.

[260] M. Li, X. Dong, and Z. Zhong, “a software tool for molecular visualization improving marching cubes.pdf,” *Comput. Theor. Chem.*, vol. 1028, 2014.

[261] C. A. Dietrich, C. E. Scheidegger, J. L. . Comba, L. P. Nedel, and C. T. Silva,

"Marching Cubes without Skinny Triangles," *Comput. Sci. Eng.*, vol. 11, pp. 82–87, 2009.

[262] B. K. Natarajan, "On generating topologically consistent isosurfaces from uniform samples," *Vis. Comput.*, vol. 11, pp. 52–62, 1994.

[263] A. DOI and A. KOIDE, "An Efficient Method of Triangulating Equi-Valued Surfaces by Using Tetrahedral Cells," *IEICE Trans.*, vol. E74 D, pp. 214–224, 1991.

[264] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," *ACM SIGGRAPH Comput. Graph.*, vol. 21, no. 4, pp. 163–169, 1987, doi: 10.1145/37402.37422.

[265] K. Miltenberger, "Polygonising A Scalar Field," *Gene*, pp. 1–15, 2010.

[266] Ryan Geissn (NVIDIA Corporation), "Chapter 1. Generating Complex Procedural Terrains Using the GPU," 2015. https://developer.nvidia.com/gpugems/GPUGems3/gpugems3_ch01.html.

[267] H. Robert, "Marching Cubes Robert Hunt CS 525."

[268] L. Custodio, S. Pesco, and C. Silva, "An extended triangulation to the Marching Cubes 33 algorithm," *J. Brazilian Comput. Soc.*, vol. 25, no. 1, 2019, doi: 10.1186/s13173-019-0086-6.

[269] J. C. F. Winter, "Using the student's t-test with extremely small sample sizes," *Pract. Assessment, Res. Eval.*, vol. 18, no. 10, pp. 1–12, 2013.

[270] ttable.org, "T Value Table." http://www.ttable.org/.

[271] M. Krzywinski and N. Altman, "Points of significance: Significance, P values and t-tests," *Nat. Methods*, vol. 10, no. 11, pp. 1041–1042, 2013, doi: 10.1038/nmeth.2698.

[272] C. Zhang, H. Wang, Z. You, and X. Yang, "Compaction characteristics of asphalt mixture with different gradation type through Superpave Gyratory Compaction and X-Ray CT Scanning," *Constr. Build. Mater.*, vol. 129, pp. 243–255, 2016, doi: 10.1016/j.conbuildmat.2016.10.098.

APPENDIX A. EXPERIMENTAL WORK

A.1 Particle´s samples characterization

A.1.1 ImageJ procedure

The particles samples were arranged manually on a white sheet in their most stable position without touching each other, as shown in Figure A-2. A camera and lights were used to capture the 2D shape´s characteristics of the aggregate´s samples follow the procedure described in [147], which consist on: (i) establishing the scale, (ii) highlighting the borders of the particles and correcting the shading through the option *Bandpass Filter* of FFT (Fast Fourier Transform) [83] and, (iii) measuring and calculation the 2d shape´s features, using the BioVoxel plugin and its functions for extended particle analysis [158]. The workflow of this process is pictured in Figure A-2.

Figure A-1 – Original image (left) and binary image (right) of a sample of limestone (10mm) placed on a clean surface free of dirt

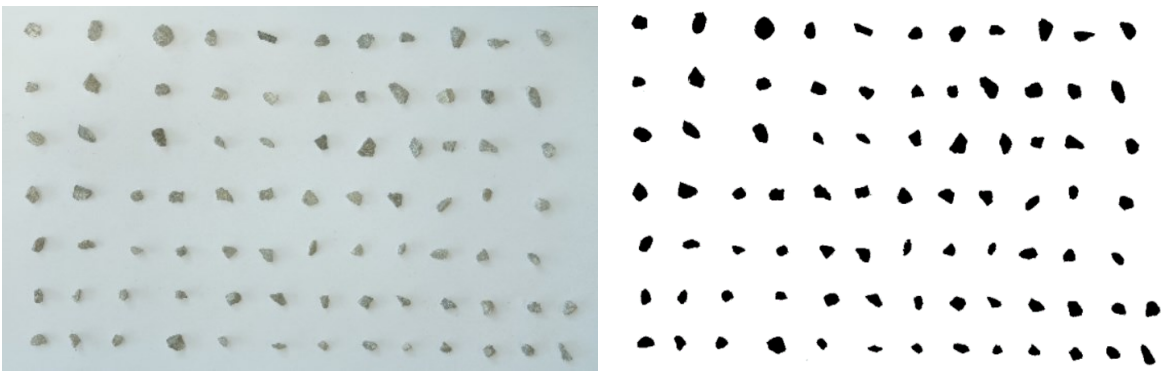
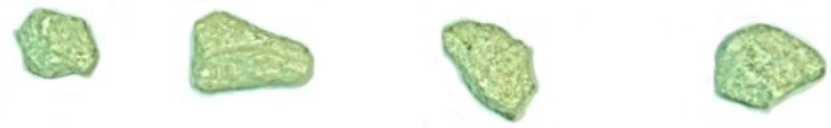


Figure A-2 – Original Original image (left) and binary image (right) of a sample of limestone (10mm) placed

- *Aggregate image capturing*



- *Apply Band pass Filter*



- *Make image as a binary*



- *Apply the binary option Fill holes*



The option *Bandpass Filter* of FFT (Fast Fourier Transform) serves to highlight the borders of the particles and correct the shading, by means of a Gaussian filtering in the Fourier space to obtain a greater visual contrast [249]. The *fill holes* function fills the background of connected elements (in this case the borders of the particles) [83].

A.1.2 Anderson-Darling adjustment though the software Minitab

The size and the shape´s factors of an aggregate´s sample tend to follow a mathematical law in their distribution. For that reason, the data obtained from ImageJ were analysed by the statistical software Minitab, which can make a comparison between various distribution curves using the Anderson-Darling adjustment. This latter

indicates the goodness-of-fit to a distribution curve; the closer the result is to zero, the better the fit of the curve to the data points [250].

Results showed that the Weibull distribution model for all the shape´s factors analyzed, was the most accurate for most of the samples, since the Anderson-Darling values were the lowest than all other distribution (which is consistent with previous research as can be read in: [63] [151] [152]); with the exception of the 3-parameters Weibull distributions. Nevertheless, in order to simplify the virtual model of generation of aggregates, the three-parameter distribution models were not taken into account.

A.1.2.1 Gradation distribution curves

In Figure A-3, an example of the comparisons between the distribution models for a representative sample of each type of stones is drawn. This figure shows that the Weibull model had the best accuracy for all types of stones (only surpassed by the three-parameter model, already mentioned); except for Crushed Glass (CG), and Round Gravel (RG) samples. However, as the difference with respect to the other distribution models is not too large, for convenience it was decided to characterise the Minor Feret diameter by Weibull statistics in all the samples. Figure A-4 shows the cumulative distributions of the Minor Feret´s shape parameter, for all the particle´s samples on Weibull plots with fitting lines.

Figure A-3 – Original Comparison of distribution models accuracy for the Minor Feret diameter

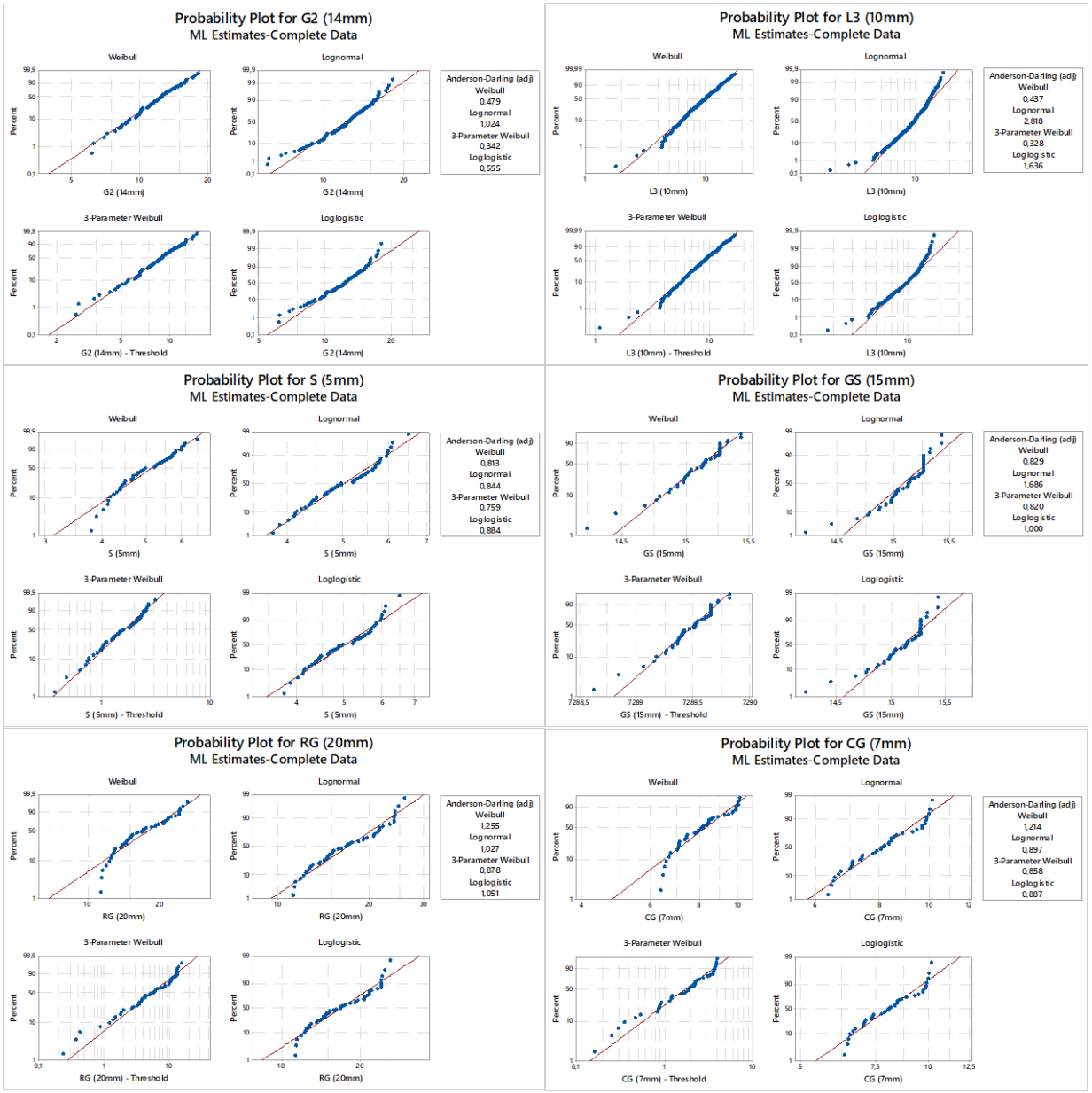
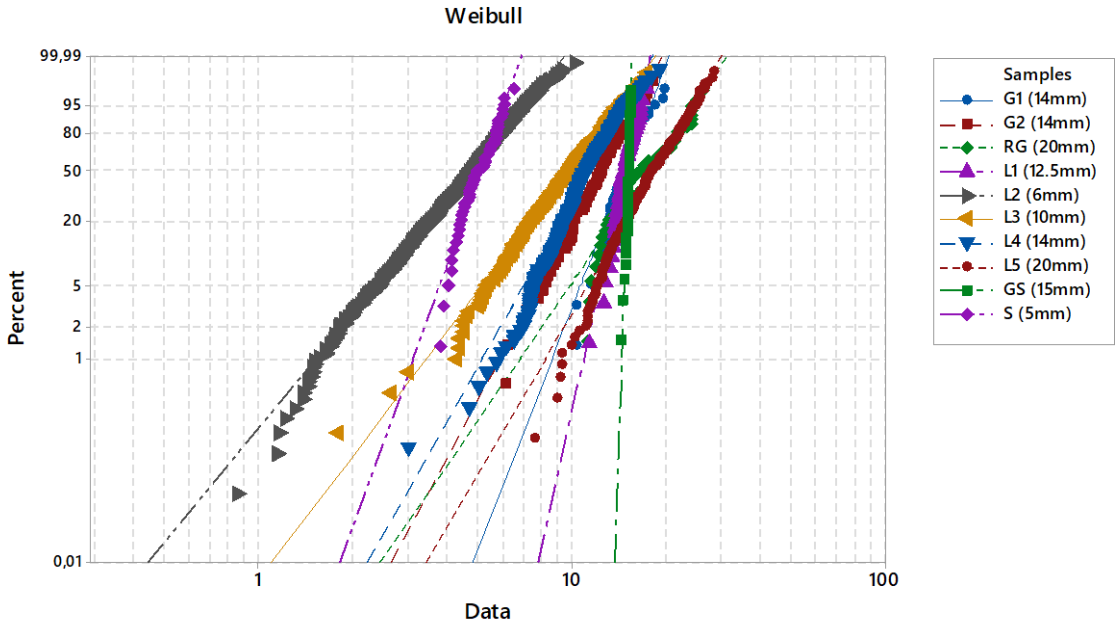


Figure A-4 – Cumulative distributions of the shape parameter Minor Feret for all the particle´s samples on Weibull plots with fitting lines



A.1.2.2 Aspect Ration distribution curves

The same phenomenon occurred for the Minor Feret diameter is observed for the shape's factor Aspect Ratio, as shown by Figure A-5; where the Weibull distribution is the most precise (without taking into account the three-parameters Weibull distribution, which, as mentioned, is the most accurate), over all the samples analyzed, except for the samples: Crushed Glass (CG), and Round Gravel (RG); in which the Weibull distribution is not the most precise, although with very little difference with respect to the other models studied.

Figure A-6 shows the Cumulative distributions of the Aspect Ratio shape´s factor, for all the particle´s samples on Weibull plots with fitting lines.

Figure A-5 – Comparison of distribution models accuracy for the shape´s factor Aspect Ratio

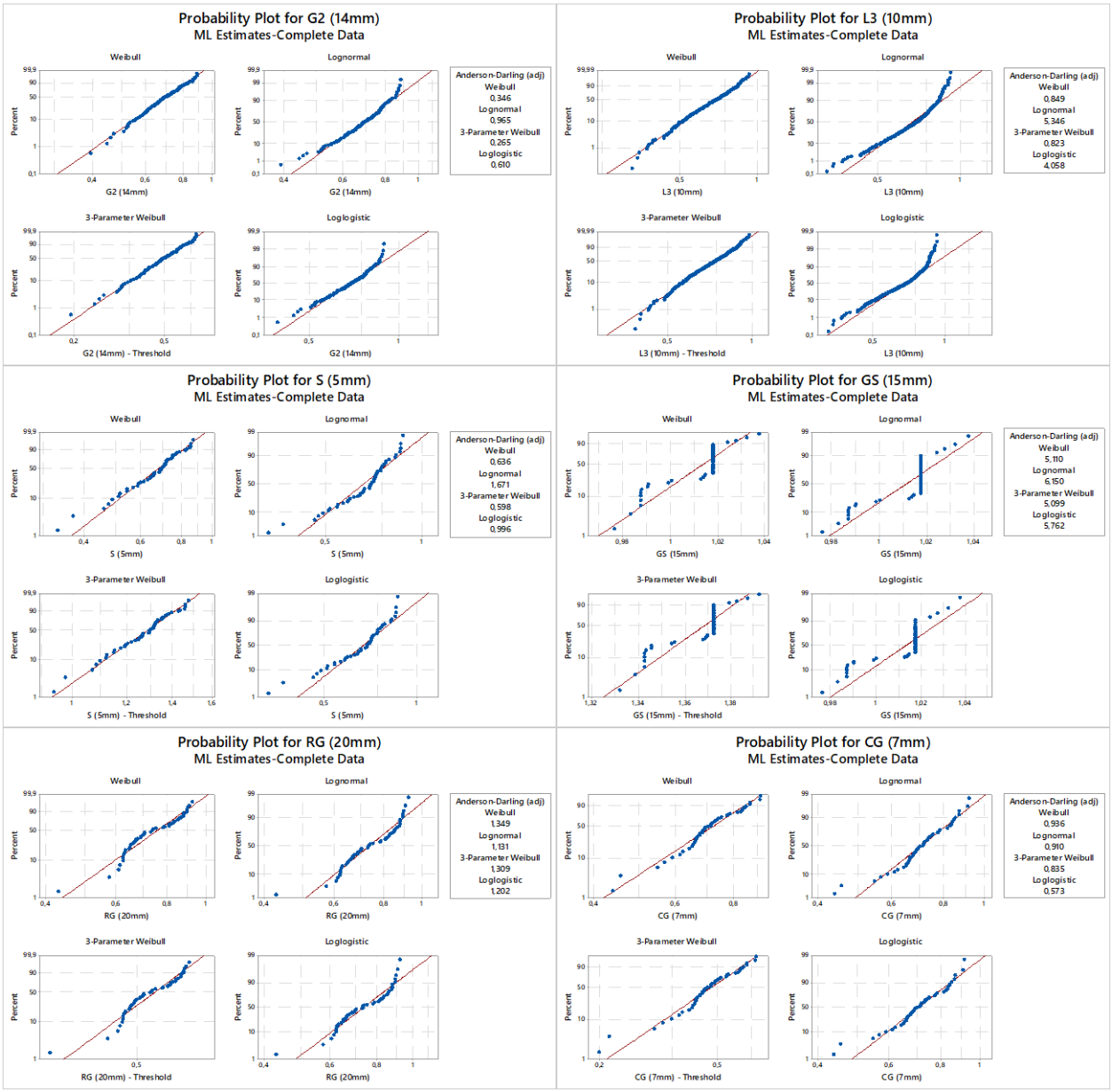
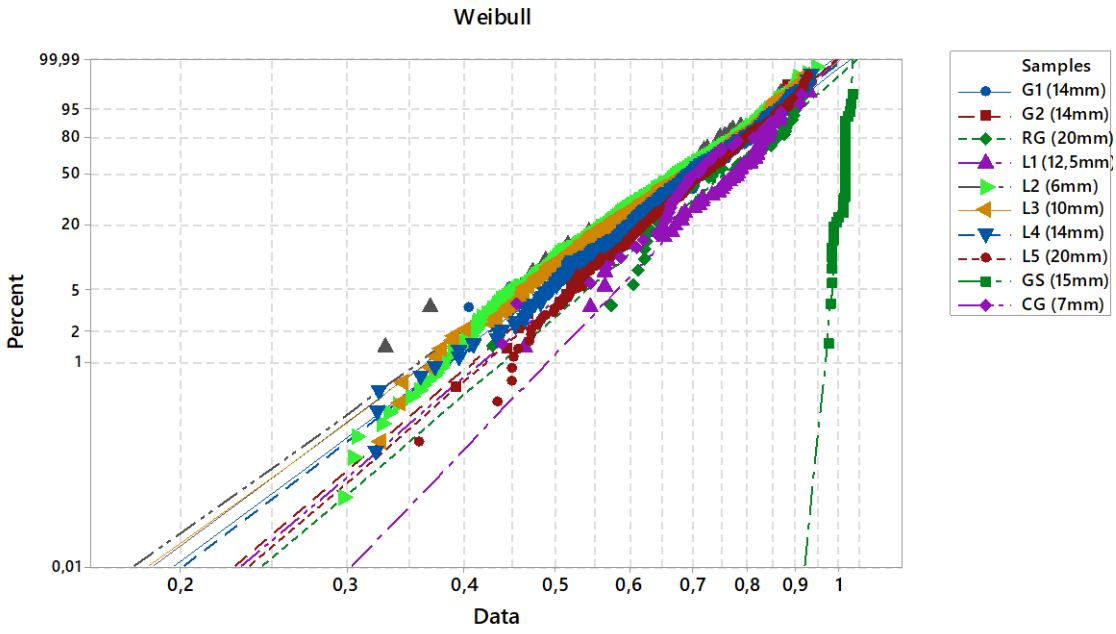


Figure A-6 – Original – Cumulative distributions of the shape parameter Aspect Ratio for all the particle´s samples on Weibull plots with fitting lines



A.1.3 Iterative least squares technique to find the Weibull statistic parameters

The probability density function of a variable *t*, $f(t)$, of the Weibull distribution is given by Equation A-1[251], from it can be seen that the Weibull distribution is a two-parameter function defined by: the shape parameter, k, and scale parameter, λ, plus a location parameter ρ, which serve to find the Weibull statistic´s parameters by an iterative least squares technique.

This iterative least squares technique is used in this study to find the Weibull distribution parameters of the size (minor Feret diameter) and the Aspect Ratio obtained from the experimental data.

The Least Squares Estimation (LSE) is a regression method often used for this purpose, which allows calculation of the Weibull parameters: shape and scale, by the

double logarithmic transformation of the accumulated function, $F(t)$, of the Weibull probability Equation A-3.

The localization parameter, ρ, is used to fit the cumulative Weibull distribution to the linear regression straight by using the SOLVER routine operator of the Microsoft Excel software. Then, the Weibull distribution of a sample is plotted in the coordinate system (x, y) following Equation A-5.

From Equation A-4, it can be seen that the shape parameter, k, approximates the slope of the straight regression line.

Finally the scale parameter, λ, approximation is obtained by Equation A-2, where it can be noticed that it is a function of the intercept of the linear Equation A-4 and the shape parameter, k. A more detailed description of this method can be found in [251] [252].

$$f(t) = \begin{cases} \frac{k(t-\rho)^{k-1}}{\lambda^k} e^{-(t-\rho/\lambda)^k} & t \geq 0 \\ 0 & t < 0 \end{cases} \quad \text{A-1}$$

Where,

t: is the variable studied.

ρ: is a location parameter which indicates the moment from which the distribution is generated. This parameter serves to fit the cumulative Weibull distribution function $F(t)$ to the regression line through the SOLVER operator of the Excel software.

k: is the shape parameter which serves to adjust the slope of the probability curve (Equation A-4) to a certain event. In this study, the events are the size (minor Feret) and the aspect ratio of the particle´s samples.

λ: is the scale parameter which is calculated using the next Equation:

$$\lambda = e^{-\left(\frac{k\ ln\lambda}{\mathrm{k}}\right)} \quad \text{A-2}$$

The cumulative Weibull distribution curve can be expressed as follow:

$$F(t) = 1 - e^{-\left(\frac{t-\rho}{\lambda}\right)^{k}} \quad \text{A-3}$$

The double logarithmic transformation of the cumulative function of the Weibull probability, F(t), can be written as a linear equation as follow:

$$F(t) = \ln(-\ln[1 - f(t)]) = \ k \ln\,(t - \rho) - \ k\ ln\lambda \quad \text{A-4}$$

The coordinate system values are:

$$\begin{cases} x = \ln(t - \rho) \\ y = \ln(-\ln[1 - f(t)]) \end{cases} \quad \text{A-5}$$

A.1.4 Analysis of the Particle Size Distribution through different techniques

As explained in Section 3.1, an Image analysis techniques has been employed to obtained the Particles Size Distribution (PSD) of the particle´s samples (which in turn have been characterized by a Weibull distribution curve), assuming that it is more accurate than the sieve size, since this latter is no sensitive to shape variation [182].

However, it is convenient to evaluate, with a view to future research, the differences between the gradation curves determined by Image Analysis techniques and from sieve analysis, since they might give different results and therefore, different

statistical parameters of the Weibull distribution curve; in order to take them into account when entering the distribution curve parameters of the stone sizes into the aggregate generation model presented in Chapter 4.

The different methodologies used for this analysis are: an image-based analysis through a low cost vision system, the ImageJ software, following the procedure explained in Appendix A.1.1; sieving in the laboratory; and the Digital Dynamic Image Analysis through a Camsizer device, the operation of which was explained in Section 2.2.3.

Since some research found some inconsistencies between the gradation (minor Feret) curves obtained by an Image-based analysis based on the number of particles and an Image-based analysis by volume [147][253], in the present work both gradations will be plotted to draw the author's own conclusion. The PSD number-based approach has been converted into a PSD mass-based following the procedure described in Section A.1.4.1.

For all the different technique employed, the same stones samples have been used which are: L2, L3, L4 and L5, (already described in Section 3.1.1). Those limestone samples correspond to those used to manufacture the asphalt mixtures (see Section 3.3). Since the aim of this study is to evaluate differences between the different methods employed by different researchers and not to evaluate the accurate of the Image Analysis through the ImageJ software, four samples set were considered enough. The samples were washed and dried before the analysis to eliminate the dust and small particles.

The comparison has been made by plotting the cumulative frequency distribution curves and calculating their Weibull distribution parameters.

A.1.4.1 Conversion of the PSD number-based into PSD mass-based of the ImageJ software Analysis

The PSD number- based has been converted into a PSD mass-based following the next procedure:

Equation A-6 is used to obtain the mass of the samples, assuming that the volume of the particles is similar to the volume of an ellipse (Equation A-7), which is the most appropriate according to previous research [147].

$$\sum_{i}^{n} m_i = \rho V_i \qquad \text{A-6}$$

Where,

m_i is the mass of the i_{th} particle;

ρ is the density, assumed 2.7 kg/cm3 for all the samples;

V_i is the volume of the i_{th} particle, defined following the next equation:

$$V = \frac{4\pi}{3} x \frac{a}{2} x \frac{b}{2} x \frac{c}{2} \qquad \text{A-7}$$

Where,

a is the mayor Feret diameter (largest dimension);

b is the intermediate axis in three dimensional (which correspond to the minor Feret in 2D projection) and

c is the thickness of the stones.

The area of an ellipse (placed in its more stable position) is given by Equation A-9 and therefore the Volume of an ellipse can be re-written as follow:

$$V = \frac{4}{3} x\, A\, x \frac{c}{2} \qquad \text{A-8}$$

Where,

$$A = \pi \times \frac{a}{2} \times \frac{b}{2} \qquad \text{A-9}$$

The area of the particles can be measured directly by the ImageJ software, which are placed in their most stable position, as mentioned. However, the third dimensions, c (the thickness) of the particles cannot be measured by the Image-based technique. To overcome this, c is determined as a direct relationship with the remaining axes of the particle: a or b, as shown by the next equations A-10 and A-11; and assuming that the shape characteristics are the same in all the samples since they are coming from the same source [147].

$$\alpha = \frac{c}{b}; \qquad \beta = \frac{c}{a} \qquad \text{A-10 \& A-11}$$

Where,

β is a constant which represent the ratio between the thickness and the major Feret diameter, which serves to adjust the third dimension that cannot be measured through the Image-based technique (see Table 3-2).

α is a constant which represent the ratio between the thickness and the minor Feret diameter.

Then, the axis relationships that offer the best adjustment are chosen, as will explain later. Thus, the volume can be defined as follow:

$$V = \frac{4}{3} x\ A\ x \frac{\beta b}{2} \quad or \quad V = \frac{4}{3} x\ A\ x \frac{\alpha b}{2} \qquad \text{A-12}$$

To obtain the constants β and α of the Equation A-10 and A-11 respectively; all three dimensions of 50 particles from each sample have been measured with a caliper.

Then, the volume (Equation A-12) has been calculated for both constant β and α. The comparison has been achieved by implementing the different volume obtained in Equation A-6, to see which axis relation gives the better match, based in the real sample weight (m_i), previously measured in the laboratory.

It was found that the constant β gives better results than those obtained by α and that they were very close to those obtained experimentally (about 7% and 14% higher for the smallest size samples and the biggest size samples respectively). This could be probably because of the volume of the particles is smaller than an ellipse, due to the stones might present shapes with concave surfaces that the original approach didn't allow for.

Finally, the β factor was adjusted to match the assumed mass with the real weight of the samples to fine tune of the adjustment when comparing the PSDs. As expected, the adjusted β is smaller than the original one obtained by using a calliper.

A.1.4.2 Results

This section presents a comparison between the gradation curves obtained from the different methodologies employed in this analysis, as well as the statistical

parameters of the Weibull distribution (shape and scale) for the four limestone samples: L2, L3, L4 and L5. The gradation curves obtained from the optical techniques are based on the minor Feret diameter (see Section 2.2.6), while the gradation curves obtained with sieve analysis depends on the opening of the sieves.

In Figure A-7, the cumulative distribution curves fitting between the different PSD models are plotted, which include: the Image Analysis based on the number of particles (ImageJ number-based); the Image Analysis based on the volume (ImageJ volume-based); the sieve Analysis (Laboratory sieve) and the Dynamic Image Analysis through the Camsizer Analyser device (Camsizer). Further Table A-1 shows the Statistical parameters of the Weibull distribution for each technique.

Figure A-7 –Comparison between the different PSD curves obtained with the different techniques

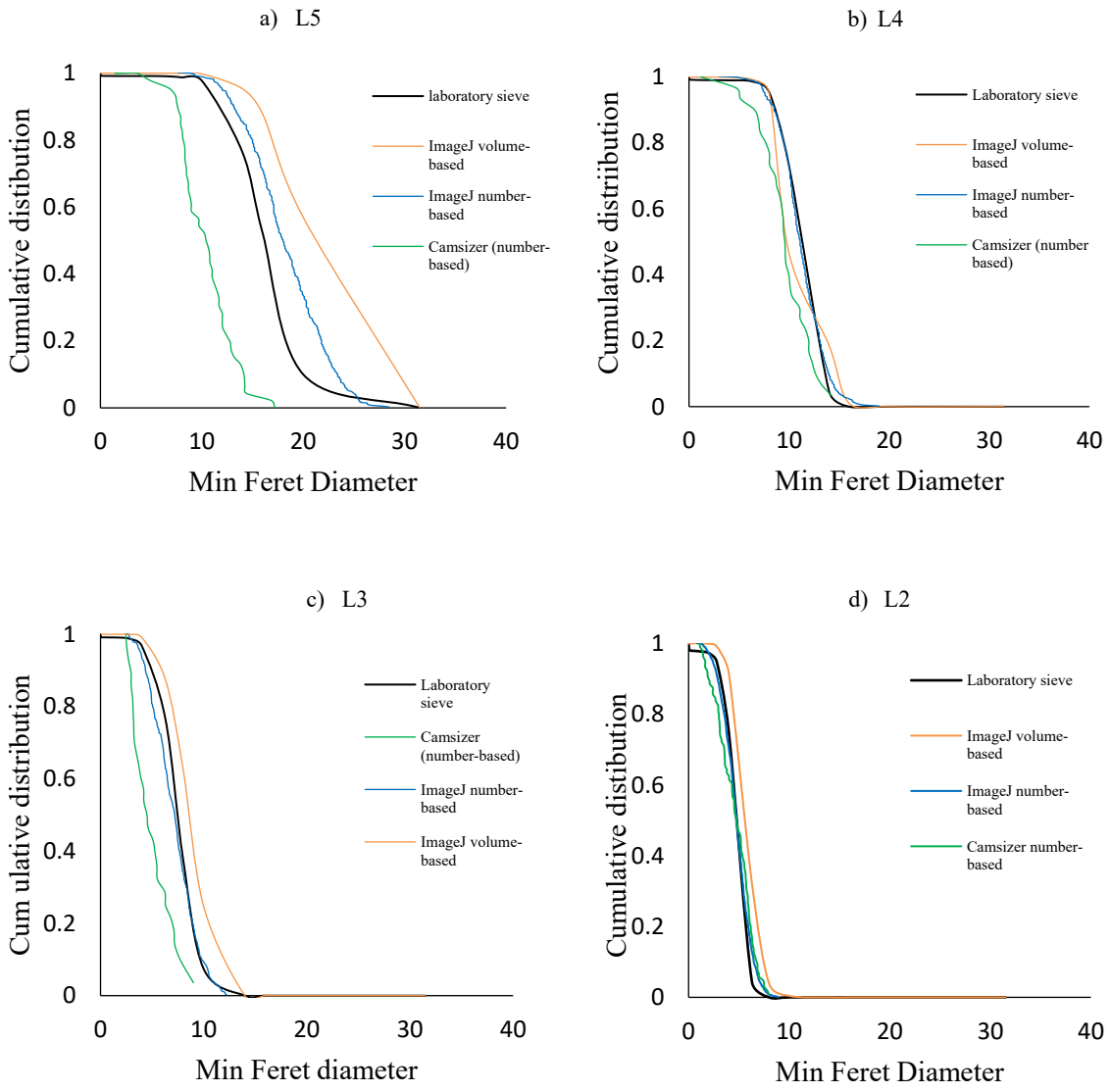


Table A-1 – Statistical parameters of the Weibull distribution curves

Techniques employed	Weibull parameters	L2	L3	L4	L5
Laboratory sieve	k_{MF}	2.8	4.61	6.57	5.75
	λ_{MF} (mm)	4.29	7.16	10.18	15.72
ImageJ Volume-based	k_{MF}	3.8	4.9	5.04	10.75
	λ_{MF} (mm)	5.19	8.3	11.8	20.15
ImageJ number-based	k_{MF}	3.77	4.01	5.43	5.26
	λ_{MF} (mm)	5.23	7.81	12.07	18.78
Camsizer	k_{MF}	2.59	3.48	3.96	4.32
	λ_{MF} (mm)	5.1	5.48	10.12	11.37

From the graphs plotted in Figure A-7, it can be noticed that all the samples follow a similar relationship pattern:

The comparison shows that, in general, the cumulative distribution curves obtained by the optical analysis through ImageJ overestimated the percent passing for a given grain size, as compared to the sieve analysis, which was also corroborated in [254].

Another observation is that the volume-based curves obtained by the ImageJ technique and the mechanical sieving do not match. This fact could be due to: i) the assumption of considering the stones as ellipses to find their volume and ii) that the size obtained by sieving depending on how particles pass through sieves; that is, which particle dimension(s) control the passage – they may not be the medium axis (or the minor Feret in particle projection measured through ImageJ) –. The particle-mass estimation with some assumption described in Section A.1.4.1 needs to be improved to match both curves.

Regarding the differences between the Image Analysis volume-based (yellow curve) and the Image Analysis number-based (blue curve), from Figure A-7 it is

observed that the volume-based curves overestimated the particle´s size number-based curves, as was also found in [253].

The grading curves obtained from the video grain size analyzer, Camsizer (green curve), tend to underestimate the grains sizes as compared to both: the sieve analysis and the Image Analysis through the ImageJ software. This could be attributed to the actual configuration of the software that the Camsizer has, which use a "detection measurement" through the imaging module and does not consider the entire range of sizes within a coarse sample, as the number of stones it analyzes is limited. Thus, if the sample contains fine particles due to stone breakage or dust, it is likely that most of the particles measured are those smaller particles giving thus, a poor representative value for that specific aggregate size. The fact of how the images of the stones is captured also influences, since the dimension of the stone that is considered to evaluate its size is that perpendicular to the image module and therefore, could vary.

In general, the gradation curves constructed from the mechanical sieve and the number-based curves are quite similar (blue and yellow curves respectively), despite of the fact that sieve analysis measures the percentage of particles, by weight, passing a given size, whereas ImageJ number-base measures percentage of number of particles of a given size compared to the total number of particles.

One explanation could be that the study has been made for samples with a specific coarse size fraction. A more pronounced difference may occur if the study would have been done for samples with a larger size range. An example of that can be seen in Figure A-7 a) where the ImageJ number-based overestimates the sizes of the particles as compared to the sieve analysis. However, in Figure A-7.d all the curves are in good agreement, surely due to the L2 sample is the one with the smallest size range.

In general, it is observed that the particle size range strongly affects the results obtained, since the differences between curves increase when the size range increases.

This same phenomenon is also observed regarding the statistical parameters of the Weibull distribution curves obtained by the different techniques, plotted in Table A-1.

Table A-1 shows a small difference between the statistical parameters obtained from the different methodologies in sample L2, whereas biggest differences appear as the size of the samples increases. In general, the Weibull parameters, shape and scale, obtained by mechanical sieve and by ImageJ have similar values for the samples L2, L3 and L4. However, the shape and scale values from the Camsizer are between two and eight units smaller compared to the other techniques (i.e. reflecting underestimation of particle size).

Therefore, it can be said that the sieve gradation is governed by the length of the intermediate axis (or Minor Feret diameter in a particle's projection) due to the proximity between the data obtained from the mechanical sieve and the ImageJ.

According to this finding, the implementation of the Weibull parameters from an Image Analysis based in a number percent in both: the virtual aggregate generation model (Chapter 4) and the porosity prediction model for porous asphalt (Chapter 6), is not a problem when working with aggregate material subject to a certain size range, since the Weibull statistical data obtained from the ImageJ test based on the number of particles were very similar to those obtained by sieving.

A.2 Analysis of the real aggregate´s packing specimens

A.2.1 Aggregate´s segregation and orientation of the various size aggregate´s packings specimens

In Section 3.2.3.3, the aggregate´s segregation and orientation of three various sizes aggregate´s packing specimens, which different aggregate´s composition and maximum aggregate´s size, were analyzed, to see how their different size´s ratios could influence in the particle´s distribution.

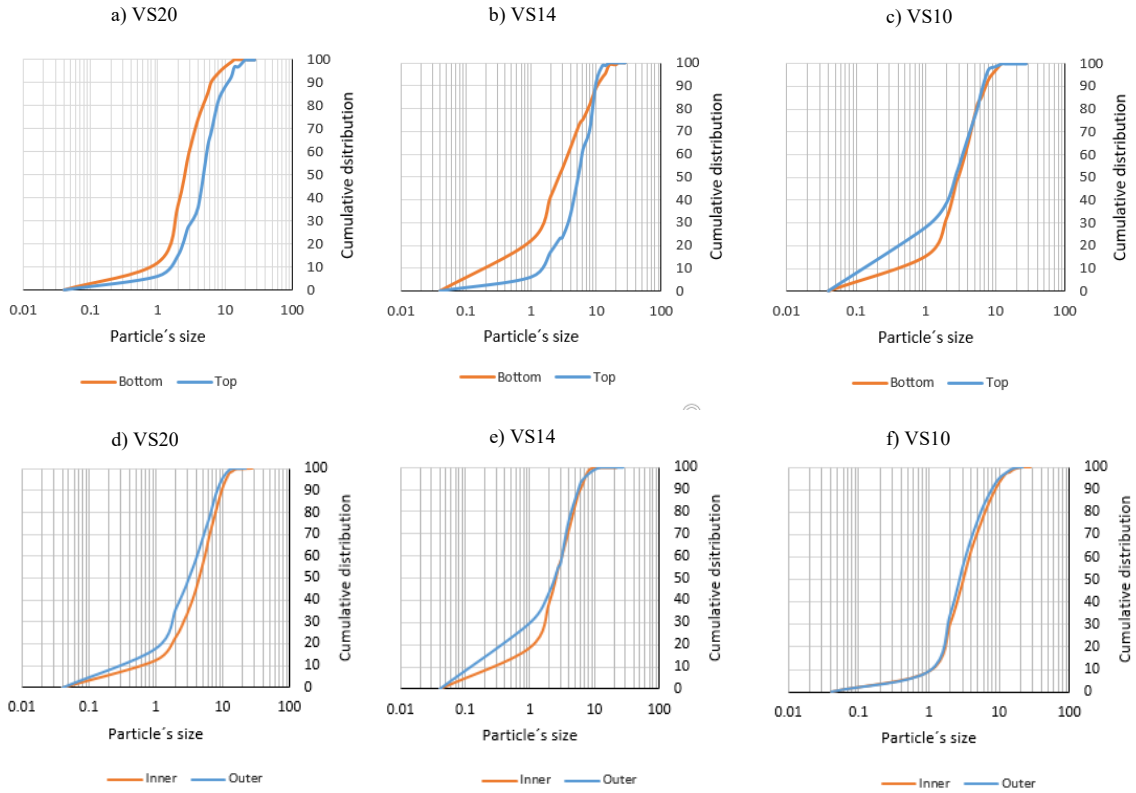
The three various size specimens are labeled: VS20, VS14 and VS10, which have maximum aggregate´s size of 20mm, 14mm and 10mm respectively. Their gradation curves are plotted in Figure 3-3.

The vertical and radial segregations were studied through the ratio of average aggregate size between sections, as explained in Section 3.2.3.3.

It was found that the samples VS20 and VS14 have larger aggregate at the top, (due to the smaller particle tend to move toward the bottom, by sneaking between the big one); whereas the sample VS10 presents less segregation than the other samples. Besides VS10 presents larger aggregate at the bottom. This information has been corroborated by plotting their respective aggregate distribution curve in vertical direction, as can be seen in Figure A-8.a; b and c.

The particles distribution curves in radial direction (Figure A-8..d, e and f); also confirm the finding explained in Section 3.2.3.3, regarding that all the various size samples present larger stones in the center of the sample. Nevertheless, the aggregates distributions in radial direction in all the samples analysed do not present a pronounce segregation.

Figure A-8 – The aggregate distribution in vertical and radial direction for the three specimens



Note: a, b and c, show the aggregate distribution in vertical direction for the specimens analyzed, while Figure A-7.d, e and f; detailed the aggregate distribution in radial direction for the same specimens.

Concerning the particle´s orientation, those were analyzed through the average orientation angle of the grains ($\overline{\varepsilon^{o}}$) in both directions: radial and vertical; which indicated that most of the particles were randomly oriented (see Section 3.2.3.3).

Nevertheless, it was found that the particles angle´s orientation present bigger difference between stones in radial direction, compared to the vertical one. For this reason, the gradient of the distribution of the angle´s orientation as a function of the particles size has been calculated for the radial direction. The results are shown in Table A-2.

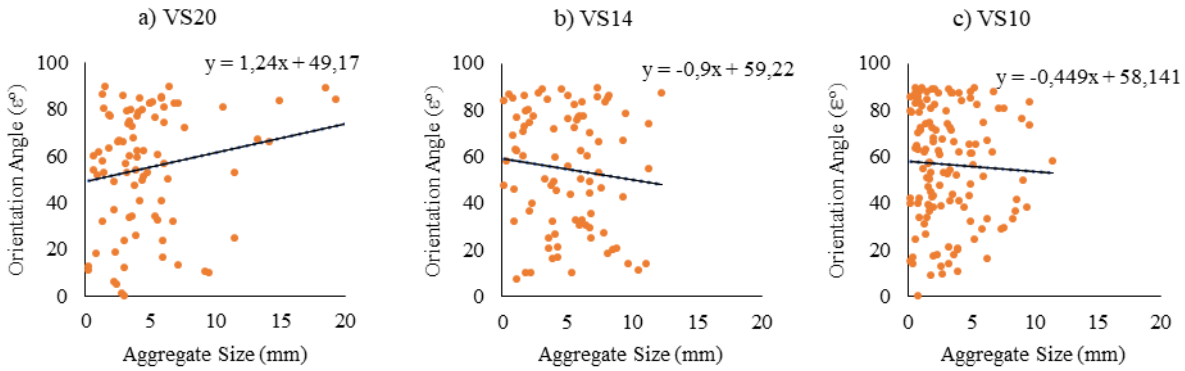
Table A-2 –Average Gradient ($(\overline{\varepsilon^{o}})$/mm) of Particle Orientation Angle versus Particle Size For the radial direction

	Radial direction	
Specimen by Nominal Size	Outer	Inner
20mm	3.62	0.14
14mm	2.18	-1.72
10mm	1.24	-0.83

From the results we observe a positive value in the outer section, which indicate a trend to a circumferential alignment when the particle size increase, probably due to the aspect ratio of the bigger particles are more pronounced and therefore, the directionality of the particle is more noticeable. On the other hand, the gradient in the inner section is less pronounced even reaching negative values (sample VS14). This indicates that the stones in the center tend to a radial distribution.

Figure A-9 shows the detailed distribution of the orientation angle for a middle image slice of the core sample, for each of the specimens analyzed. It could be appreciated a considerable dispersion as not all the sample´s slices from the CT-scan were analyzed. Nevertheless, it can be noticed a different trend over the samples.

Figure A-9 –Aggregate´s Orientation Angle, εº, as a function of the aggregate´s size



Note: measures taken from an image of the middle section of the specimen´s cores.

Figure A-9.a shows that the VS20 sample has the slope of the linear graph more pronounced and with a positive value, indicating that the largest particles tend to be circumferentially aligned. This could be because the effect of the lateral confinement of the stones is more noticeable in larger particles. Whereas, the VS14 sample (Figure A-9.b), presents a negative gradient with a more random distribution. From the Figure A-9.c (VS10 sample); it is observed that the slope of the linear graph is the least pronounced and also with a negative value. The random orientation tendency could lie on the fact that the smaller particles have a less pronounced orientation, due to their lower aspect ratio.

A.3 Analysis of the real asphalt mixtures specimens

A.3.1 Aggregate´s gradation of the asphalt mixtures

Table A-3 – Aggregate grading and binder content used to produce porous asphalt mixtures

Max aggregate size	Mix 1	Mix 2	Mix 3	Mix 4	Mix 5	Mix 6	Mix 7	Mix 8
Target porosity	10%	21%	10%	21%	10%	26%	5%	20%
sieve size (mm)								
31.5	100	100	100	100	100	100	100	100
20	100	100	100	100	100	100	98	96
16	100	100	100	100	100	100	90	77
14	100	100	100	100	97	96	84	65
10	100	100	97	96	56	34	65	52
8	100	100	71	65	37	14	55	47
6.3	98	97	38	25	28	11	48	41
4	56	44	23	12	23	8	36	24
2.8	37	21	19	11	20	6	30	17
2	30	15	17	10	18	5	26	14
1	21	11	13	7	13	4	19	10
0.5	16	9	9	6	10	3	14	8
0.25	12	7	7	5	7	2	11	6
0.125	10	6	6	4	6	2	8	5
0.063	8	5	5	3	5	2	7	4
Binder content (%)	6	5.3	5.5	4.3	5	4	4.7	5

A.3.2 Bulk density and Air Void content measurement procedure

The bulk density of bituminous specimens is defined as the mass of the sample per unit volume, including the air voids content.

The simulation model presented in Chapter 6 is based on the final height of the compacted samples and the dimensions of the cylindrical mould that contains them. Therefore, the bulk density of compacted specimens were determined using the data output generated by the gyratory compactor (ρ_{bgyro}).

This variable is back calculated by the Superpave gyratory compactor based on: the recorder specimen heights, which in turn is determined by the position of the base of the top and bottom plates placed in the mould previous the compaction process, the mass of the samples and the dimensions of the mould (Eq. A-6).

$$\rho_{bgyro} = \frac{m_1}{\frac{4}{3} x\ h\ x\ d^2} \quad \text{A-13}$$

Where,

ρ_{bgyro} is the bulk density of the compacted specimen calculated by the gyratory compactor, in Megagram per cubic meter (Mg/m^3);

m_1 is the mass of the dry specimen, in gram (g);

h is the height of the compacted specimen calculated by the gyratory compactor;

d is the diameter of the cylindrical mould.

However, uncertainty caused by this test procedure is not desired and, due to the density plays a very important role on the AV content calculation, the same six cylindrical cores for each mixture´s design were analysed by laboratory tests using the sealed samples techniques (ρ_{bsea}), according to the BS EN 12697-6:2012 [179], since it is the most appropriate for open coarse surface. This will serve to see if exist under a significant difference in the air void results for both techniques. Shrinkage foil to cover the mixtures was used for this process. The formula for its calculation can be seen in Eq.A-7.

$$\rho_{bsea} = \frac{m_1}{(m_2 - m_3)/\rho_w - (m_2 - m_1)/\rho_{sm}} \quad \text{A-14}$$

Where,

ρ_{bsea} is the bulk density of the sealed specimen, in Megagram per cubic meter (Mg/m^3);

m_1 is the mass of the dry specimen, in gram (g);

m_2 is the mass of the dry sealed specimen, in gram (g);

m_3 is the mass of the sealed specimen in water, in gram (g);

ρ_w is the density of the water, in Megagram per cubic meter (Mg/m^3);

ρ_{sm} is the density of the sealing material, in Megagram per cubic meter (Mg/m^3);

The Air Void content of each test specimen was calculated based on the previous calculation of the maximum and bulk densities, as follows:

$$\rho_{bsea} = \frac{m_1}{(m_2 - m_3)/\rho_w - (m_2 - m_1)/\rho_{sm}} \quad \text{A-15}$$

Where,

AV is the percentage of the Air Void content in the volume of the mixture;

ρ_m is the maximum density of the mixture, which in this case correspond to the maximum density of the mixture by calculation ρ_{mc}, in Megagrams per cubic metre (Mg/m^3);

ρ_b is the bulk density of the mixture, in Megagram per cubic meter (Mg/m^3);

A.3.3 Air void and standard deviation obtained from the sealed asphalt specimens and the SGC.

A.3.3.1 Sealed asphalt mixtures specimen´s results

The next tables present the results obtained from the laboratory tests for the sealed specimens.

Table A-4 – Results of Mix 1

Number of samples	1	2	3	4	5	6
m_1 *(gr)*	1045.8	1048.1	1043.5	1044.7	1048.4	1042.98
m_2 *(gr)*	1051.3	1053.97	1048.2	1048.1	1053.2	1047.64
t º	20.9	20.9	20.9	20.9	20.9	20.9
ρw (Mg/m3)	0.998	0.998	0.998	0.998	0.998	0.998
m_3	560.88	571.16	556	565.7	557.8	568.69
ρ_{sm} *(Mg/m3)*	1.668	1.668	1.668	1.668	1.668	1.668
ρ_{bsea} *(Mg/m3)*	2.143	2.183	2.128	2.171	2.125	2.186
ρ_{max}	2.461	2.461	2.461	2.461	2.461	2.461
Air Void (%)	12.93	11.31	13.52	11.80	13.67	11.17

Table A-5 – Results of Mix 2

Number of samples	1	2	3	4	5	6
m_1 *(gr)*	920.6	918.8	914.9	919.9	920.4	922.1
m_2 *(gr)*	925	923.3	919.4	926.4	919.2	926.9
t º	20.9	20.9	20.9	20.9	20.9	20.9
ρw (Mg/m3)	0.998	0.998	0.998	0.998	0.998	0.9980
m_3	458.4	458.7	456	455.7	457.8	454
ρ_{sm} *(Mg/m3)*	1.668	1.668	1.668	1.668	1.668	1.668
ρ_{bsea} *(Mg/m3)*	1.980	1.985	1.982	1.967	1.988	1.958
ρ_{max}	2.486	2.486	2.486	2.486	2.486	2.486
Air Void (%)	20.34	20.14	20.27	20.88	20.04	21.24

Table A-6 – Results of Mix 3

Number of samples	1	2	3	4	5	6
m_1 *(gr)*	1052.3	1051.34	1055.73	1039	1038	1031
m_2 *(gr)*	1055.8	1058.8	1060.21	1042.2	1042.43	1034.9
t °	20.9	20.9	20.9	20.9	20.9	20.9
ρw (Mg/m3)	0.998	0.998	0.998	0.998	0.998	0.998
m_3	568	569	565.35	562.758	566.04	560.32
ρ_{sm} *(Mg/m3)*	1.668	1.668	1.668	1.668	1.668	1.668
ρ_{bsea} *(Mg/m3)*	2.162	2.162	2.140	2.171	2.186	2.179
ρ_{max}	2.48	2.48	2.48	2.48	2.48	2.48
Air Void (%)	12.80	12.81	13.67	12.43	11.81	12.13

Table A-7 – Results of Mix 4

Number of samples	1	2	3	4	5	6
m_1 *(gr)*	942.2	948.1	944.4	933.5	900	940
m_2 *(gr)*	945.8	952.7	948.9	937.8	933.8	945.2
t °	20.9	20.9	20.9	20.9	20.9	20.9
ρw (Mg/m3)	0.998	0.998	0.998	0.998	0.998	0.998
m_3	462.9	476.5	451.2	465.6	464.3	454.7
ρ_{sm} *(Mg/m3)*	1.668	1.668	1.668	1.668	1.668	1.668
ρ_{bsea} *(Mg/m3)*	1.956	1.998	1.904	1.983	1.999	1.924
ρ_{max}	2.526	2.526	2.526	2.526	2.526	2.526
Air Void (%)	22.56	20.87	24.61	21.45	20.84	23.79

Table A-8 – Results of Mix 5

Number of samples	1	2	3	4	5	6
m_1 *(gr)*	1052.1	1059.23	1057.4	1057	1049	1047.8
m_2 *(gr)*	1056.5	1061.82	1064.48	1060.72	1050.84	1053.67
t °	20.9	20.9	20.9	20.9	20.9	20.9
ρw (Mg/m3)	0.998	0.998	0.998	0.998	0.998	0.998
m_3	568	569.8	563.77	569.5	574.5	566.3
ρ_{sm} *(Mg/m3)*	1.668	1.668	1.668	1.668	1.668	1.668
ρ_{bsea} *(Mg/m3)*	2.161	2.155	2.125	2.157	2.203	2.161
ρ_{max}	2.487	2.487	2.487	2.487	2.487	2.487
Air Void (%)	13.09	13.32	14.52	13.25	11.41	13.09

Table A-9 – Results of Mix 6

Number of samples	1	2	3	4	5	6
m_1 *(gr)*	887.4	883.9	884.6	880	870	874
m_2 *(gr)*	992.3	987.5	988.6	983.11	973.8	978.6
t °	20.9	20.9	20.9	20.9	20.9	20.9
ρw *(Mg/m3)*	0.998	0.998	0.998	0.998	0.998	0.998
m_3	468	469.8	463.77	460.5	469.5	466.3
ρ_{sm} *(Mg/m3)*	1.668	1.668	1.668	1.668	1.668	1.668
ρ_{bsea} *(Mg/m3)*	1.919	1.935	1.908	1.905	1.963	1.939
ρ_{max}	2.536	2.536	2.536	2.536	2.536	2.536
Air Void (%)	24.32	23.66	24.74	24.85	22.561	23.51

Table A-10 – Results of Mix 7

Number of samples	1	2	3	4	5	6
m_1 *(gr)*	1109	1125	1120	1117	1107	1118
m_2 *(gr)*	1112.8	1123	1123.5	1122.5	1113.2	1125.2
t °	20.9	20.9	20.9	20.9	20.9	20.9
ρw *(Mg/m3)*	0.998	0.998	0.998	0.998	0.998	0.998
m_3	639.3	641.4	629.6	631.5	643.8	642
ρ_{sm} *(Mg/m3)*	1.668	1.668	1.668	1.668	1.668	1.668
ρ_{bsea} *(Mg/m3)*	2.348	2.325	2.272	2.285	2.372	2.330
ρ_{max}	2.507	2.507	2.507	2.507	2.507	2.507
Air Void (%)	6.305	7.231	9.335	8.818	5.3627	7.056

Table A-11 – Results of Mix 8

Number of samples	1	2	3	4	5	6
m_1 *(gr)*	924	930.3	937.04	938.1	941.7	940.08
m_2 *(gr)*	929.2	934.3	940.3	944.7	945	945.6
t °	20.9	20.9	20.9	20.9	20.9	20.9
ρw *(Mg/m3)*	0.998	0.998	0.998	0.998	0.998	0.998
m_3	460.12	473.8	477.9	475.9	482.3	487.2
ρ_{sm} *(Mg/m3)*	1.668	1.668	1.668	1.668	1.668	1.668
ρ_{bsea} *(Mg/m3)*	1.979	2.026	2.034	2.014	2.040	2.06
ρ_{max}	2.491	2.491	2.491	2.491	2.491	2.491
Air Void (%)	20.54	18.63	18.45	19.140	18.10	17.23

A.3.3.2 Superpave Gyratory Compactor´s results

The next tables present the results obtained from the back calculation of the SGC.

Table A-11 – Result of Mix 1

Number of samples	1	2	3	4	5	6
Mass (gr)	1045.8	1048.1	1043.5	1044.7	1048.4	1042.98
Height (cm)	6.12	6.28	6.2	6.147	6.18	6.07
ρ_{bgyro} *(Mg/m3)*	2176.84	2126.04	2144.03	2165.00	2161.07	2188.85
Air Void (%)	11.54	13.61	12.87	12.02	12.18	11.05

Table A-12 – Result of Mix 2

Number of samples	1	2	3	4	5	6
Mass (gr)	928.8	920.9	929.9	922.4	925.1	925
Height (cm)	5.935	5.995	6.01	5.988	5.86	5.996
ρ_{bgyro} *(Mg/m3)*	1993.57	1956.83	1971.02	1962.31	2011.04	1965.21
Air Void (%)	19.80	21.28	20.71	21.06	19.10	20.94

Table A-13 – Result of Mix 3

Number of samples	1	2	3	4	5	6
Mass (gr)	1052.3	1051.34	1055.73	1039	1038	1031
Height (cm)	6.25	6.18	6.17	6.215	6.172	6.345
ρ_{bgyro} *(Mg/m3)*	2144.81	2167.13	2179.70	2129.63	2142.40	2069.93
Air Void (%)	13.51	12.61	12.10	14.12	13.61	16.53

Table A-14 – Result of Mix 4

Number of samples	1	2	3	4	5	6
Mass (gr)	942.2	948.1	944.4	933.5	900	940
Height (cm)	6.301	5.97	6.18	6.07	5.925	6.192
ρ_{bgyro} *(Mg/m3)*	1904.86	2023.06	1946.69	1959.09	1935.01	1933.86
Air Void (%)	24.58	19.91	22.93	22.44	23.39	23.44

Table A-15 – Result of Mix 5

Number of samples	1	2	3	4	5	6
Mass (gr)	1052.1	1059.23	1057.4	1057	1049	1047.8
Height (cm)	6.126	6.149	6.214	6.255	6.246	6.17
ρ_{bgyro} *(Mg/m3)*	2187.81	2194.40	2167.69	2152.67	2139.45	2163.33
Air Void (%)	12.03	11.76	12.838	13.44	13.97	13.014

Table A-16 – Result of Mix 6

Number of samples	1	2	3	4	5	6
Mass (gr)	887.4	883.9	884.6	880	870	874
Height (cm)	6.05	5.8	5.897	5.968	5.957	5.928
ρ_{bgyro} *(Mg/m3)*	1868.50	1941.35	1910.93	1878.38	1860.46	1878.16
Air Void (%)	26.320	23.44	24.64	25.93	26.63	25.93

Table A-17 – Result of Mix 7

Number of samples	1	2	3	4	5	6
Mass (gr)	1119.2	1117.9	1119.1	1118.1	1112.2	1121.6
Height (cm)	6.16	6.12	6.1	6.25	6.18	6.09
ρ_{bgyro} *(Mg/m3)*	2314.50	2326.92	2337.05	2278.92	2292.58	2346.12
Air Void (%)	7.47	6.92	6.51	8.84	8.29	6.15

Table A-18 – Result of Mix 8

Number of samples	1	2	3	4	5	6
Mass (gr)	924	930.3	937.04	938.1	941.7	940.08
Height (cm)	5.902	5.81	5.776	5.898	6.11	6.008
ρ_{bgyro} *(Mg/m3)*	1994.35	2039.75	2066.62	2026.16	1963.36	1993.26
Air Void (%)	19.93	18.11	17.03	18.66	21.18	19.98

APPENDIX B. VIRTUAL PARTICLE GENERATION MODEL

B.1 Volume Reconstruction from implicit shape´s representation

Several techniques for the surface reconstruction from a set of points exist, being the most common the construction of a triangular mesh, since it is the simplest and therefore the most efficient for the subsequent rendering process (or visualization); which is a very important requirement in the games engines, since they require real-time visualization. It is necessary to mention that the rendering process consists of the discretization of a surface boundary of a volume in smaller shapes (such as triangles) suitable for its visualization [255].

The most popular discretization methods are: Voxel-based approach, Delaunay-based approach and Voronoi-based approach [240]. However, the last two types of methods mentioned are mainly used when the input is an explicit shape, and not an implicit surface as is our case [256]; since the free- form shapes of the stones are drawn from scalar functions, as it was explained in Chapter 4.

Despite nowadays there are Delaunay approach and Voronoi approach for the reconstruction of implicit forms, as can be read in [240] and [257] respectively, those do not have an easy access as those for the voxel-based approach.

The voxel-based approaches discretize the domain of a scalar function on R^3 (f: $R^3 \rightarrow R$) into a group of cells, which conform the 3D grid. This process is named voxelization. Then, these cells are processed independently in sequential and traversal order in order to find the intersection between the grid and the input shape. The most famous algorithm of this type the Marching Cube (MC), created by Lorensen and Cline in 1987, reason that makes it easily accessible and for that, is the one chosen for the 3D

random aggregate generation model presented in this thesis. In addition the MC algorithm is a powerful surface rendering, well known for its time efficiency [258].

Nevertheless its biggest drawback consists in the possibility of the appearance of topological inconsistency or holes in the structure, due to its non-isotropic nature in three-dimensional space. However, many researchers propose different solution to solve these problems, for instance, Nielson (2003) use a tri-linear interpolation instead of a linear interpolation for calculating the intersection between the implicit shape and the grid, to eliminate the ambiguous face [259]. Other examples to improve Marching Cubes Algorithm can be seen in [260][261][262][263].

In summary, in the actual research we decided to use the Marching Cubes algorithm to model and visualize our virtual stones for three main reasons:

1) it is the most suitable for the conversion from implicit surface to explicit volume (polyhedrization) [257];

2) it has a good performance in terms mesh quality and computational cost, allowing a quick surface reconstruction suitable for real-time visualization [258] and

3) it has an extensive documentation available and an easy and free access to the source codes in C-sharp language, which is the language used by the platform where our simulation is developed, Unity 3D. The code can be download from the page:

https://graphics.stanford.edu/~mdfisher/MarchingCubes.html

B.1.1 The marching Cubes algorithm

The principal objective of the Marching Cube algorithm (MC) is to generate a triangular mesh of a volumetric data along an equidistant lattice (or 3D grid) to form the geometry of the object being modelled [264].

When the input is an implicit shape in a scalar field, this method subdivides the domain of a scalar function on R^3 (f: $R^3 \rightarrow R$) into a group of cells with the same size which conform the 3D grid. These cells are processed independently in order to find the intersection between the boundary of the shape and the cells, to output its volumetric representation by constructing a final polygonal mesh. The conversion from implicit surfaces to explicit volume is called polyhedrization or mesh generation [257]. The surface of the object corresponds to an isosurface that satisfies f(x, y, z) = ISO, where ISO is a constant named isovalue, which has to be defined by the user.

The following paragraph explains how the MC works in Unity 3D – the platform chosen to create the model described here.

i. Input: implicit surface. To visualize a volume, it is necessary to establish scalar volumetric data by storing a collection of 3D points in Unity 3D as voxels, through the use of the types of equations: f(x, y, z) = ISO, already described. This type of function is used to know when a point is above or below the isosurface. In Unity, the scalar function is a density function which gives different levels of opacity to the centers of the voxels, as shown in Figure B- 1; where their representation are dots of different grey intensity. The shape is designed by selecting all the points (x, y, z) where the equation give the ISO value. In the example drawn in Figure B-1for the explanation, an equation of a sphere and an isovalue equal to zero has been chosen.

ii. Cubical grid (voxelization). Once the space where the objects are going to be generated has been chosen, a set of voxels are assembled to bound the 3D domain where the MC acts. A 3D grid with N x N x N cells (or voxels) that contains the implicit surface (a sphere in this case) is shown in Figure B-2. The number of voxels in the lattice defines the resolution and level of detail of the objects to be modelled. The centres of the voxels represent the vertices of the lattice where the MC algorithm will acts [265].

iii. The Marching Cube process

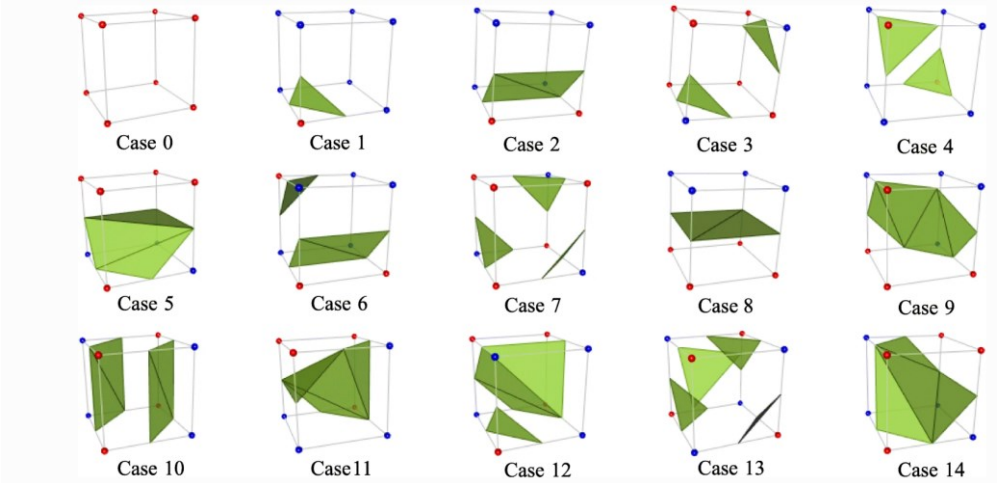
a. Determine the sign of the vertices of the lattice. The first step that MC does is to determine which vertices of the lattice are above, below or on the surface of the object to be generated. the MC evaluate the user-defined function and label each vertex assigning a density value to each voxel's centre (red numbers in Figure B-2)[266]. As we have previously explained, these points represent the vertices of each cell of the lattice, which can be positive (outside the isosurface) or negative (inside the isosurface). When two consecutive vertices have different signs, it means that between them the value zero exist (isovalue), located somewhere along the edge that joins them. Each cell is examined via a look-up table to know which vertices are positive and which are negative, encountering a total of 256 (2^8) possible cell configurations. However, discarding the redundant cases such as the mirror effects, the topologies configuration can be reduce to 15 cases [255], as is shown in Figure B-1.

b. Determine the locations of the isovalue points. As described in the previous paragraph, an edge of the lattice whose vertices have different sign contains an isovalue point belonging to the isosurface of the implicit shape. To determine its exact position along the edge, a linear interpolation is normally used, when only implicit function values are available [257]. Detailed information about how this interpolation is performed can be found in [267]. This process estimates the intersection between the implicit shape and the 3D lattice.

c. Determine the number of triangle to be generated. Once the cell's cases are known, the final step is to construct the triangle(s) in the correct order, depending on which edge of the cell the isovalue is located, by using another look-up table. MC traverses each cell's lattice sequentially to assign them one of the possible cases to generate triangle. Between 0 and 5 triangles can be generated, that is, each cell can have a maximum of 15 surface points. Finally, the triangles are connected together to construct the final polygonal mesh which represent the object´s surface (polyhedrization). From Figure B-2 it can be appreciate a green triangular mesh that represents the sphere.

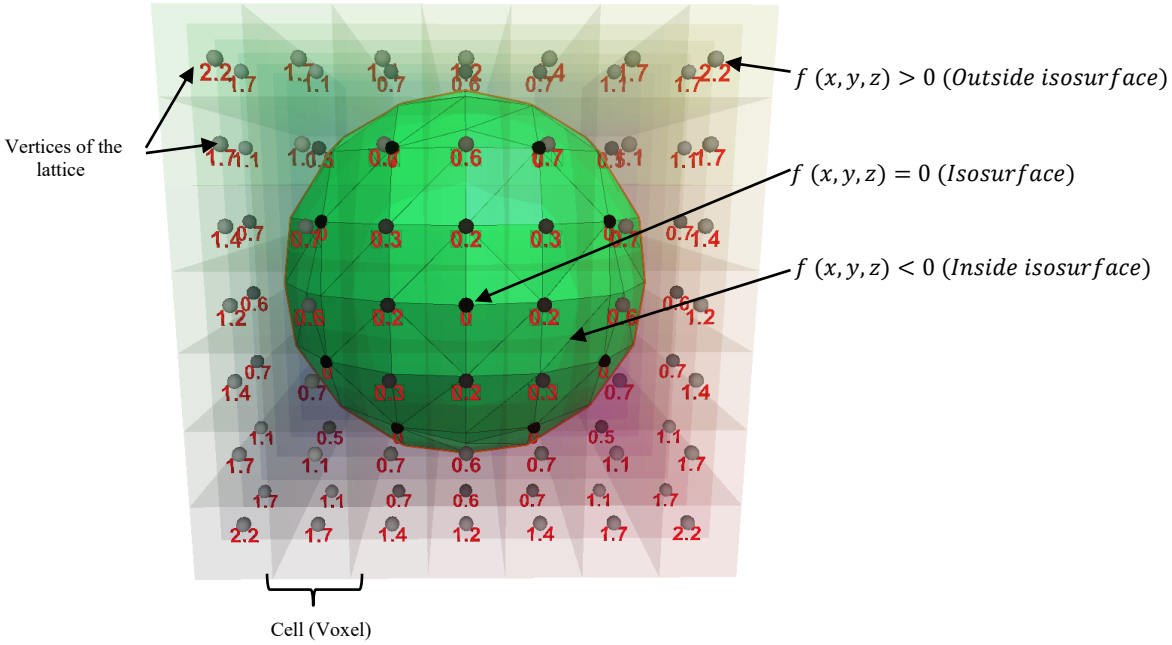
iv. Rendering for the volume visualization. Finally, the Mesh Renderer component of Unity 3D "facetizes" all the triangles to allow visualization of the final volume by computing the surface normal for the triangle´s vertices, necessary requirement for renderer shading.

Figure B-1 – Marching Cube cases from the lookup table



Note: Image from: An extended triangulation to the Marching Cubes 33 algorithm [268].

Figure B-2 – A spherical Particle created in Unity 3D software using the Marching Cube



B.2 Video of the creation of different type of particles

A video demonstration on creating particles with different levels of angularities can be seen at:

https://youtu.be/v30tYYiIw2Q

B.3 One sample T-test for the virtual aggregate generation model validation

The effectiveness of the aggregate generation model was evaluated using the one sample student t- test, which was chosen since it is appropriate for small sample´s size [269].

The one sample student t-test has been applied in order to evaluate if the morphological output parameter of our virtual aggregate generation model are significantly different from the actual one. This statistic test serve to compare one sample´s mean to a hypothetical value, which in this case is the real P_{er50} of the different shape´s factors measured from the real stone´s samples.

In this study, ten virtual particle´s samples of each type of stone were simulated to obtain the mean of their P_{er50} of each shape´s factors and test whether the mean of the measured variable is different from the null expectation. The null expectation as follow:

Null hypothesis H_0: μ=Real P_{50}

Alternative hypothesis H_1: $\mu \neq$Real P_{50}

The acceptance or rejection of the null hypothesis depend on whether the statistic parameter t_s is larger or smaller than the t critical (t_{crit}). The critical point (t_{crit}) according to the look up table is 2.262 [270].

Where the t parameter is defined as:

$$t = \frac{\bar{x} - \mu}{s/\sqrt{n}} \qquad \text{B-1}$$

Where,

$\overline{x}$: is the mean of the population, in this case, the mean of the P50 of 10 samples for each shape´s parameters.

μ: is the expected value, in this case the real P50 of the different shape´s parameters evaluated.

s: is the standard deviation.

n: the sample size.

The statistical concept required to state a decision are [271]:

- The test is a two tails test because we want to evaluate both: if the mean of a population ($\overline{x}$) is greater or less than the hypothetic value (μ).
- The degree of freedom is 9 (sample´s size (n) minus one).
- The level of significance is 5% (by convention).
- The critical Rejection of Null hypothesis if: $t \geq t_{crit}$

According to this statistical concept, if the t_s obtained is located outside the critical region, $t \leq t_{crit}$, the null hypothesis will be accepted and the alternative hypothesis rejected. Additionally, the p value has also been calculated [271], which indicate the percentage of probability that the difference between our sample's mean ($\overline{x}$) and the theoretical value (μ) is due to a random variability. The bigger is the p value, the bigger is the chance that this difference is due to a random variability and therefore less significant is the difference.

The following tables shown the one sample T-test for the different shape´s factors evaluated, for the different particle´s samples. The results reveals that for the shape factors: minor and major Feret diameters; aspect ratio, area and perimeter (Table B-1– Table B-4), the null hypothesis is accepted, indicating no significant difference between the real and the virtual data.

Whereas for the shape´s factors: roundness and circularity (Table B- 5 and Table B- 6), the null hypothesis is rejected for many of the samples. Furthermore, samples such as Slag (S) or Crushed Glass (CG) show very low P values, which, as already mentioned, indicates more significant differences. This could be due to the fact that they are the ones with the most complex shapes and therefore the most difficult to simulate.

Therefore, and optimization would be convenient in order to improve the model´s accuracy of certain parameters, as well as to make the aggregate´s generation model more functional, as is described in Chapter 6.

Table B-1 – T-test results with 95% Confidence Interval for Minor Feret

P_{er50} of 10 virtual samples	G1	G2	RG	L1	L2	L3	L4	L5	GS	CG	S
1	15.16	10.87	15.2	13.45	5.67	7.35	12.98	14.77	15.4	6.89	5.37
2	14.13	11.56	16.98	11.87	5.23	6.91	12.12	18.9	15.12	8.24	5.7
3	13.76	12.76	17.12	15.2	4.17	7.45	11.82	19.83	15.43	5.689	4.7
4	13.98	12.14	15.22	16.17	5.61	6.27	9.98	15.93	14.98	9.02	3.99
5	14.56	13.1	16.52	13.97	4.88	7.03	10.24	15.65	14.96	8.3	5.1
6	14.97	11.39	14.97	14.4	3.98	8.3	11.53	14.8	15.1	8.12	6.7
7	15.7	12.81	15.13	14.58	4.89	7.51	11.09	17.98	15.23	7.09	5.23
8	14.22	13.2	18.34	15.7	4.12	8.34	10.33	18.12	15.2	8.19	4.67
9	14.13	11.07	17.6	14.56	5.51	7.97	11.35	17.27	14.8	9.15	3.87
10	15.1	13.4	14.99	13.69	5.32	5.98	12.8	19.2	14.87	7.2	5.2
Sample´s mean ($\bar{x}$)	14.57	12.23	16.20	14.35	4.93	7.31	11.42	17.24	15.10	7.78	5.05
Real P_{Per50} value (μ)	14.62	12.36	16.82	14.68	4.76	7.19	11.16	17.8	15.12	8.04	4.92
Sample Standard Deviation (s)	0.63	0.94	1.25	1.22	0.64	0.79	1.04	1.85	0.21	1.06	0.82
Sample Size (n)	10	10	10	10	10	10	10	10	10	10	10
t	-0.246	-0.435	-1.544	-0.829	0.875	0.484	0.801	-0.946	-0.165	-0.747	0.511
P value	0.811	0.674	0.157	0.429	0.405	0.640	0.444	0.369	0.873	0.474	0.621

Table B-2 – T-test results with 95% Confidence Interval for Aspect Ratio

P_{er50} of 10 virtual samples	G1	G2	RG	L1	L2	L3	L4	L5	GS	CG	S
1	0.674	0.635	0.745	0.76	0.634	0.674	0.706	0.768	1	0.611	0.627
2	0.683	0.673	0.725	0.78	0.647	0.683	0.786	0.752	0.96	0.652	0.643
3	0.734	0.652	0.771	0.81	0.64	0.707	0.731	0.75	0.99	0.631	0.622
4	0.692	0.73	0.794	0.748	0.62	0.672	0.707	0.763	0.98	0.688	0.673
5	0.734	0.681	0.752	0.76	0.667	0.668	0.714	0.79	1	0.637	0.678
6	0.728	0.664	0.788	0.82	0.658	0.728	0.75	0.78	0.98	0.673	0.632
7	0.73	0.723	0.741	0.756	0.631	0.73	0.708	0.769	0.99	0.654	0.668
8	0.758	0.672	0.787	0.741	0.671	0.758	0.727	0.755	0.99	0.656	0.663
9	0.671	0.72	0.791	0.799	0.602	0.671	0.748	0.794	0.99	0.679	0.641
10	0.735	0.641	0.782	0.753	0.61	0.715	0.769	0.786	1	0.694	0.602
Sample´s mean (□)	0.71	0.68	0.77	0.77	0.64	0.70	0.73	0.77	0.99	0.66	0.64
Real P_{Per50} value (μ)	0.71	0.69	0.76	0.78	0.65	0.68	0.74	0.76	1	0.64	0.66
Sample Standard Deviation (s)	0.030	0.03	0.02	0.027	0.023	0.031	0.027	0.016	0.0121	0.026	0.024
Sample Size (n)	10	10	10	10	10	10	10	10	10	10	10
t	0.400	-1.0	0.96	-0.829	-1.62	2.06	-0.6	2.10	-2.0	2.09	-1.91
P value	0.697	0.342	0.357	0.428	0.137	0.068	0.556	0.064	0.075	0.065	0.088

Table B-3 – T-test results with 95% Confidence Interval for Area

P_{er50} of 10 virtual samples	G1	G2	RG	L1	L2	L3	L4	L5	GS	CG	S
1	1.85	1.27	4.98	1.39	0.37	0.61	1.25	4.29	1.84	1.06	0.291
2	2.78	1.08	3.15	2.78	0.42	0.84	1.18	6.72	1.98	1.13	0.467
3	2.72	2.14	2.86	2.73	0.17	1.02	1.41	4.14	1.87	0.62	0.359
4	2.77	1.525	2.31	4.07	0.36	0.47	0.98	2.83	1.87	0.66	0.213
5	1.875	0.87	3.98	3.06	0.216	0.54	1.246	3.14	1.85	0.92	0.124
6	0.883	1.87	1.79	2.46	0.35	0.61	1.12	5.38	1.78	0.78	0.472
7	2.16	1.02	3.14	1.24	0.29	0.38	0.951	1.51	2.1	1.03	0.357
8	1.0792	1.31	1.08	2.62	0.26	0.52	1.17	2.35	1.98	0.682	0.224
9	0.931	1.28	1.32	1.08	0.15	0.42	3.1	4.12	1.67	0.612	0.265
10	1.391	1.05	1.89	1.91	0.28	0.38	2.87	3.12	2.02	0.41	0.34
Sample´s mean ($\bar{x}$)	1.84	1.34	2.65	2.33	0.29	0.58	1.53	3.76	1.90	0.79	0.31
Real P_{Per50} value (μ)	2.08	1.56	2.73	2.02	0.24	0.52	1.28	3.12	1.83	0.64	0.25
Sample Standard Deviation (s)	0.756	0.397	1.220	0.9330	0.089	0.207	0.781	1.514	0.125	0.234	0.110
Sample Size (n)	10	10	10	10	10	10	10	10	10	10	10
t	-0.987	-1.727	-0.207	1.064	1.645	0.901	1.002	1.336	1.660	2.02	1.749
P value	0.349	0.118	0.840	0.315	0.134	0.391	0.342	0.214	0.131	0.074	0.114

Table B-4 – T-test results with 95% Confidence Interval for Perimeter

P_{er50} of 10 virtual samples	G1	G2	RG	L1	L2	L3	L4	L5	GS	CG	S
1	3.34	5.89	4.98	5.29	4.07	2.93	5.71	4.91	4.56	2.31	2.91
2	3.54	4.79	7.15	3.98	3.53	1.08	2.82	4.27	4.91	4.1	0.77
3	8.12	4.14	5.46	6.73	2.91	3.45	4.92	6.03	4.61	5.2	1.25
4	4.18	5.5	4.7	5.97	1.15	3.14	3.995	8.19	5.2	2.66	1.13
5	5.7	3.16	9.01	8.72	0.95	1.19	2.97	5.72	4.43	2.17	3.27
6	4.92	6.17	8.15	4.76	1.01	2.6	4.991	4.8	4.67	2.03	1.72
7	5.78	5.2	4.4	4.1	3.16	2.61	3.67	6.77	4.98	1.11	2.08
8	3.92	3.13	5.08	3.72	1.36	2.98	4.072	4.56	4.14	2.63	2.01
9	4.2	4.8	7.23	3.64	2.07	3.82	4.109	5.17	4.72	1.17	0.996
10	6.41	5.56	4.01	2.7	4.61	2.36	2.81	2.03	4.54	2.91	0.87
Sample´s mean ($\bar{X}$)	5.01	4.83	6.02	4.96	2.48	2.62	4.01	5.25	4.68	2.63	1.70
Real P_{Per50} value (μ)	6.03	5.50	6.72	5.81	2.13	3.15	4.69	6.21	4.79	3.25	2.11
Sample Standard Deviation (s)	1.498	1.064	1.729	1.780	1.354	0.889	0.986	1.630	0.300	1.245	0.866
Sample Size (n)	10	10	10	10	10	10	10	10	10	10	10
t	0.400	-1.0	0.96	-0.829	-1.62	2.06	-0.6	2.10	-2.0	2.09	-1.91
P value	0.059	0.079	0.230	0.165	0.432	0.089	0.056	0.094	0.259	0.149	0.169

Table B-5 – T-test results with 95% Confidence Interval for Roundness

P_{er50} of 10 virtual samples	G1	G2	RG	L1	L2	L3	L4	L5	GS	CG	S
1	0.712	0.741	0.726	0.842	0.657	0.717	0.678	0.693	1	0.743	0.689
2	0.768	0.71	0.79	0.863	0.752	0.751	0.695	0.686	0.947	0.729	0.654
3	0.68	0.664	0.738	0.736	0.751	0.712	0.768	0.625	0.979	0.712	0.69
4	0.772	0.692	0.724	0.871	0.617	0.687	0.741	0.73	1	0.742	0.716
5	0.75	0.683	0.778	0.854	0.727	0.654	0.764	0.753	0.962	0.703	0.668
6	0.794	0.677	0.747	0.863	0.678	0.715	0.69	0.794	0.881	0.659	0.682
7	0.737	0.741	0.782	0.848	0.73	0.79	0.656	0.63	1	0.772	0.73
8	0.652	0.723	0.695	0.751	0.6967	0.66	0.624	0.711	0.983	0.813	0.77
9	0.692	0.699	0.815	0.861	0.63	0.8	0.692	0.765	0.997	0.752	0.727
10	0.725	0.725	0.76	0.786	0.753	0.73	0.742	0.698	1	0.736	0.759
Sample´s mean ($\bar{X}$)	0.73	0.71	0.76	0.83	0.70	0.72	0.71	0.71	0.97	0.74	0.71
Real P_{Per50} value (μ)	0.65	0.70	0.71	0.69	0.61	0.61	0.61	0.61	1.00	0.65	0.61
Sample Standard Deviation (s)	0.044	0.026	0.036	0.050	0.051	0.048	0.047	0.054	0.037	0.041	0.038
Sample Size (n)	10	10	10	10	10	10	10	10	10	10	10
t	5.514	0.647	3.963	8.636	5.474	7.224	6.325	5.692	-2.100	6.623	8.124
P value	0.0003	0.5334	0.0032	1.197E-05	0.0003	4.95E-05	0.0001	0.0002	0.0650	9.66E-05	1.956E-05

Table B-6 – T-test results with 95% Confidence Interval for Circularity

P_{er50} of 10 virtual samples	G1	G2	RG	L1	L2	L3	L4	L5	GS	CG	S
1	0.712	0.674	0.717	0.875	0.557	0.610	0.678	0.753	0.980	0.872	0.714
2	0.768	0.876	0.730	0.867	0.855	1.010	0.720	0.586	0.970	0.764	0.767
3	0.868	0.864	0.715	0.974	0.685	1.020	0.598	0.868	0.980	0.679	0.624
4	0.872	0.692	0.746	0.842	0.862	0.470	0.614	0.673	0.970	0.713	0.869
5	0.759	0.584	0.768	0.865	0.764	0.540	0.634	0.682	0.960	0.728	0.692
6	0.794	0.677	0.767	0.663	0.765	0.861	0.659	0.694	0.930	0.542	0.775
7	0.574	0.741	0.728	0.874	0.767	0.738	0.664	0.873	1.000	0.890	0.862
8	0.652	0.892	0.840	0.851	0.678	0.652	0.628	0.715	0.970	0.972	0.793
9	0.769	0.699	0.782	0.726	0.869	0.823	0.672	0.735	1.080	0.856	0.724
10	0.825	0.625	0.765	0.690	0.871	0.778	0.640	0.772	1.000	0.754	0.872
Sample´s mean ($\bar{X}$)	0.76	0.73	0.76	0.82	0.77	0.75	0.65	0.74	0.98	0.78	0.77
Real P_{Per50} value (μ)	0.7	0.67	0.73	0.78	0.68	0.67	0.68	0.68	1.00	0.68	0.70
Sample Standard Deviation (s)	0.093	0.109	0.038	0.098	0.104	0.186	0.035	0.088	0.039	0.124	0.083
Sample Size (n)	10	10	10	10	10	10	10	10	10	10	10
t	2.009	1.816	2.157	1.384	2.663	1.365	-2.617	1.985	-1.290	2.476	2.639
P value	0.075	0.103	0.059	0.200	0.026	0.205	0.028	0.078	0.229	0.035	0.027

B.4 Experimental and simulated cumulative frequency distribution curves

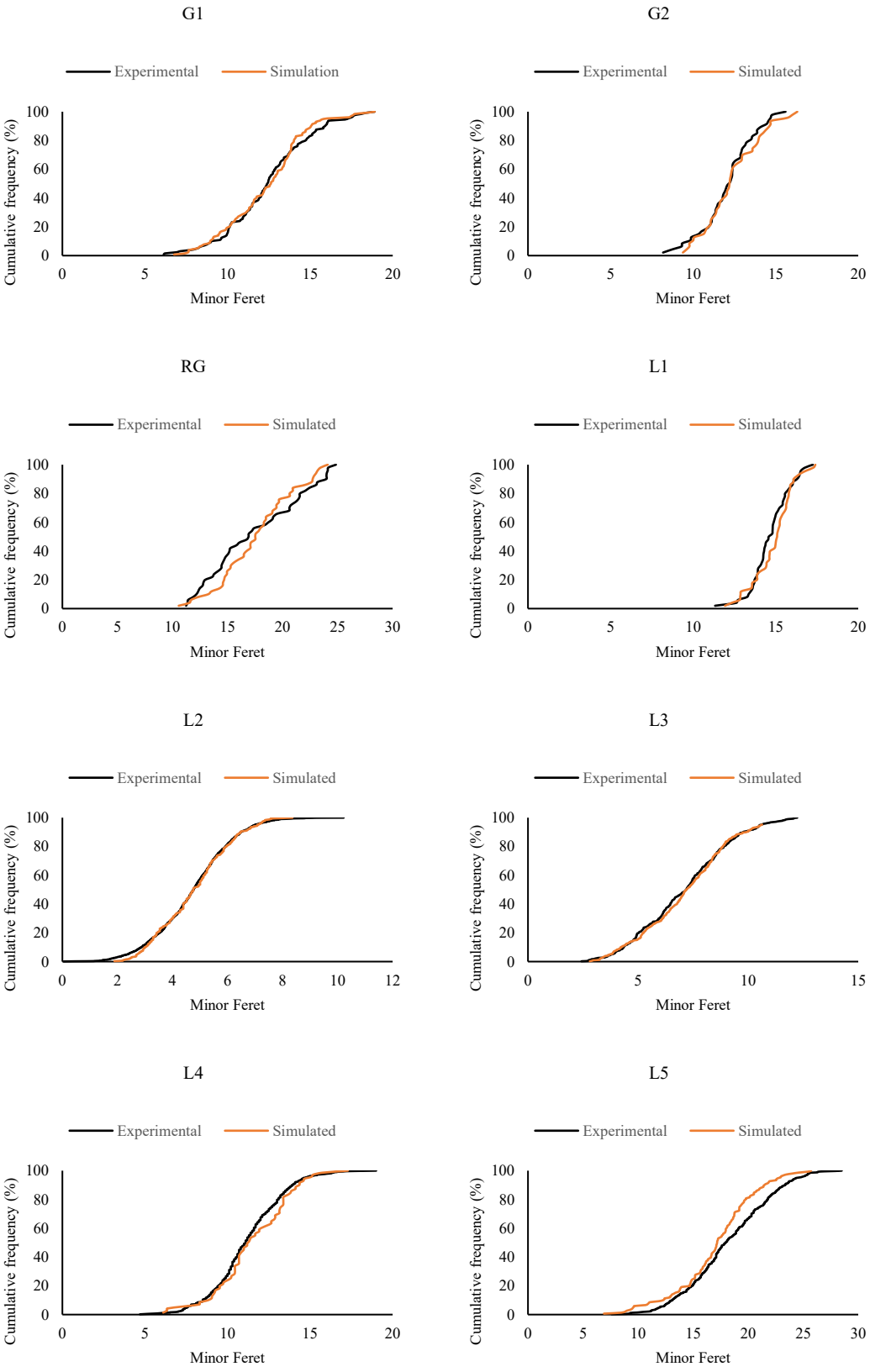
To compare the real versus virtual aggregates, their cumulative frequency distribution curves have been simply represented one versus the other.

Figure B-3-Figure B-8 show the distribution curves of the shape´s factors: minor Feret diameter, mayor Feret diameter, area, perimeter, circularity and roundness.

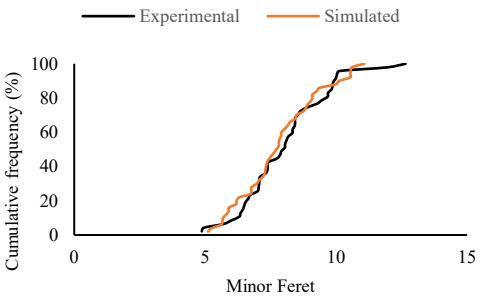
A very good fit between the real and simulated data can be seen for the major and minor Feret diameters properties, over all the samples. However, the rest of the parameters, which depend on the noise applied on a spherical primitive as explained in Section 4.3.1, give worse results.

An optimization in the virtual aggregate generation model was implemented, with the aim of adjusting the noise by means of a Differential Evolution algorithm and thus achieve a higher accuracy in the model (Chapter 6).

Figure B-3 – Minor Feret distribution curves



CG


Experimental
Simulated
Cumulative frequency (%)
100
80
60
40
20
0
0
5
10
15
Minor Feret

S

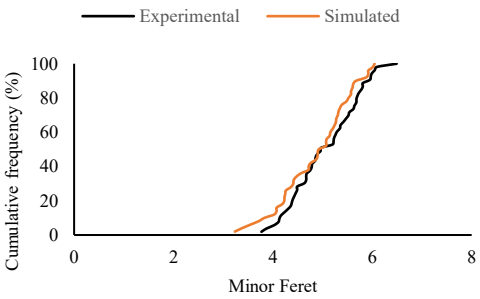
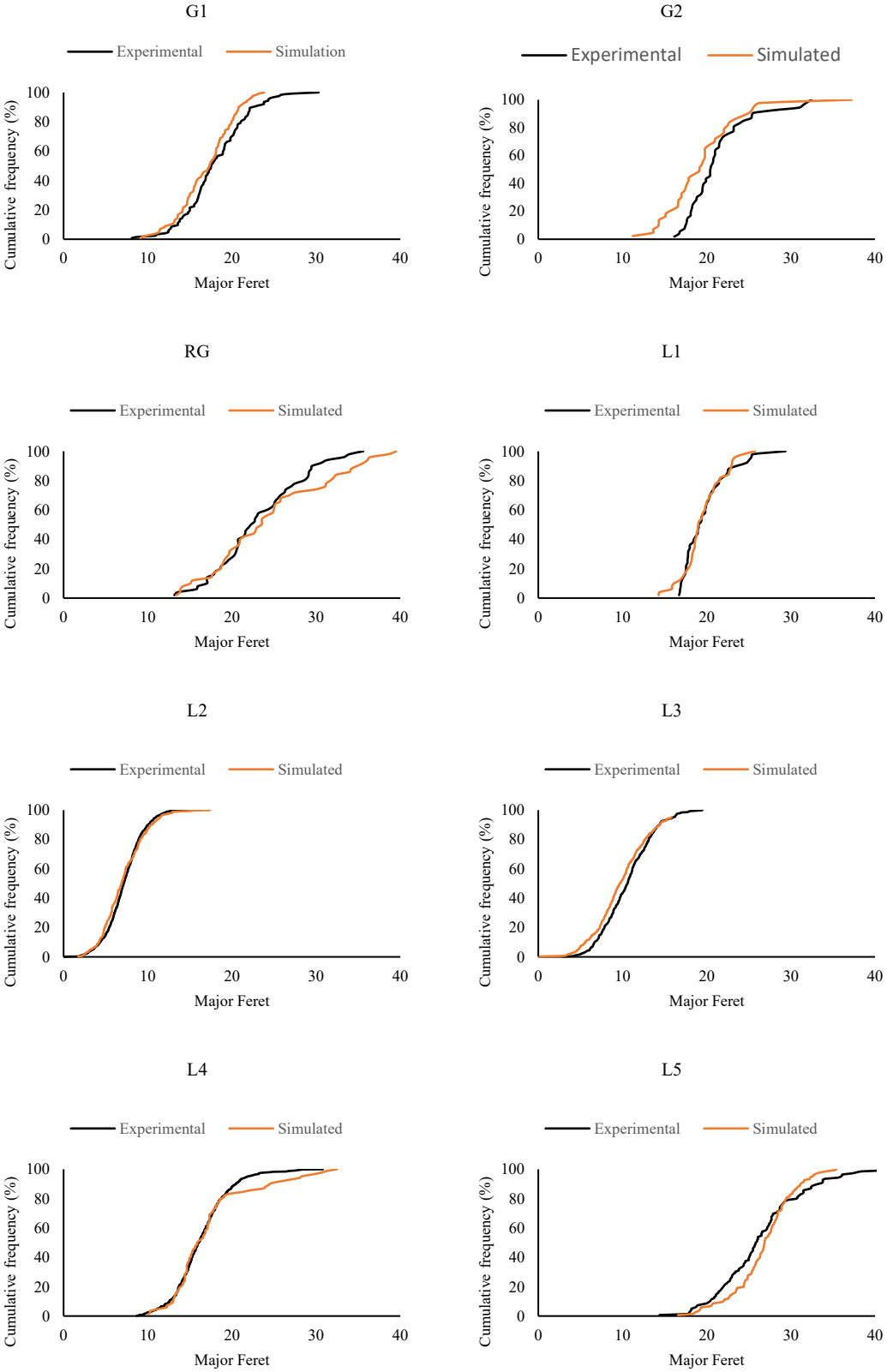
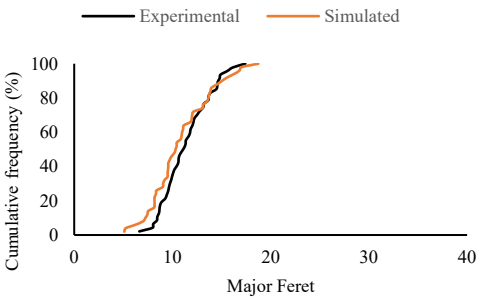

Experimental
Simulated
Cumulative frequency (%)
100
80
60
40
20
0
0
2
4
6
8
Minor Feret

Figure B-4 – Major Feret distribution curves



CG


Experimental
Simulated
Cumulative frequency (%)
100
80
60
40
20
0
0
10
20
30
40
Major Feret

S

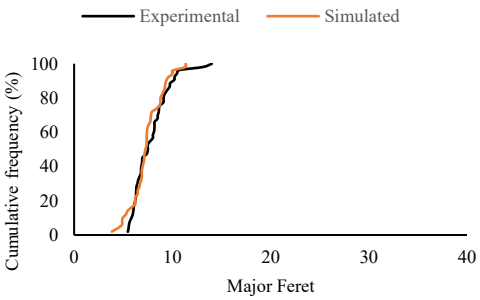
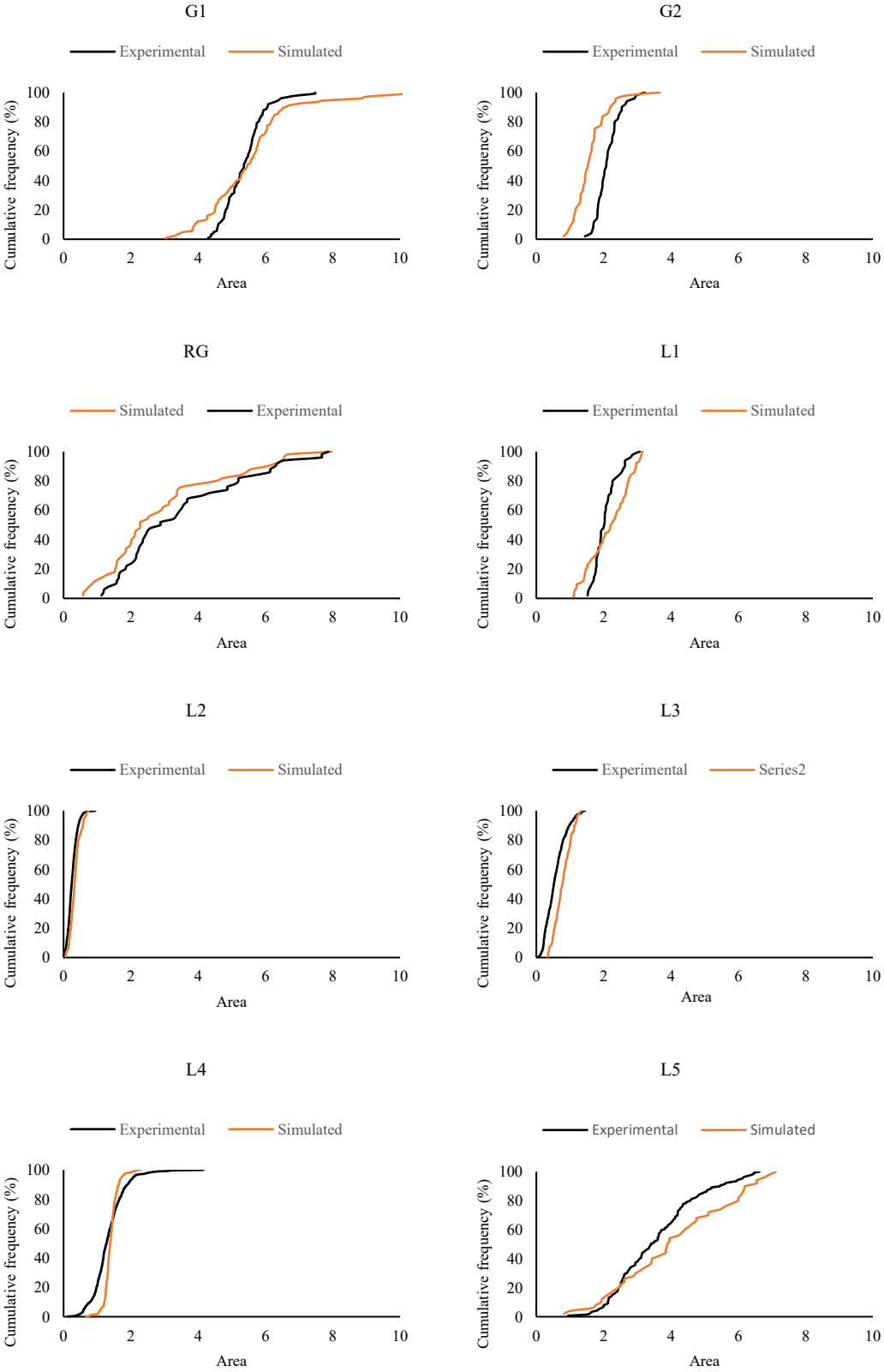
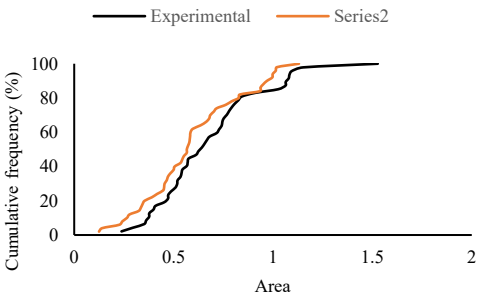

Experimental
Simulated
Cumulative frequency (%)
100
80
60
40
20
0
0
10
20
30
40
Major Feret

Figure B-5 – Area distribution curves



CG


Experimental
Series2
Cumulative frequency (%)
100
80
60
40
20
0
0
0.5
1
1.5
2
Area

S

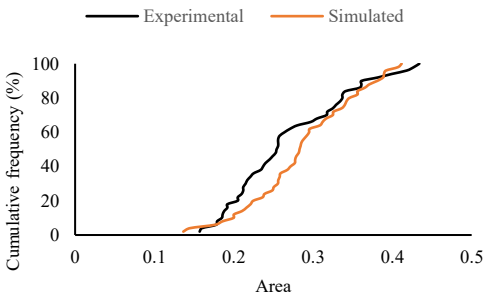
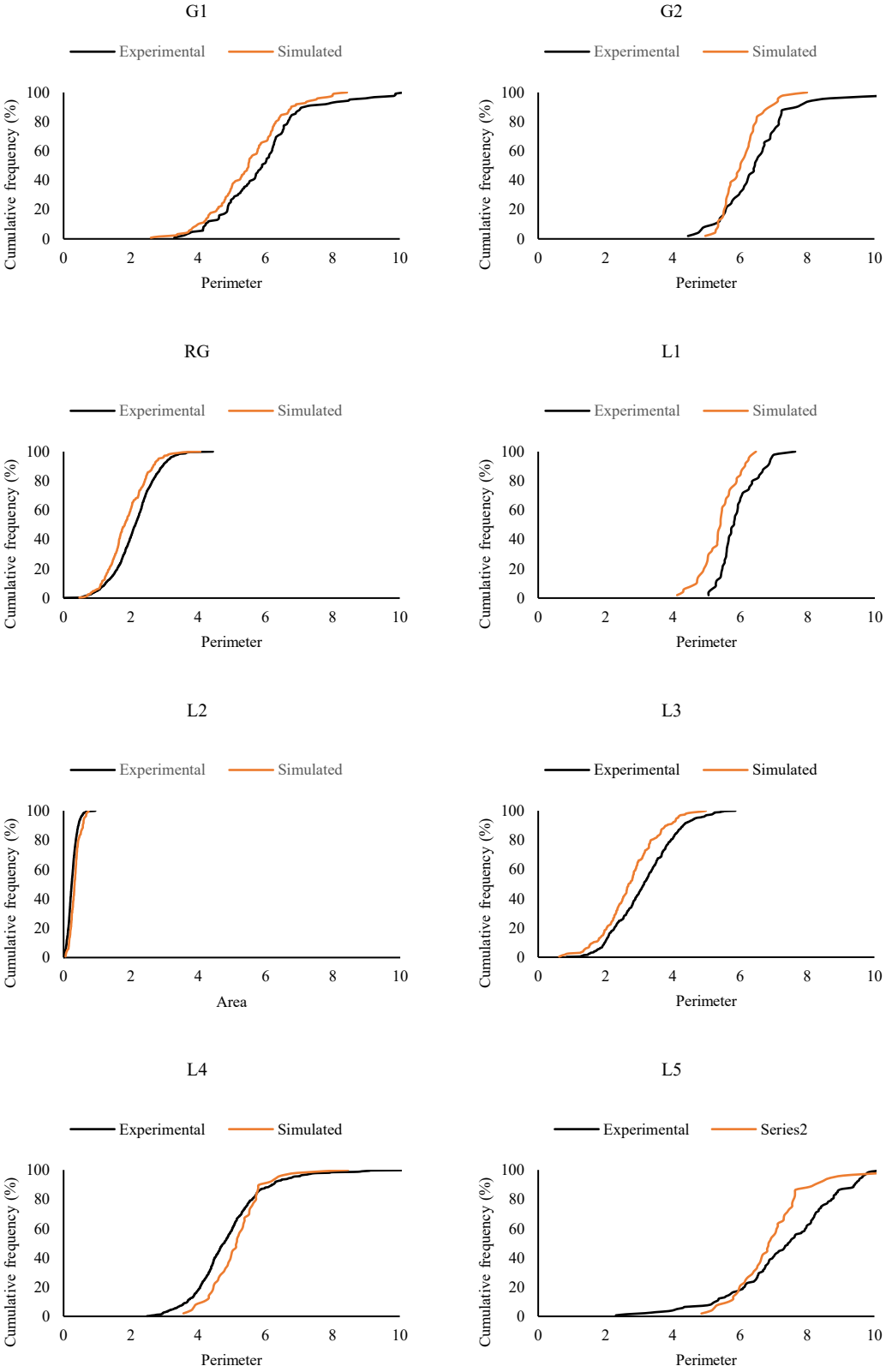
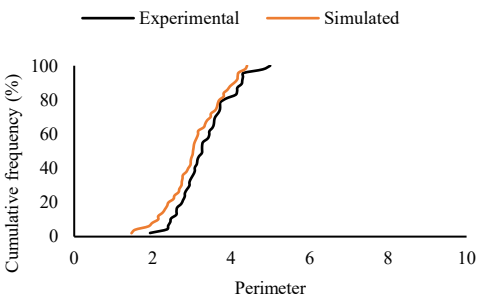

Experimental
Simulated
Cumulative frequency (%)
100
80
60
40
20
0
0
0.1
0.2
0.3
0.4
0.5
Area

Figure B-6 – Perimeter distribution curves



CG


Experimental
Simulated
Cumulative frequency (%)
100
80
60
40
20
0
0
2
4
6
8
10
Perimeter

S

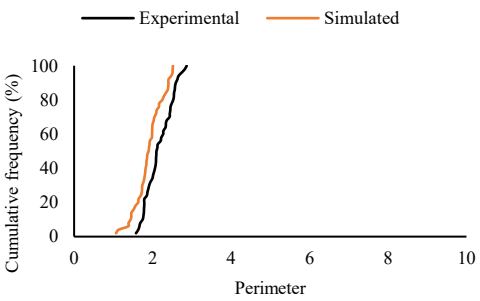
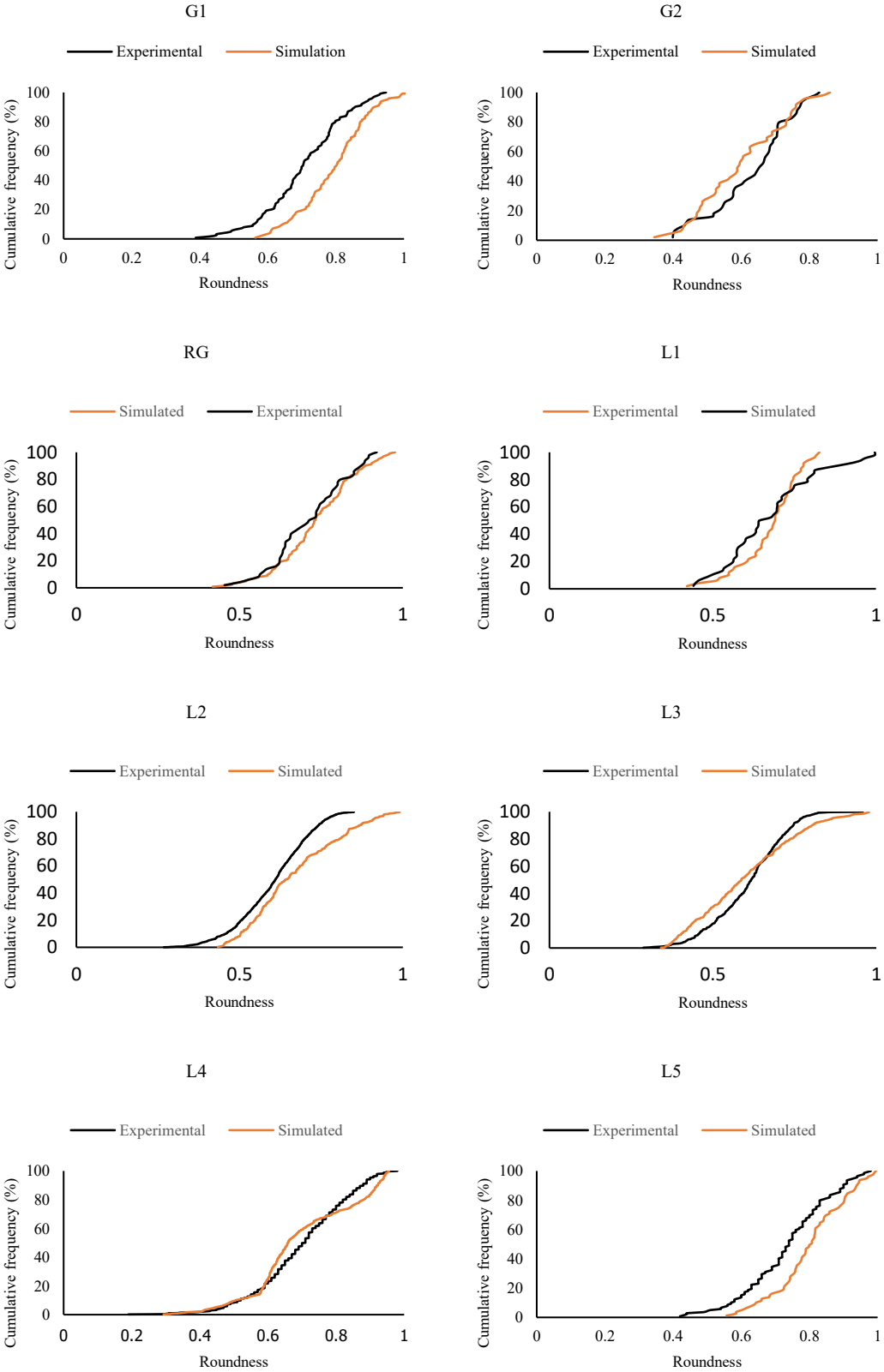
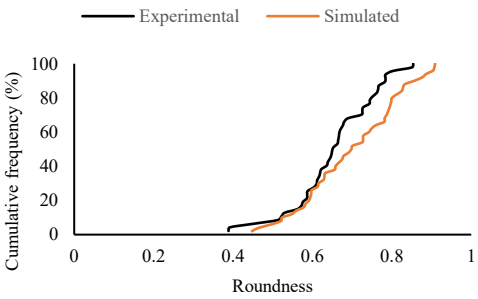

Experimental
Simulated
Cumulative frequency (%)
100
80
60
40
20
0
0
2
4
6
8
10
Perimeter

Figure B-7 – Roundness distribution curves



CG


Experimental
Simulated
Cumulative frequency (%)
100
80
60
40
20
0
0
0.2
0.4
0.6
0.8
1
Roundness

S

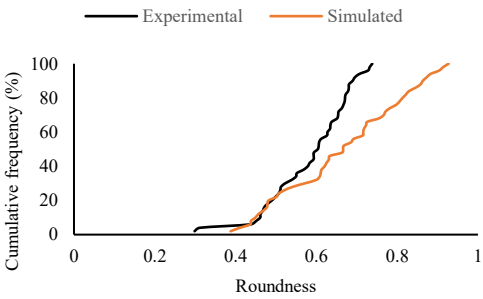
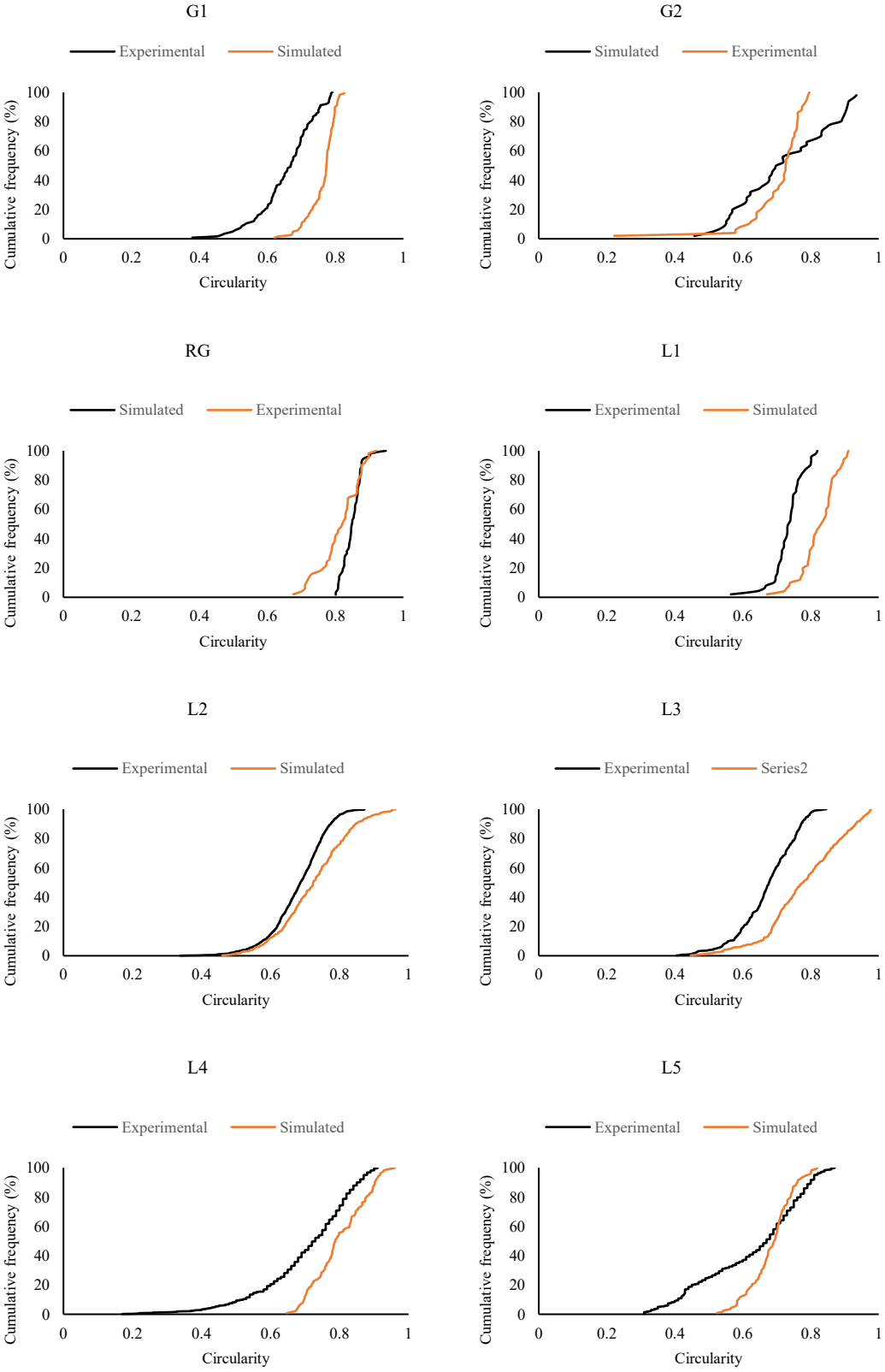
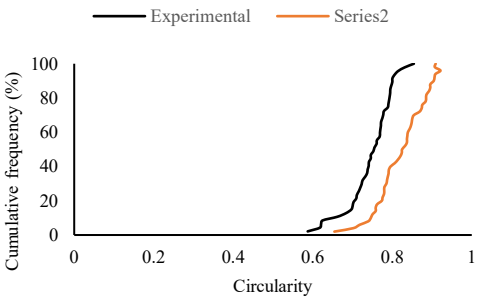

Experimental
Simulated
Cumulative frequency (%)
100
80
60
40
20
0
0
0.2
0.4
0.6
0.8
1
Roundness

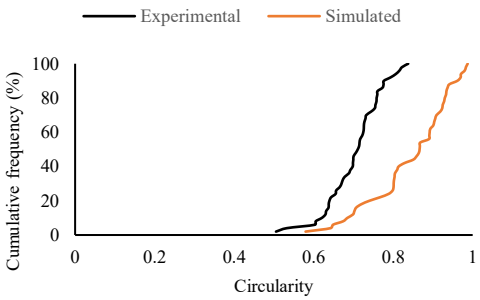
Figure B-8 – Circularity distribution curves



CG


Experimental
Series2
Cumulative frequency (%)
100
80
60
40
20
0
0
0.2
0.4
0.6
0.8
1
Circularity

S


Experimental
Simulated
Cumulative frequency (%)
100
80
60
40
20
0
0
0.2
0.4
0.6
0.8
1
Circularity

APPENDIX C. VIRTUAL PARTICLE PACKING

C.1 Results from different collision meshes

Table C-1 shows the ratio between virtual and experimental measurements, and the corresponding percentage of error of the virtual results for the different mesh collision configurations tested in the *Concave Collider* plugin.

It can be seen that the difference between the CT Scans results and results obtained in models configured with a concave mesh and a convex mesh of medium precision ranges between 10% and 25%. The high precision mesh gives results closer to the real ones; however, the differences in results between a medium precision concave collision mesh and a high precision one range between 3% and 5%, therefore the medium precision concave mesh has been considered acceptable.

Table C-1 – Comparison of results for the different collision meshes for sample G1

			Mesh collider		
	G1 structure properties		**Concave**	**medium precision**	**High precision**
Air Void Properties		%	40.2	36.8	35.87
	Porosity (p)	R_{index}	1.17	1.07	1.05
		%error	17.20	7.29	4.58
		cm	0.386	0.35	0.336
	Average pore size ($\bar{\Phi}_v$)Φ_v)	R_{index}	1.21	1.09	1.05
		%error	20.63	9.38	5.00
		cm	0.978	0.77	0.7432
	Maximum pore size ($\Phi_{v,max}$)	R_{index}	1.36	1.07	1.03
		%error	35.83	6.94	3.22
		-	1.06	1.11	1.115
	Tortuosity (τ)	R_{index}	0.84	0.88	0.88
		%error	15.87	11.90	11.51
		-	3.741	3.57	3.421
	Connectivity (z)	R_{index}	1.16	1.11	1.06
		%error	15.82	10.53	5.91
		-	0.7239	0.61	0.589
	Aspect Ratio (AR)	R_{index}	1.54	1.30	1.25
		%error	54.02	29.79	25.32
		-	0.8759	0.82	0.789
	Circularity	R_{index}	1.46	1.37	1.32
		%error	45.98	36.67	31.50

C.2 Video of a particle packing process

A video comparing the particle packing process with the virtual approach can be seen at:

https://youtu.be/7qTtQ0ZH7-g

C.3 Virtual compaction parameter´s configuration

It is very difficult to achieve a 100% realistic simulation due to all the factors involved in the simulation process and its limitations, which are still not well-known. Therefore, the virtual frequency, the container's amplitude of the displacement, and the vibrating times were found by trial and error tests, so as to avoid 1) particle´s interpenetration and particles that pass through the container (due to high object's velocities, impossible to compute by the physics engine) and 2) the instability of the system, since at higher particle´s velocities, larger amount of contacts between object are formed in each time step.

A displacement amplitude was applied with a determined frequency in the center of the container in each of the three axes. For all the samples, the frequencies and the vibrating times were kept the same, summarized in the following table:

Table C-2 – Virtual compaction parameter´s configuration

Compaction Algorithm´s Parameters	Values
X axis amplitude (mm)	4
X axis frequency (Hz)	50
Y axis amplitude (mm)	4
Y axis frequency (Hz)	50
Z axis amplitude (mm)	0.6
Z axis frequency (Hz)	80
Vibration´s time previous compaction (s)	120
Vibration´s time during compaction (s)	120
Vibration time after adjusting the height of the weight	240

The duration of vibration during previous the compaction remained the same as the experimental one.

The height of the simulated samples was controlled by monitoring the height of the base of the compressor weight. It was found that, with the configuration presented in the above table, after two minutes of vibration during the compaction process, the particles system starts becoming stable, being the height of the assemblage's constant, with slight variations due to particles movement. However, the heights of the virtual samples were between 2 and 5cm higher than the real ones.

Therefore, it was decided to adjust the height of the weight to the actual height of the samples and continue vibrating the particles for four minutes more for the particle's rearrangement, ensuring like this not overlapping between particles.

C.4 Slicing process of the particle´s assembly with Blender

To be able to analyze the virtual stone´s assemblies generated, with the image processing software ImageJ, so as done with the real one, the free license software Blender was employed under the purpose to create "virtual tomographies" of the 3D particles packed, imitating thus a CT-scan process.

The procedure consist mainly on creating a stack of 2D images by slicing the virtual grain´s packing, using a plane perpendicular to its vertical direction (Figure C-1), which plot the 2D intersection between the stones and the layer (Figure C-2). The distance between layers can be set by the user and, in this case they were chose according to the vertical distance between slices used during the CT-scan process.

Figure C-1 – Layer perpendicular to a virtual particle´s assembly

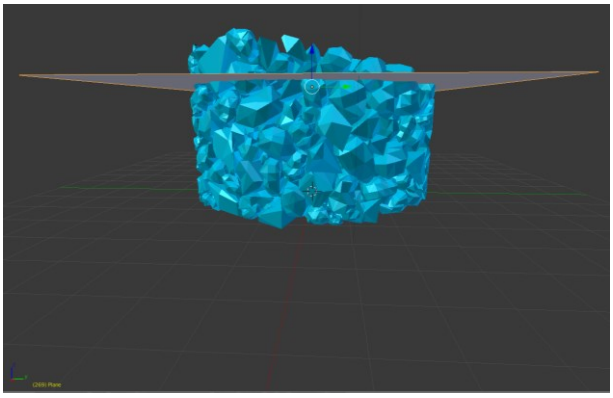
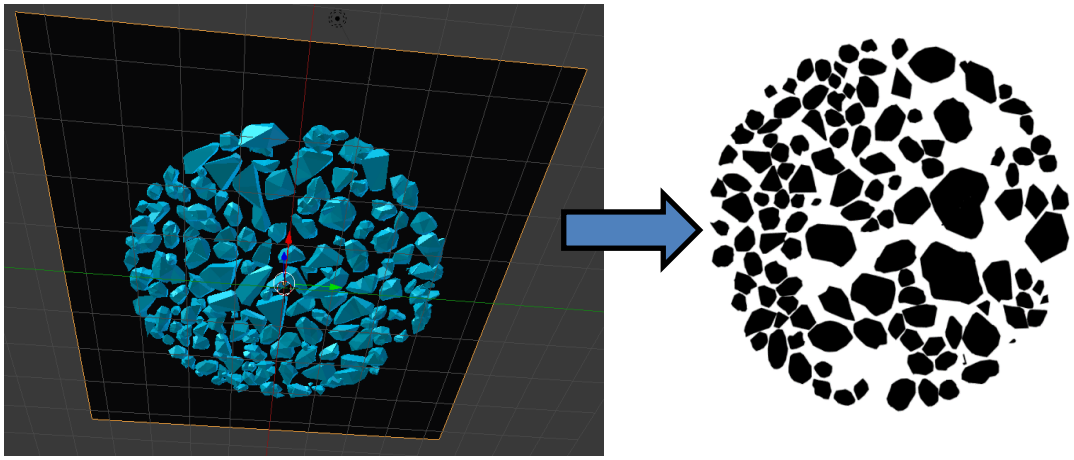


Figure C-2 – 2D intersection between the stones and the layer

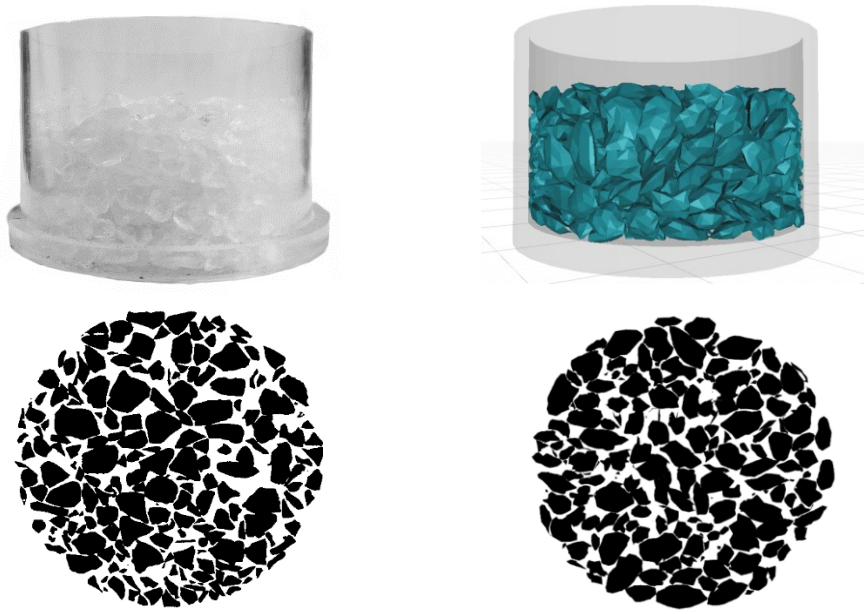


Some examples about virtual and real slices extracted from the aggregat´s assemblies are shown in the next picture:

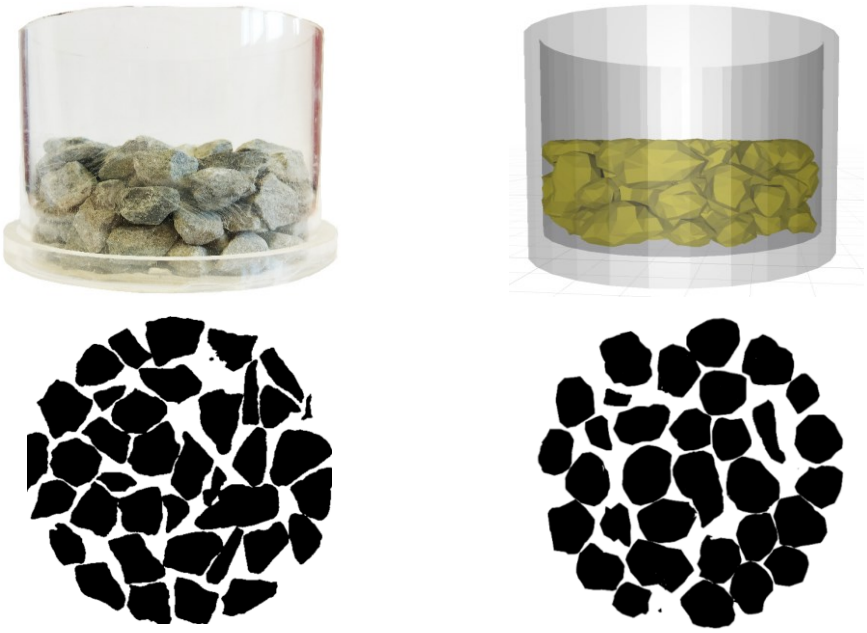
Figure C-3 – Real and virtual particle´s assemblies and they corresponding 2D slices for samples: a) Crushed Glass (CG) and b) Limestone 1 (L1)

Experimental **Simulation**

a) Cristal Glass samples (CG) with 7mm size



b) Limestone 1 (L1) with 12.5mm size

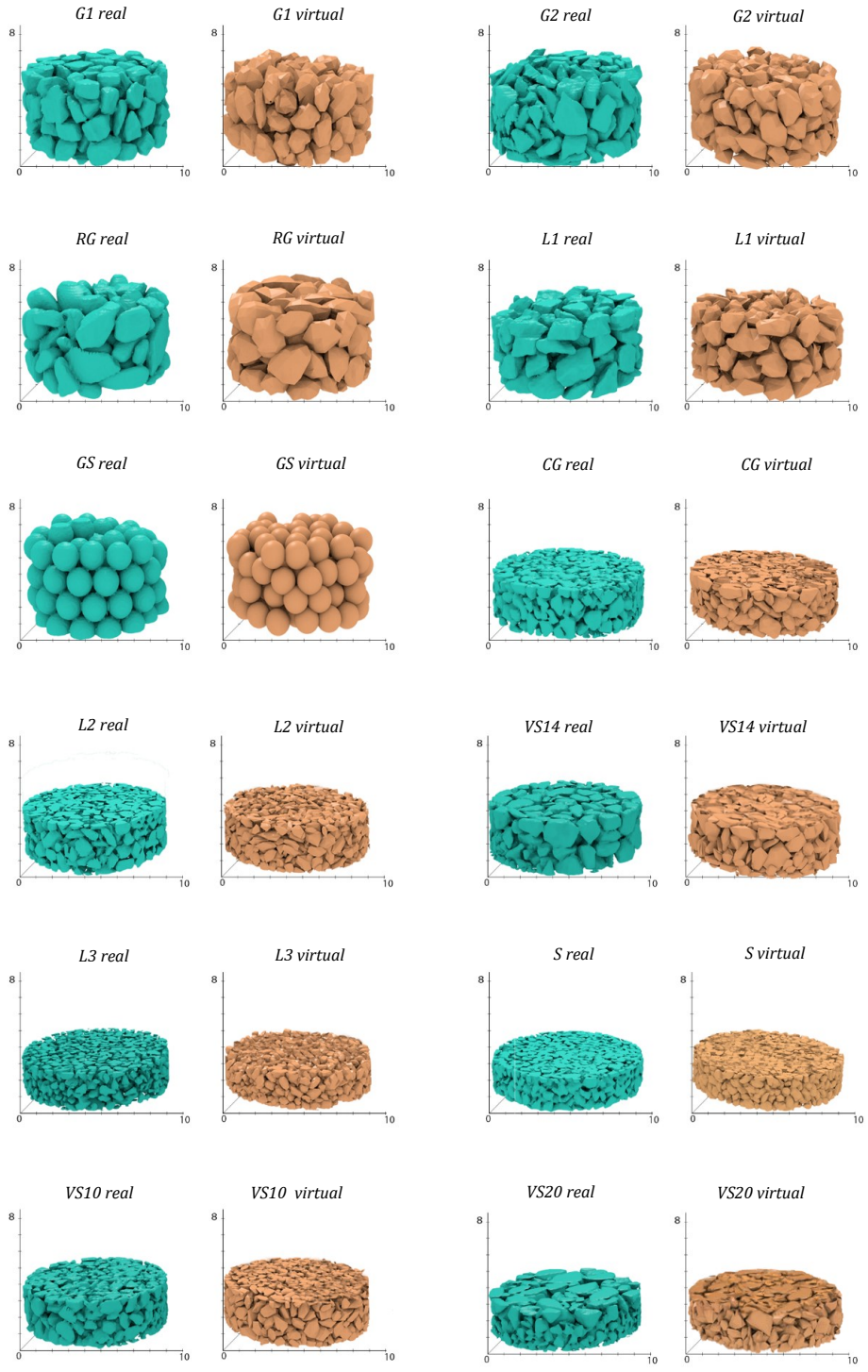


C.5 Experimental and simulated Air void distribution curves

C.5.1 Visual comparison between real and simulated packed aggregates

The next picture shows a comparison between the real specimens and their corresponding simulation.

Figure C-4 – Visual comparison between real and simulated packed aggregates



Internal structure characteristics of the simulated aggregate´s packings

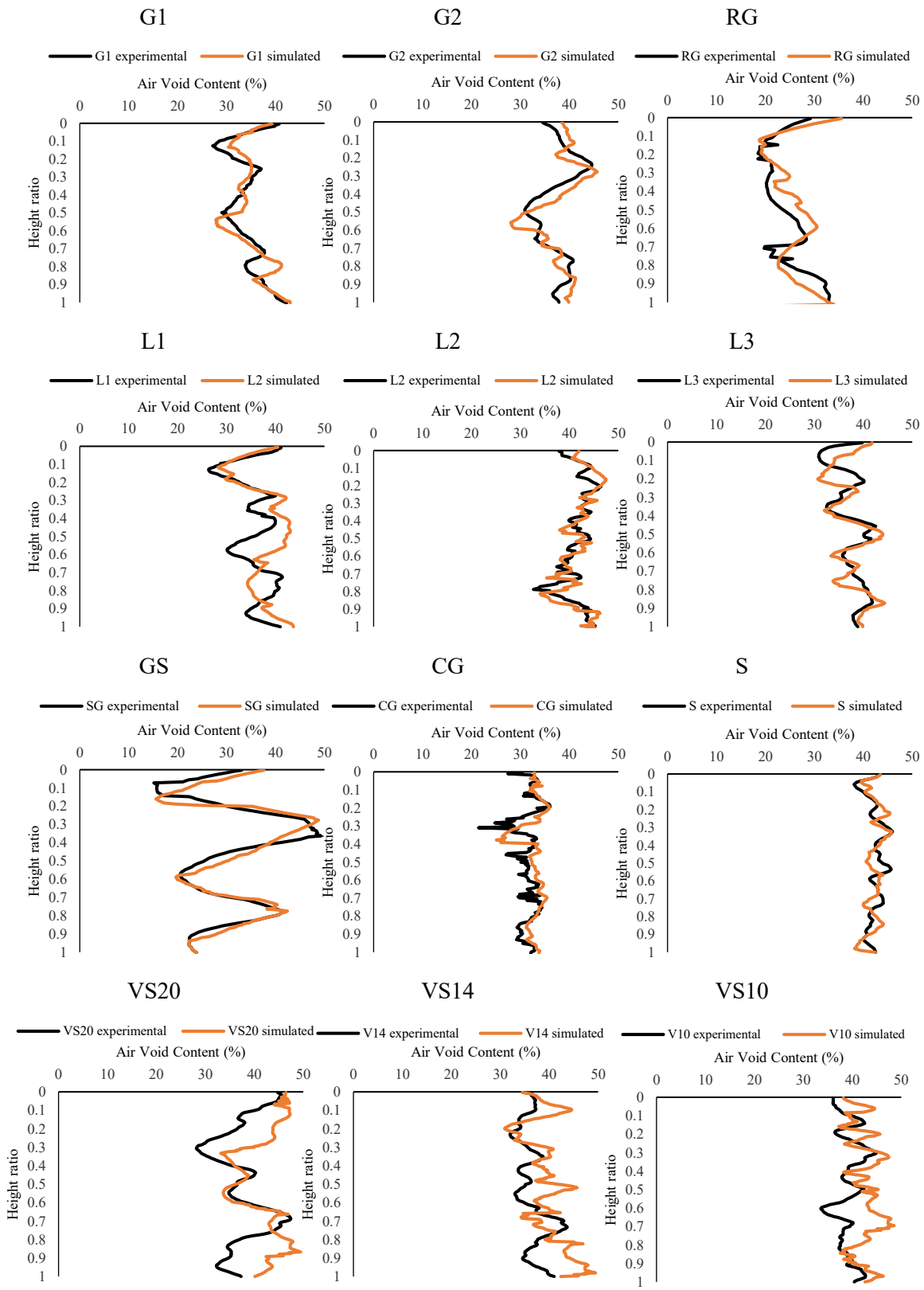
Table C-3 – Void and particle characteristics derived from the ImageJ analysis of the virtual particle´s packed.

			G1	G2	RG	L1	L2	L3	GS	CG	S	VS20	VS14	VS10
Air Void Properties	Porosity (p)	%	36.8	37.4	32	33.2	44.2	40.2	37	34.3	45.8	37.1	39.3	40.9
	Average pore size ($\bar{\Phi}_v$)Φ_v)	cm	0.35	0.44	0.46	0.43	0.19	0.28	0.62	0.2	0.198	0.32	0.33	0.24
	Maximum pore size ($\Phi_{v,max}$)	cm	0.77	0.99	1.07	1.02	0.489	0.87	1.02	0.46	0.485	0.75	0.113	0.54
	Tortuosity (τ)	-	1.11	1.21	1.18	1.26	1.09	1.16	1.05	1.43	1.12	1.05	1.07	1.04
	Connectivity (z)	-	3.57	3.41	3.43	3.61	3.79	3.6	3.39	3.78	3.82	3.81	3.82	3.78
	Pore density (n_v)	-	0.0396	0.0067	0.0029	0.0087	0.0197	0.022	0.0078	0.115	0.027	0.0278	0.0281	0.0291
	Aspect Ratio (AR)	-	0.61	0.54	0.41	0.49	0.59	0.51	0.52	0.56	0.6	0.59	0.55	0.57
	Circularity	-	0.82	0.58	0.67	0.59	0.79	0.71	0.45	0.76	0.78	0.78	0.72	0.73
Aggregate Properties	Average aggregate size ($\bar{\Phi}_a$)	cm	0.6	0.83	1.21	0.72	0.27	0.44	1.51	0.48	0.25	7.18	5.87	4.42
	Maximum aggregate size ($\Phi_{a,\ max}$)	cm	1.27	1.26	1.84	1.5	0.57	1.12	1.67	0.72	0.64	19.07	12.44	7.56
	Node density (n_d)	-	0.052	0.011	0.0078	0.012	0.077	0.048	0.00175	0.18	0.078	0.022	0.0116	0.026
	Roundness	-	0.75	0.8	0.77	0.88	0.86	0.78	0.98	0.84	0.8	-	-	-
	Circularity	-	0.78	0.73	0.75	0.83	0.74	0.74	0.98	0.8	0.79	-	-	-
	Aspect Ratio	-	0.72	0.75	0.81	0.84	0.71	0.71	0.99	0.68	0.74	-	-	-

C.5.2 Experimental and simulated Air Void distribution curves

A comparison between the experimental and simulated air void distribution curves in vertical direction have been done, for the particles packed described in Chapter 5.

Figure C-5 – Air void curve of the different real and virtual mixtures



APPENDIX D. ASPHALT SKELETON COMPRESSION SIMULATION

D.1 Video of the virtual compaction with rotational movement

The discarded virtual approach for the compaction of the coarse aggregates, with include the angle of gyro to imitate the Superpave Gyratory Compactor motion can be seen at:

https://youtu.be/h5KLy8FpKWc

D.2 DE optimisation algorithm

Detailed information about how the Differential Evolution technique works is provided in this section.

First, the user needs to define de size, (*NP*) of the initial population, G, of individuals I_i^G (Eq. D-1), where the population member I_1^G consist on numerical vectors

$$Initial\ population, G\ = [I_1^G\ , I_2^G , I_3^G\ \ldots\ldots I_{NP}^G\ ],\ I_i^G\ \in R^n \qquad \text{D-1}$$

Where,

NP is the population size, the number of individuals in the population.

n is the dimension of the optimization (the number of variables to be optimized, which in this case are three).

Then, the evolution starts by applying a mutation strategy in each individual of the first population generated, named target individuals, or target vector (I_i^G), to create mutant vectors (I_{Mut}^G), as shown in Equation D-2.

Each mutation consists on selecting randomly three members of the population, (I_1^G, I_2^G, I_3^G), where I_3^G is the base vector. To I_3^G is added the weighted difference between the two remains individuals (I_1^G, I_2^G). The weighted difference is done thought a mutation constant, F, named scaling factors.

The scaling factor is a real number defined by the user, which control the amplification of the difference vector $[I_1^G - I_2^G]$, and its optimal value ranges between 0 and 1 according to [230].

$$I_{Mut}^G = F * [I_1^G - I_2^G] + I_3^G, \qquad I_1^G \neq I_2^G \neq I_3^G \tag{D-2}$$

Once this mutation process is finished, a child individual, I_{Child}^G, is created by crossing the mutant vector I_{Mut}^G with other random vector selected from the population, I_{Target}^G, based on a crossover rate, Cr, and being $I_{Target}^G \neq I_1^G \neq I_2^G \neq I_3^G$. This process is known as Crossover (Equation D-3). Each I_{Child} from the population is a trial solution.

$$I_{Child}^G = \begin{cases} I_{Mut}^G \; if \; rand\,(0,1) \leq Cr \\ I_{Target}^G \qquad otherwise \end{cases} \tag{D-3}$$

Where,

Cr is the chromosome rate that controls how many components are inherited by I_{Child} from the mutant vector. As the scaling factor, F, the Cr is a user-defined number chosen from an interval [0,1] [230]; and

$rand\,(0,1)$ is a uniformly distributed random number between [0,1].

Finally, the trial vector (I_{Child}^{G}) and the current individual, I_{Target}^{G}, are compared through a Fitness Function $f(I_i)$, which is in charge to evaluate the solution created in each mutation and crossover, to know which individuals produce the fitter results, in order to choose the best for next generation of individuals (Equation D-4).

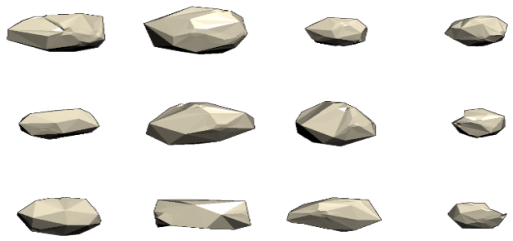
$$I_i^{G+1} = \begin{cases} I_{Child}^{G} \; if \; f(I_{Child}^{G}) \leq f(I_{Target}^{G}) \\ I_{Target}^{G} \qquad\qquad otherwise \end{cases} \qquad \text{D-4}$$

D.3 Real and virtual stone´s photos used for the mixtures manufacture

Figure D-1 – Comparison between the real and virtual limestone samples

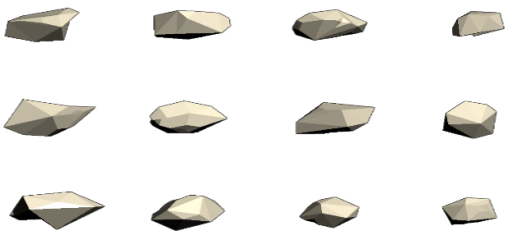
a) Limestone 20mm (L5)





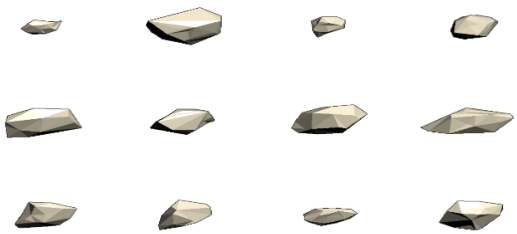
b) Limestone 14mm (L4)





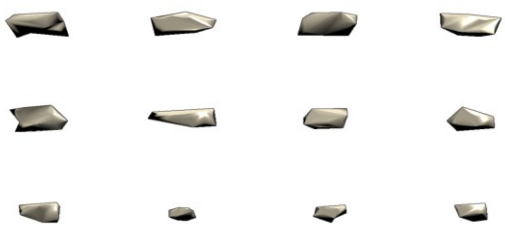
c) Limestone 10mm (L3)





d) Limestone 10mm (L2)

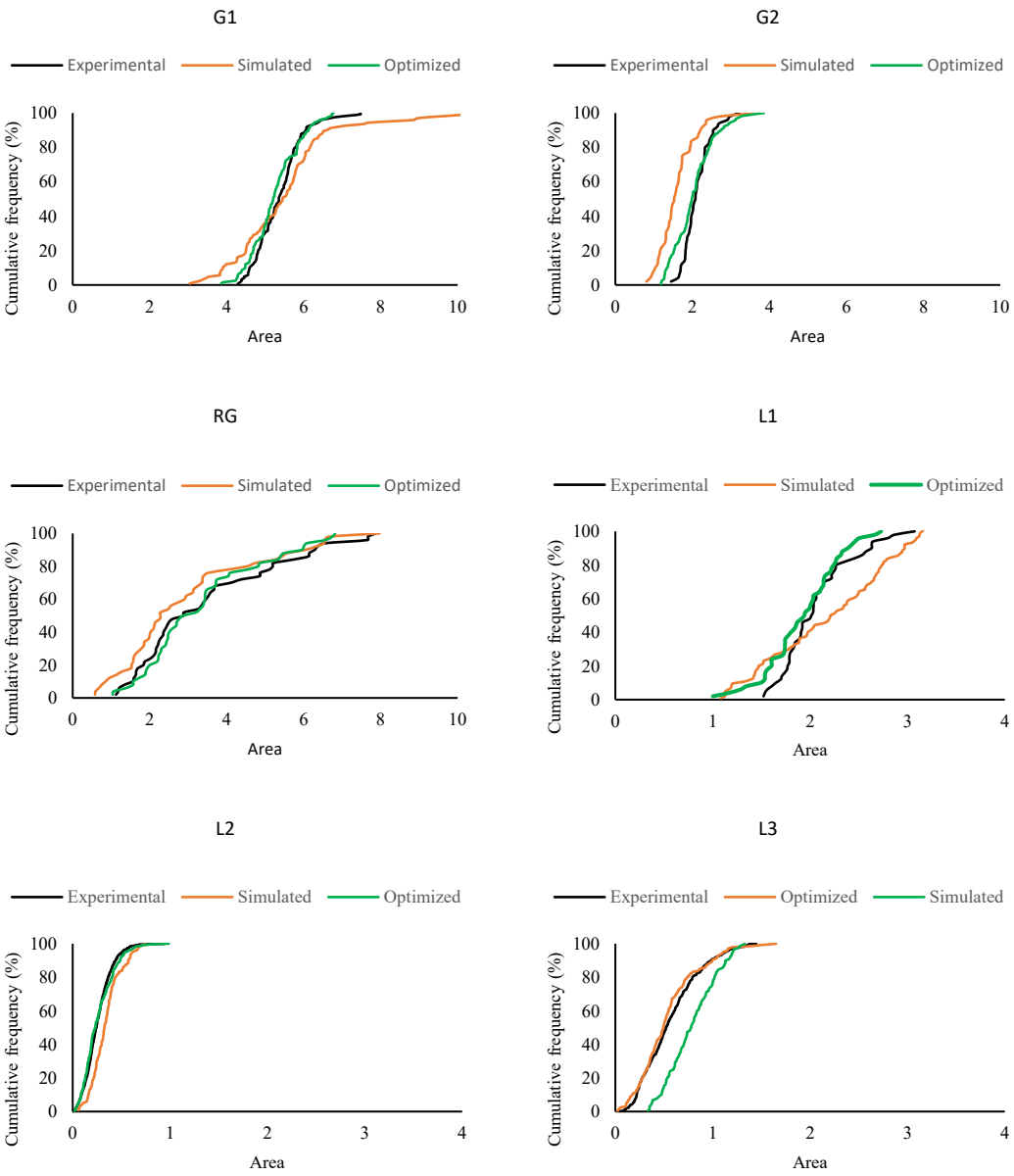




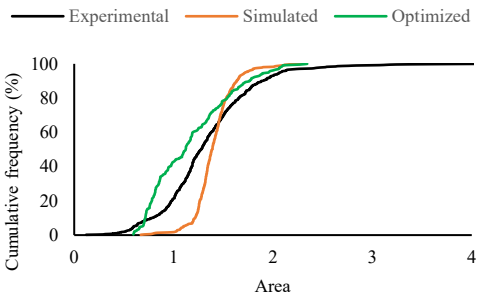
D.4 Validation of the DE optimization implemented on the aggregate generation model

D.4.1 Cumulative frequency distribution curves after the DE optimization

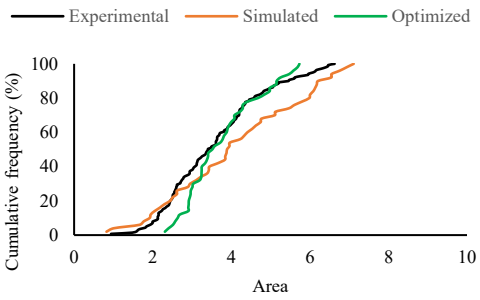
Figure D-2 –Area curves distribution



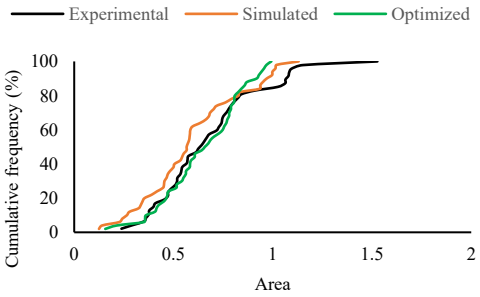
L4


Experimental
Simulated
Optimized
Cumulative frequency (%)
100
80
60
40
20
0
0
1
2
3
4
Area

L5


Experimental
Simulated
Optimized
Cumulative frequency (%)
100
80
60
40
20
0
0
2
4
6
8
10
Area

CG


Experimental
Simulated
Optimized
Cumulative frequency (%)
100
80
60
40
20
0
0
0.5
1
1.5
2
Area

S

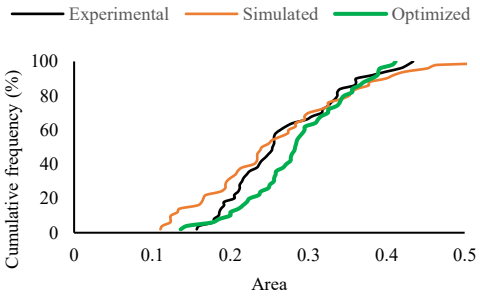
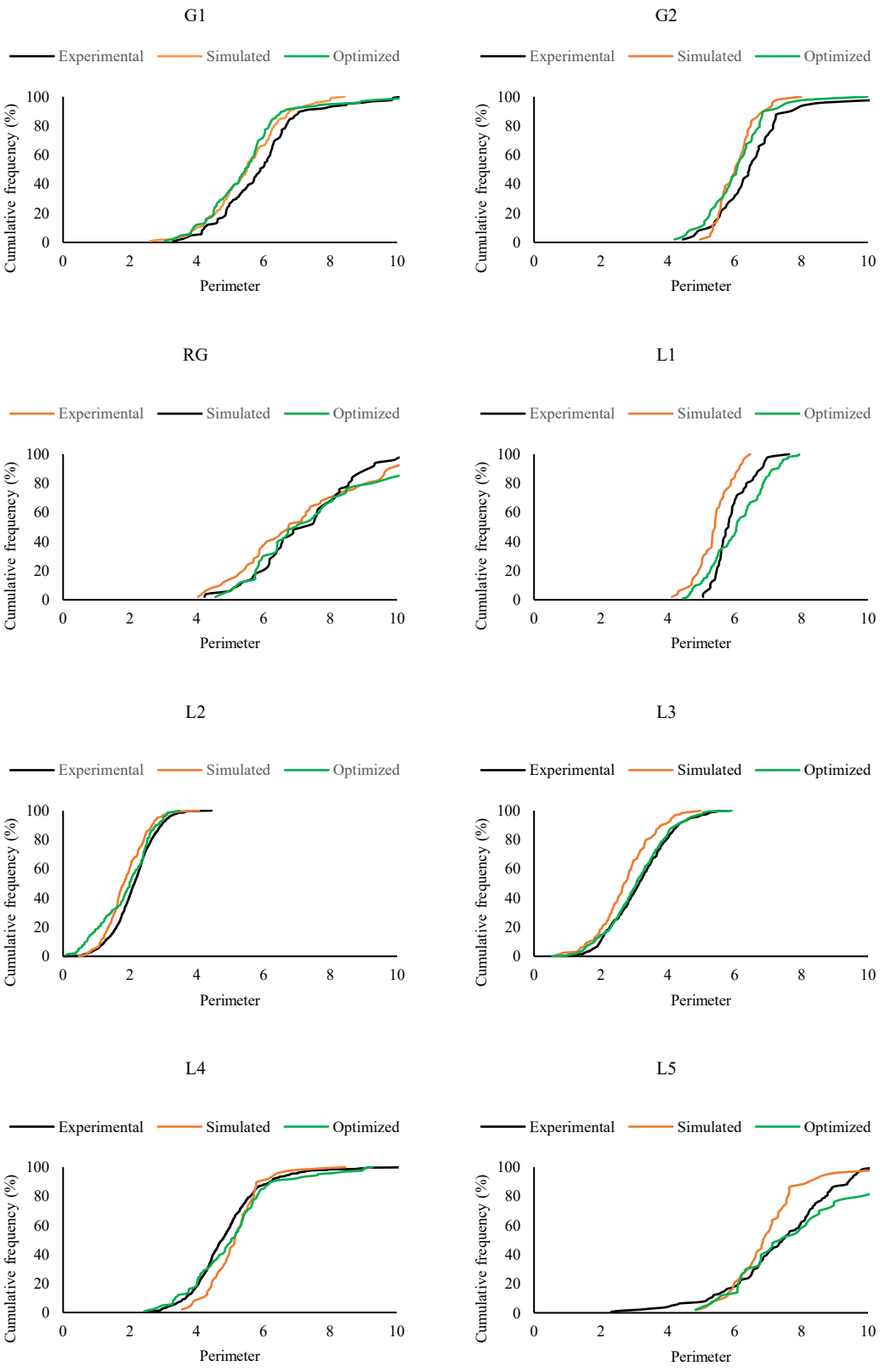

Experimental
Simulated
Optimized
Cumulative frequency (%)
100
80
60
40
20
0
0
0.1
0.2
0.3
0.4
0.5
Area

Figure D-3 –Perimeter curves distribution



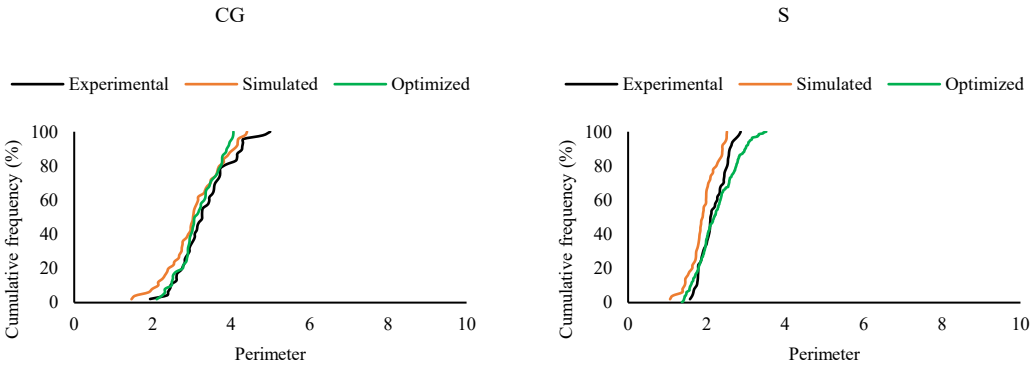
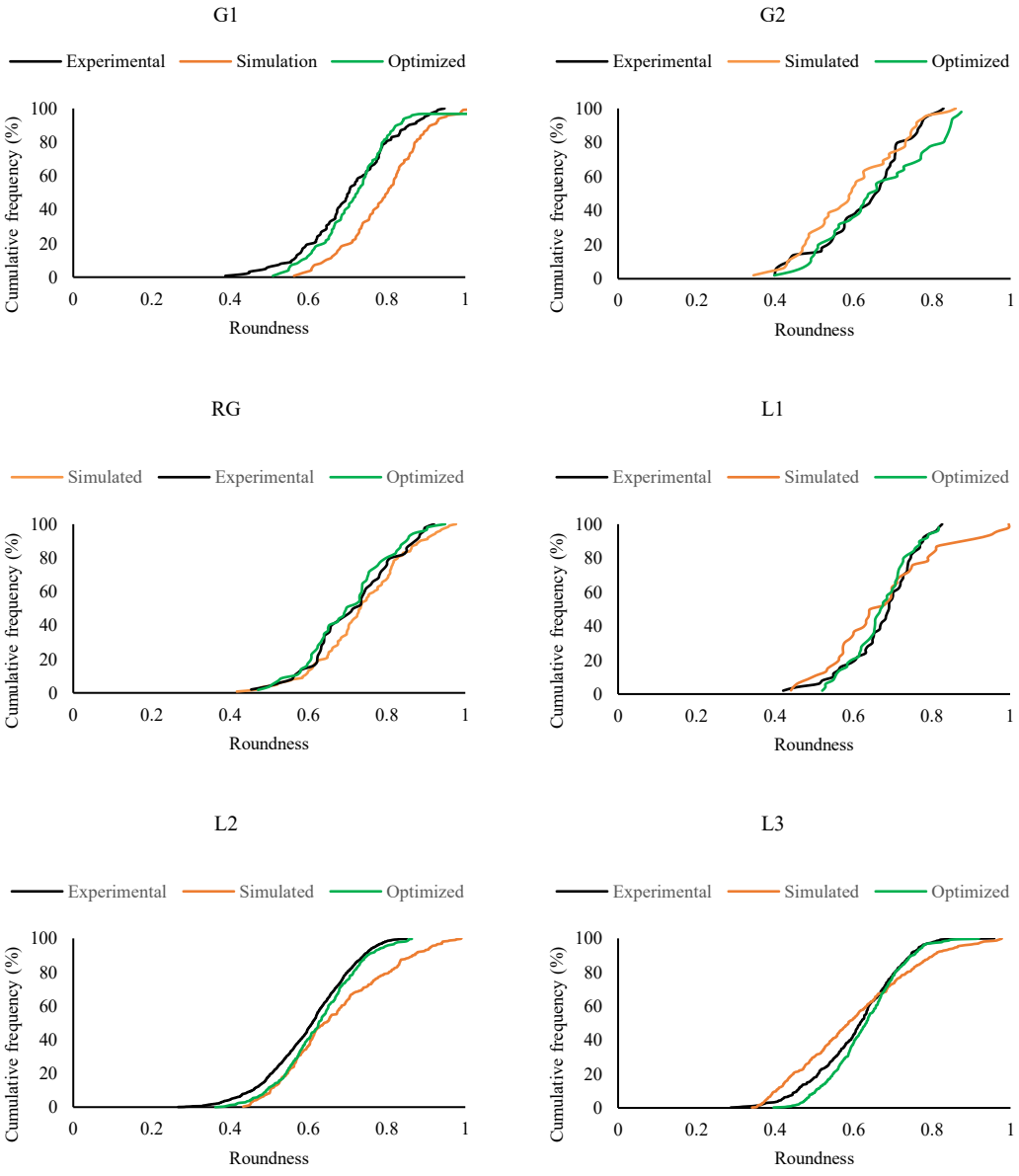
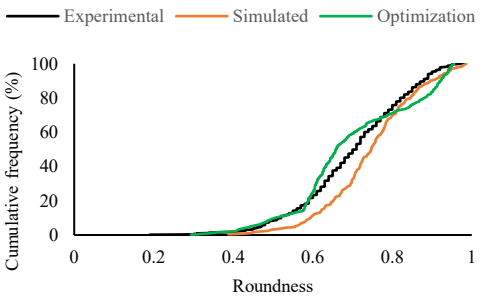


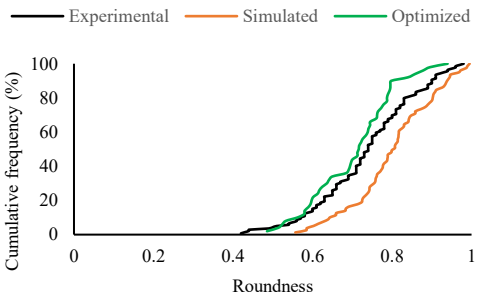
Figure D-4 –Roundness curves distribution



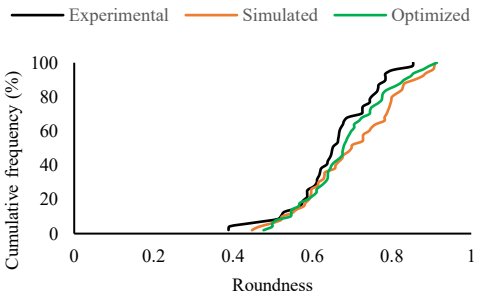
L4


Experimental
Simulated
Optimization
Cumulative frequency (%)
100
80
60
40
20
0
0
0.2
0.4
0.6
0.8
1
Roundness

L5


Experimental
Simulated
Optimized
Cumulative frequency (%)
100
80
60
40
20
0
0
0.2
0.4
0.6
0.8
1
Roundness

CG


Experimental
Simulated
Optimized
Cumulative frequency (%)
100
80
60
40
20
0
0
0.2
0.4
0.6
0.8
1
Roundness

S

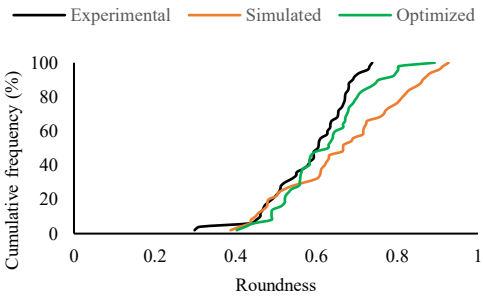
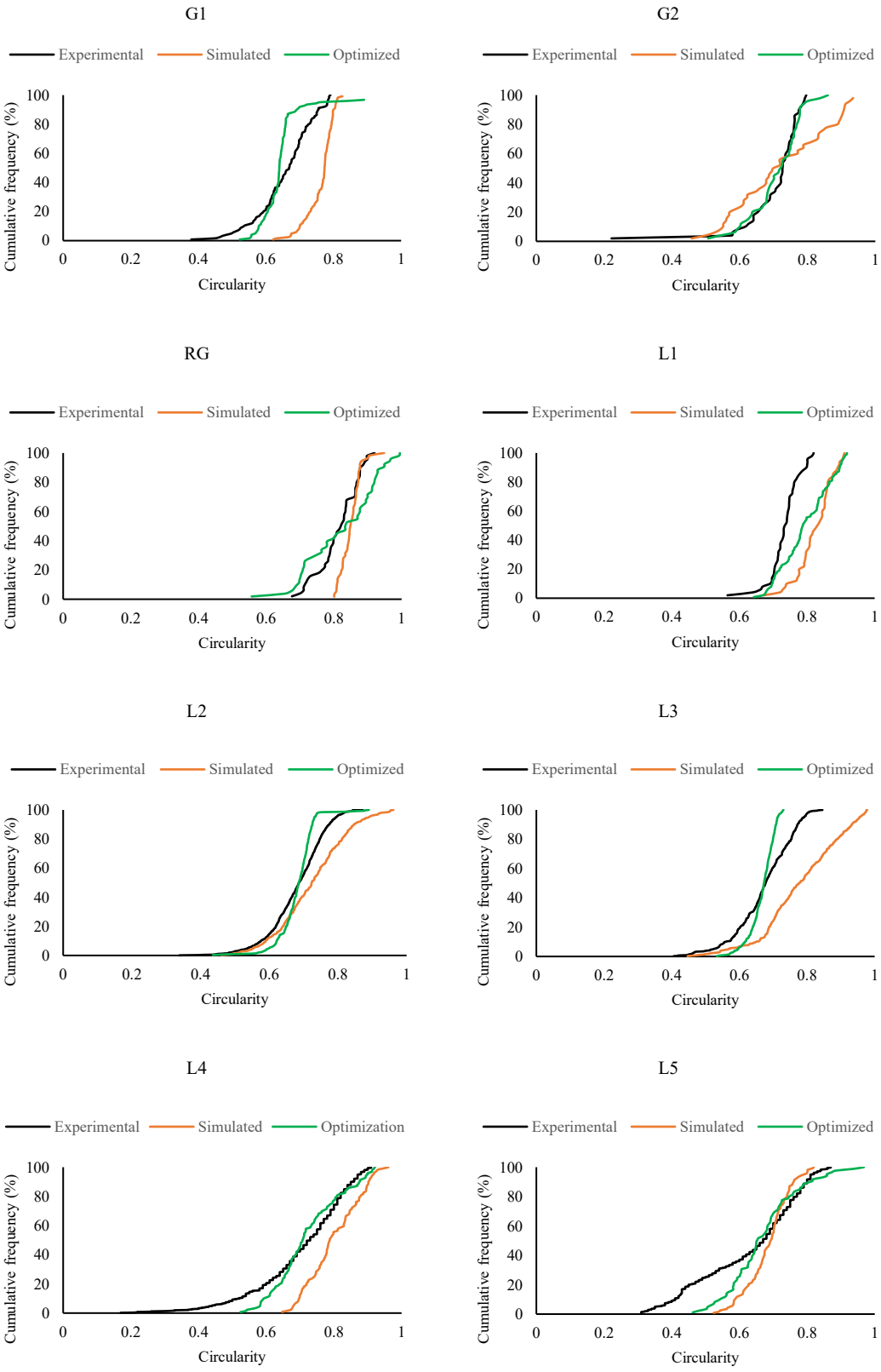
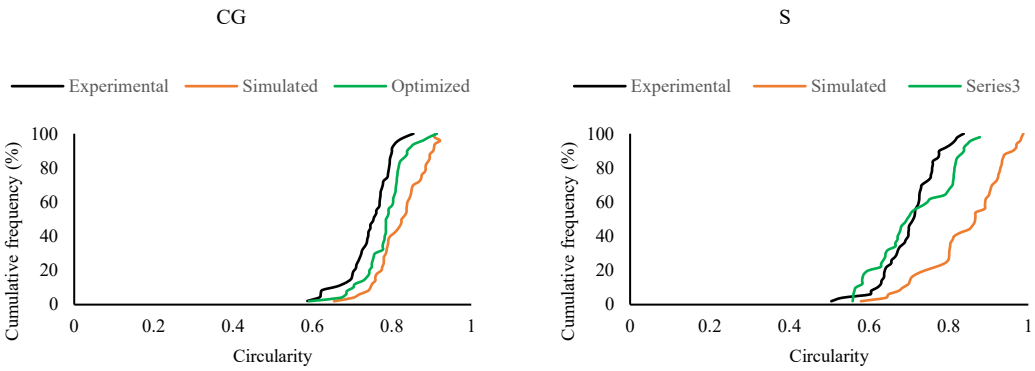

Experimental
Simulated
Optimized
Cumulative frequency (%)
100
80
60
40
20
0
0
0.2
0.4
0.6
0.8
1
Roundness

Figure D-5 – Circularity curves distribution





D.4.2 One sample T-test for the virtual aggregate generation model validation

As it was done for the validation of the aggregate generation model presented in Chapter 4, a one sample t-test was conducted in each one of the shape factors optimized in Chapter 6, to verify the effectiveness of the Differential Evolution employed for that purpose.

The results drawn on Tables D-1/ D-4 reveal that, with the implementation of the DE algorithm, the proposed aggregate generation model performance is statistically more accurate, since the *p* values [271] have increased in all cases.

The results obtained previous the DE optimization, plotted in Appendix B.3, shown significance differences between the real and the virtual data for the morphological parameters: circularity and roundness, for many of the simulated aggregate´s samples (Table B-5 and Table B-6); since t_s obtained from the statistical analysis is located inside the critical region, ($t > t_{crit}$, being $t_{crit} = 2.262$).

Thanks to DE, this has been fixed for most samples. However, there are still some stones samples; like G2 and S in the case of circularity; and L4 and S in the case of Roundness; who still do not pass the T-test, although their P_{er50} have approached the real P_{er50} (higher p values).

Table D-1 – T-test results with 95% Confidence Interval for Perimeter

P_{er50} of 10 virtual samples	G1	G2	RG	L1	L2	L3	L4	L5	GS	CG	S
1	5.09	2.986	5.08	4.153	2.59	1.897	4.08	4.883	4.846	3.634	1.875
2	5.64	6.08	6.22	4.092	1.74	2.912	6.32	5.098	4.7421	3.125	1.874
3	7.62	5.96	4.76	7.16	1.03	4.656	3.12	7.112	4.7941	4.156	3.632
4	2.14	4.21	6.14	4.69	2.80	2.762	3.35	7.084	4.8335	3.8756	1.857
5	5.81	4.77	8.31	7.256	1.87	2.462	2.97	6.174	4.6833	3.376	1.9873
6	5.71	5.02	8.46	5.094	1.74	2.012	4.11	3.986	4.7425	2.961	2.0432
7	4.22	5.92	4.07	5.37	2.06	3.145	5.27	7.345	4.7348	2.067	3.732
8	3.98	5.29	6.12	4.784	2.87	2.1674	3.23	5.326	4.684	2.87	2.01
9	5.56	4.13	6.93	2.976	1.48	2.761	5.1	4.215	4.804	2.35	1.16
10	4.78	4.63	5.43	4.42	2.88	3.592	3.461	4.512	4.6837	2.1564	4.562
Sample´s mean ($\bar{X}$)	5.06	4.90	6.15	5.00	2.11	2.84	4.10	5.57	4.75	3.06	2.47
Real P_{Per50} value (μ)	6.03	5.50	6.72	5.81	2.13	3.15	4.69	6.21	4.79	3.25	2.11
Sample Standard Deviation (s)	1.437	0.9743	1.435	1.334	0.646	0.826	1.118	1.265	0.061	0.718	1.092
Sample Size (n)	10	10	10	10	10	10	10	10	10	10	10
t	-2.144	-1.948	-1.2515	-1.921	-0.117	-1.199	-1.6647	-1.591	-1.800	-0.848	1.051
P value	0.061	0.083	0.242	0.087	0.909	0.261	0.130	0.146	0.105	0.418	0.321

Table D-2 – T-test results with 95% Confidence Interval for Area

P_{er50} of 10 virtual samples	G1	G2	RG	L1	L2	L3	L4	L5	GS	CG	S
1	3.098	1.89	2.153	2.163	0.124	0.5426	0.984	2.986	1.824	0.535	0.235
2	3.452	2.235	3.052	1.453	0.265	0.642	2.098	2.623	1.8527	0.693	0.242
3	2.845	1.245	2.86	2.874	0.267	0.512	1.853	1.986	1.870	0.704	0.59
4	1.653	1.782	3.153	2.543	0.36	0.572	1.1453	2.83	1.763	0.66	0.214
5	2.153	3.16	2.876	1.635	0.198	0.478	1.24	3.463	1.85	0.92	0.2785
6	1.465	1.17	4.12	1.832	0.213	0.5013	1.356	2.873	1.987	0.78	0.225
7	2.431	2.9	2.642	2.874	0.187	0.5824	1.51	3.874	1.7556	0.603	0.357
8	2.178	2.674	2.341	3.013	0.189	0.562	1.385	2.35	1.813	0.552	0.299
9	1.763	1.68	2.124	2.915	0.174	0.476	0.952	2.786	1.8124	0.637	0.265
10	2.865	1.093	3.523	1.684	0.201	0.476	0.873	3.623	1.986	0.907	0.215
Sample´s mean ($\bar{X}$)	2.39	1.98	2.88	2.30	0.22	0.53	1.34	2.94	1.85	0.70	0.29
Real P_{Per50} value (μ)	2.08	1.56	2.73	2.02	0.24	0.52	1.28	3.12	1.83	0.64	0.25
Sample Standard Deviation (s)	0.662	0.738	0.623	0.613	0.064	0.055	0.395	0.578	0.079	0.134	0.113
Sample Size (n)	10	10	10	10	10	10	10	10	10	10	10
t	1.481	1.81	0.782	1.437	-1.064	0.828	0.479	-0.986	0.865	1.396	1.176
P value	0.172	0.103	0.453	0.184	0.314	0.429	0.642	0.349	0.408	0.196	0.269

Table D-3 – T-test results with 95% Confidence Interval for Roundness

P_{er50} of 10 virtual samples	G1	G2	RG	L1	L2	L3	L4	L5	GS	CG	S
1	0.614	0.741	0.711	0.827	0.557	0.557	0.612	0.529	1.000	0.767	0.672
2	0.637	0.710	0.776	0.813	0.552	0.552	0.720	0.586	0.928	0.702	0.664
3	0.634	0.664	0.716	0.674	0.685	0.685	0.598	0.672	0.967	0.712	0.679
4	0.678	0.692	0.709	0.771	0.617	0.617	0.614	0.714	0.998	0.712	0.613
5	0.750	0.683	0.678	0.654	0.642	0.642	0.634	0.632	1.000	0.660	0.628
6	0.535	0.677	0.770	0.726	0.646	0.646	0.659	0.694	0.901	0.641	0.642
7	0.673	0.841	0.691	0.685	0.672	0.672	0.664	0.730	0.985	0.697	0.721
8	0.621	0.523	0.643	0.675	0.678	0.678	0.628	0.615	0.973	0.713	0.672
9	0.728	0.699	0.709	0.861	0.693	0.693	0.672	0.635	1.000	0.752	0.656
10	0.617	0.725	0.760	0.714	0.613	0.705	0.640	0.670	1.000	0.774	0.640
Sample´s mean ($\bar{X}$)	0.65	0.70	0.72	0.74	0.64	0.64	0.64	0.65	0.98	0.71	0.66
Real P_{Per50} value (μ)	0.65	0.70	0.71	0.69	0.61	0.61	0.61	0.61	1.00	0.65	0.61
Sample Standard Deviation (s)	0.062	0.079	0.042	0.073	0.051	0.054	0.036	0.061	0.035	0.043	0.031
Sample Size (n)	10	10	10	10	10	10	10	10	10	10	10
t	-0.068	-0.181	0.468	2.155	1.594	2.022	3.004	1.946	-2.259	4.656	5.050
P value	0.948	0.860	0.651	0.060	0.145	0.074	0.015	0.083	0.050	0.001	0.001

Table D-4 – T-test results with 95% Confidence Interval for Circularity

P_{er50} of 10 virtual samples	G1	G2	RG	L1	L2	L3	L4	L5	GS	CG	S
1	0.712	0.741	0.712	0.761	0.561	0.608	0.681	0.518	0.995	0.614	0.672
2	0.768	0.710	0.803	0.851	0.572	0.623	0.728	0.612	0.978	0.673	0.664
3	0.680	0.664	0.741	0.741	0.691	0.870	0.578	0.662	0.978	0.624	0.679
4	0.772	0.692	0.717	0.842	0.609	0.560	0.754	0.747	1.000	0.685	0.613
5	0.750	0.683	0.788	0.641	0.612	0.540	0.623	0.699	0.989	0.692	0.628
6	0.794	0.677	0.747	0.659	0.646	0.498	0.641	0.694	0.990	0.647	0.642
7	0.874	0.741	0.790	0.729	0.690	0.480	0.648	0.746	0.988	0.662	0.730
8	0.652	0.723	0.681	0.751	0.720	0.610	0.614	0.721	0.913	0.693	0.672
9	0.692	0.699	0.827	0.732	0.712	0.820	0.669	0.708	0.992	0.624	0.656
10	0.725	0.725	0.773	0.699	0.657	0.728	0.639	0.752	1.000	0.672	0.640
Sample´s mean ($\bar{X}$)	0.742	0.706	0.758	0.741	0.647	0.634	0.657	0.686	0.982	0.659	0.662
Real P_{Per50} value (μ)	0.700	0.670	0.730	0.780	0.680	0.670	0.680	0.680	1.000	0.700	0.680
Sample Standard Deviation (s)	0.064	0.027	0.046	0.068	0.057	0.132	0.053	0.073	0.027	0.030	0.033
Sample Size (n)	10	10	10	10	10	10	10	10	10	10	10
t	2.056	4.180	1.917	-1.834	-1.834	-0.869	-1.351	0.256	-2.114	-1.943	-4.022
P value	0.070	0.002	0.087	0.100	0.100	0.407	0.210	0.804	0.064	0.084	0.003

D.5 Analysis of the virtual compaction force in the model

A series of comparisons between variables using linear regressions analysis, including polynomial curves, has been made to see how the virtual force influences on the other parameters studied (air void, number of contacts and overlapping), to identify the adequate maximum virtual compaction force for the proposed compaction model.

However, it should keep in mind that this model is only a prototype and future research are necessary to deep analyze the virtual compaction model behaviour.

The regression analysis has been done through the Minitab Statistical Software. The standard error of the regression (*S*) and R-squared are the parameters chosen for the evaluation since they are two well-known key goodness-of-fit measures for regression analysis.

D.5.1 Virtual force versus Air void content

Table D-5 – Regression analysis of the virtual force versus Air void content

Model	R-squared	S
Linear	85.7	0.19
Quadratic	88.3	0.18
Cubic	96.5	0.1

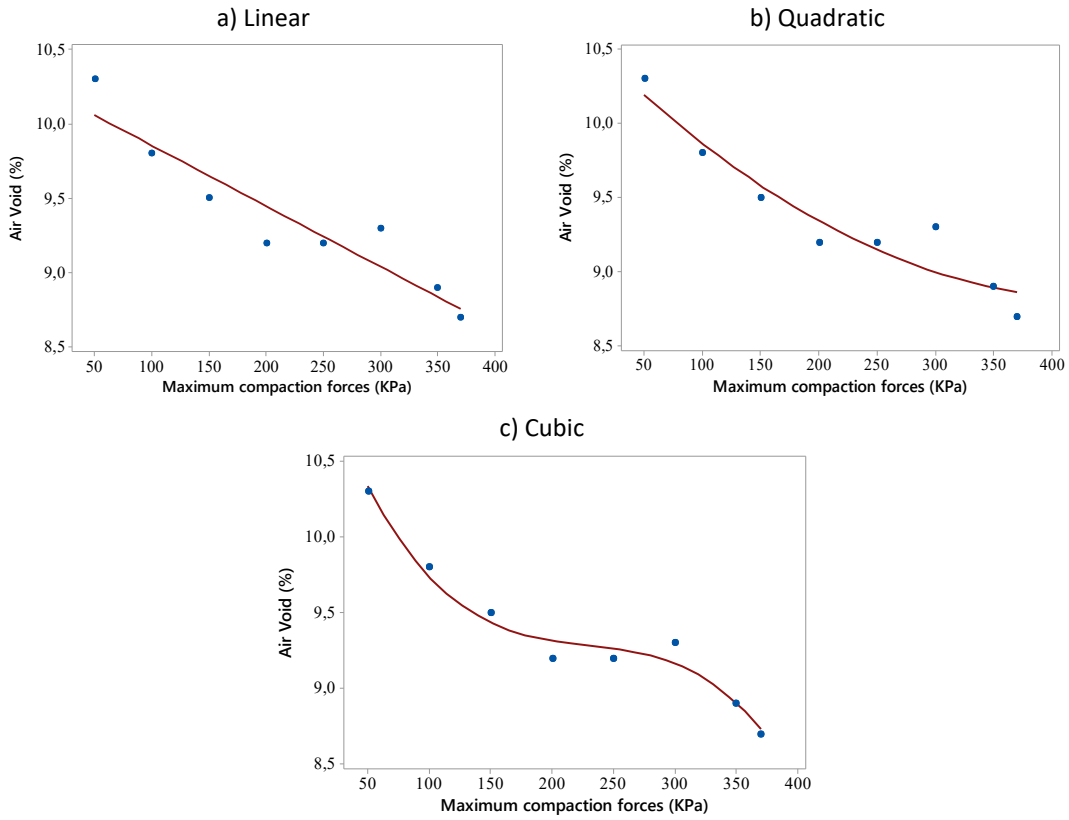
As can be seen from the Table D-5, the cubic regression models is the one which present better results, since the R square has a higher percentage (96.5%) and the S the lowest value (0.1). This latter indicates that the data points are close from the regression line on average. Despite this fact, from the graphics the relation between the pressure applied to the mixture and its air void content tends to a logistic model.

Between 50kPa and 200kPa, the void content gradually decreases from 10.5% to 9% approximately. Between 200kPa and 300kPa the air void content remains

constant (staying at ~9.3%), while from 350kPa the void content decreases gradually again achieving values below 9%.

This could be a clue for future research to study the differences between the Superpave Gyratory Compactor compacting curve, which normally follow a logarithmic curve [272], and the virtual compacting curve, since this topic falls outside the research line of this thesis due to lack of time.

Figure D-6 – Different Linear regression graphs of the air void as a function of the virtual force.



D.5.2 Virtual force versus particle's overlapping

Table D-6 – Regression analysis of the virtual force versus the particle's overlapping

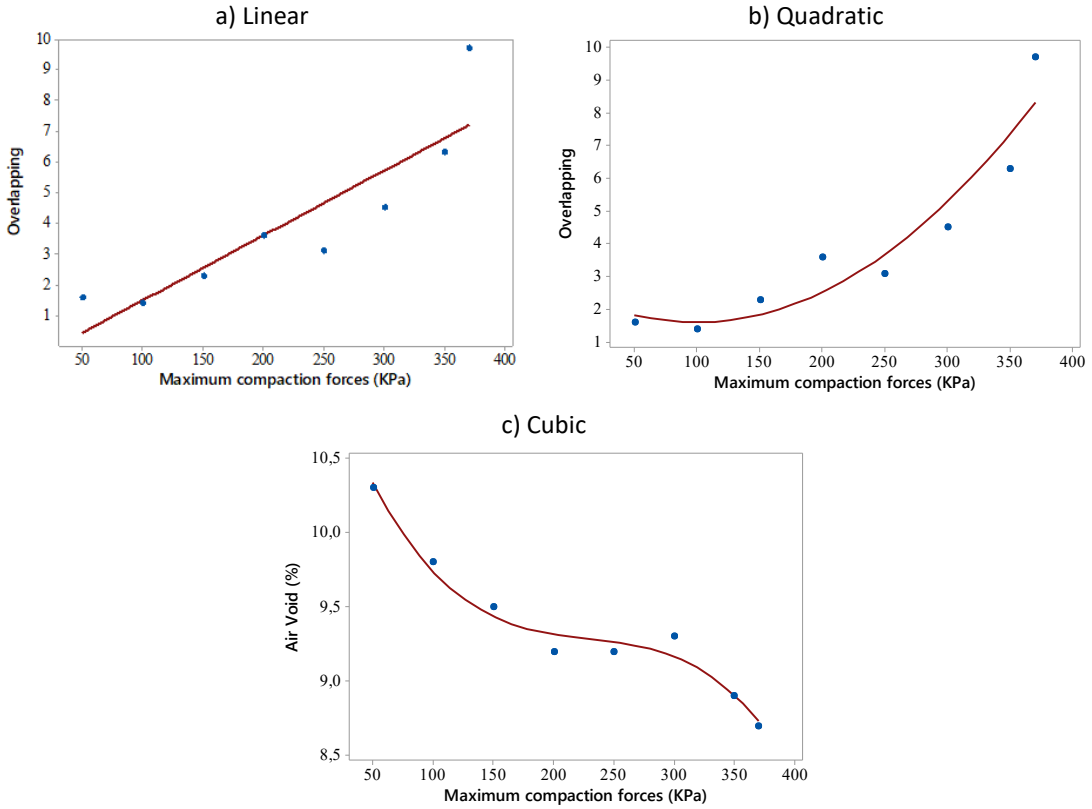
Model	R-squared	S
Linear	76.6	1.4
Quadratic	90.1	1.04
Cubic	93.9	0.91

The results of the Table D-6 shows that the cubic model is the one that best adapts to the data points. Between 50kPa and 100kPa, the overlapping between particles does not vary appreciably, having a value close to 1.5mm. From 100kPa, the overlapping grows exponentially to almost 4mm when the maximum force is 200kPa.

However, between 200kPa and 250kPa, the penetration between particles is very similar. This may be due to i) that the PE is generating numerous impulses in the opposite direction to the movement of the pistons used for compaction, to avoid the overlapping. ii) particles in constant motion causes the measurement of their overlapping to produce ambiguous results difficult to study with certainty.

Exist several elements which could affect the accuracy of the software's outcomes as the speed of compaction, the number of stones in the mixture (notice that, even though the trial tests were done on the same sample, due to the stochastic nature of the aggregate generation algorithm, the number of particles can change each time). The influence of these elements needs to be investigated in future research.

Figure D-7 – Different Linear regression graphs of the overlapping as a function of the virtual force.



D.5.3 Virtual force versus the number of contacts

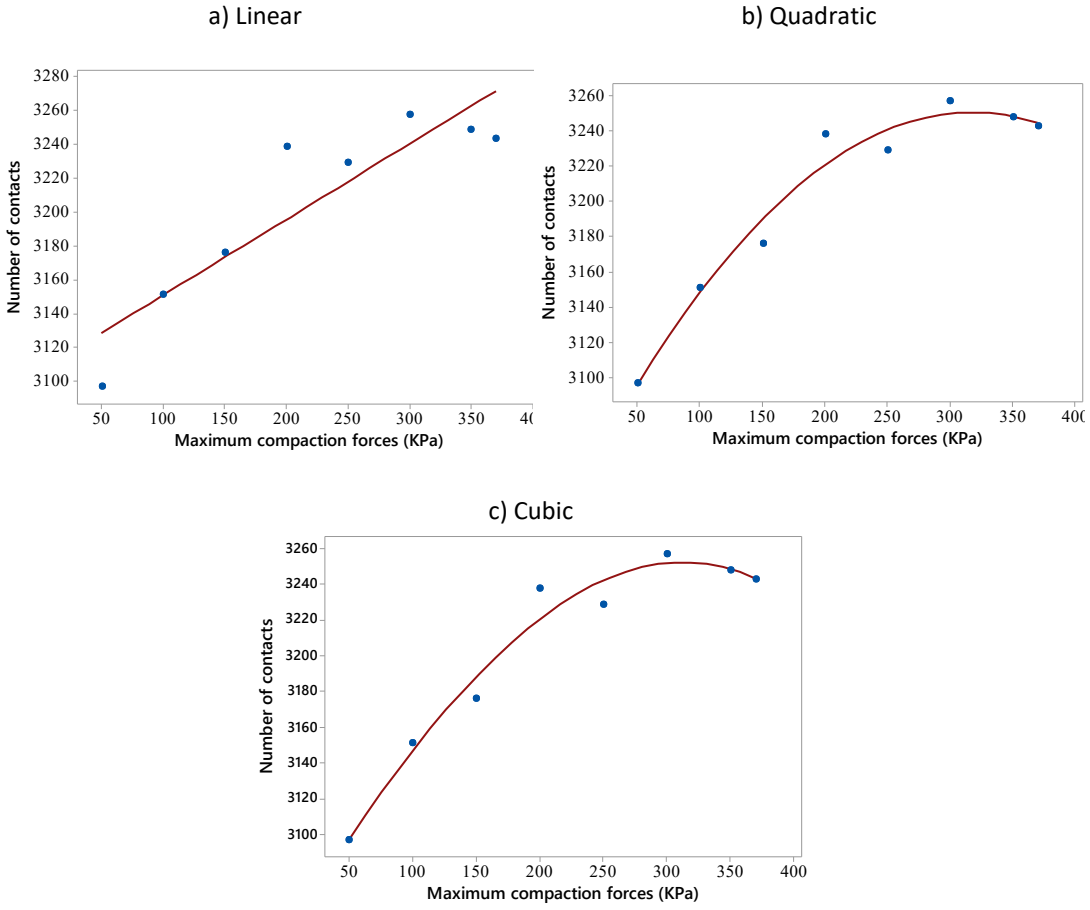
Table D-7 – Regression analysis of the virtual force versus the number of contacts

Model	R-squared	S
Linear	79.6	26.56
Quadratic	97.1	11.7
Cubic	97.0	12.91

From the table it can be seen that the two polynomial regression models give values of R square and S very similar and therefore their corresponding graphs are also graphs are very similar. It does not appear that any of the regression models studied fit well to the scatter of the data points of the variable analyzed. The number of contacts grows constantly for maximum forces between 50kPa and 200kPa.

More in-depth studies with non-linear regression models would be convenient in future research to assess the relationship between the maximum force applied and the number of contacts, since the data seem to tend towards a concave asymptotic curve.

Figure D-8 – Different Linear regression graphs of the number of contacts as a function of the virtual force.



D.6 Analysis of the of the Air Void prediction model

The effects of the independent variables on the predicted porosity of the asphalt skeleton simulation model for porous asphalt was studied through Minitab 18 software. A multiple regression analysis was done considering those variables that could affect the errors presented in the model to analyze their statistical significance.

The general expression of the model´s error is shown in Equations D-5.

$$y = c + \sum_{i}^{n} b_i\, x_i + \sum_{i=1}^{n}\sum_{j=1}^{n} b_{ij}\, x_i x_j \qquad \text{D-5}$$

Where

y is the expected value of the dependent variable, in this case, the error between the predicted air voids content and the actuals one (from now named *AV error*).

c is the value of y when all of the independent variables ($x_i x_j \dots x_n$) are equal to zero

($x_i x_j \dots x_n$) are the independent variables which in this study correspond to: the overlapping between particles, the number of particles simulated, the volume of mastic in the mixtures.

b_i ($i = 1,2 \dots n$) are the regression coeficients, wich represent the change in the dependent variable relative to a one unit change in the respective independent variables.

The errors resulting from the simulation where firstly analyze through a multiple regression analysis through the regression statistic option in Minitab software considering first all the variables involved (number of stones, volume of mastic and overlapping between particle). Then another regression analysis considering only the variable large statistical significance was done.

On the other hand, a Pearson's correlation coefficient (r) between the variables analyzed in the model has been done to quantify the correlation between then. The r values varies between -1 and 1, being 0 when there is no correlation at all, 1 when a perfect correlation and -1 when a perfect inverse correlation.

D.6.1 Multiple regression analysis with three variables

Table D-8 shows the results obtained from the multiple linear regression analysis, differentiating between the statistical significance of the independent variables of the porosity prediction model´s error and the model summary, which quantify how well the regression model explain the correlation between the independent variable and the dependent variable.

In regression analysis, there are two main terms which represent the correlation between the outcome variable y and the input variables x (Equation D-6): the R-square (R-sq) and the standard error (S).

Table D-8 – Statistical significance of the independent variables and model summary

Terms	Coefficient	Standard Error	t-value	p-value	VIF
Constant/ intercept	6.42	3.77	1.7	0.187	-
Volume of Mastic (%)	-0.3524	0.0518	-2.94	0.006	1.20
Number of particles	-0.00798	0.00698	-1.14	0.0236	2.80
Overlapping (mm)	0.176	0.284	0.97	0.0420	2.54
Model Summary	S	R-sq	R-sq (adj)		
	0.84	84.67%	69.34%		

The R-square represents the proportion of variance of y that may be predicted by knowing the independent variables. In this case, an R-sq of 85% (Table D-8) indicates that a large portion of the variation of the error on the air void predicted in the simulation (*AV error*) could be explained by the influence of the independent variables: the volume of mastic in the mixture, the number of particles in the mixture and the overlapping between particles.

The standard deviation (S) is calculated as the square root of variance by determining each data point's deviation relative to the mean and it measures the distance

between the data values and the fitted values in the regression analyses. The lower is its value the more accurate is the model. In this case, a 0.84 correspond to 8.4% of error rate. Thus, the regression model can be considered quite fitted since it has less than 10% error.

Additionally, the adjusted R-square (R-sq (adj)) was also calculated for a later comparison with the regression analysis with two variables, since R-square always increase with the number of variables involved in the analysis even if the additional input variables show no relationship with the output variables. The R-sq (adj) is a correction for the number of variables included in the model, by looking at whether additional input variables (or independent variables) are contributing to the model.

The Constant or intercept in the multiple regression is defined as the mean of the dependent variable when all the independent variables in the model are zero. In that case it is not very relevant as the volume of mastic in the mixtures as well as the number of particles are both required inputs for the simulation model and therefore always different from a zero value.

The coefficient of the independent variables indicates the effect that they have on the error of the model. Is an indicator of how much the error increase (or decrease) when those variable increase one unit. Therefore, the regression equation can be written as:

$$AV\ error = 6.42 - 0.3524\ volme\ of\ mastic - 0.00798\ number\ of\ particles + 0.176\ overlapping \quad \text{D-6}$$

From the table we can see that the error decrease when the volume of mastic and the number of particles increase, whereas the error increases as the overlapping between particles increases.

The highest coefficient is that of the volume of the mastic, which indicates a greater influence on the model error. However, although the coefficient associated with the number of particles seems small, it must be taken into account that an error variation occurs when a single virtual stone is added to the mix. Since the mixtures are made up of a multitude of aggregates, that makes their influence on the model considerable. Nonetheless, to evaluate how large is a coefficient it is necessary to observe the t-value.

The t-value or t-statistic, is the coefficient of the variables divided by the standard error. It evaluates whether the coefficient of the independent variables are significantly different from zero and therefore has more influence on the variation of the dependent variable. From Table D-8 it can be seen that the biggest t-value correspond to the volume of mastic variable, while the smallest one correspond to the overlapping between particle variable.

The p-value represent the statistical significance of the variable in the model, that is, if a significant relation between the error that the model present and the independent variables exist. A p value <0.05 means that the variable are statistically significant at a 95% confidence level. From table it can be seen that the t value of the variable overlapping is very close to 0.05, which indicated a limited influence on the error of the air void predictor model for porous asphalt proposed in this thesis.

The Variance Inflation Factor (VIF) is the measure of the multicollinearity between the independent variables. It is calculated as the ratio of the overall model variance to the variance of the model which include only a single independent variable.

In other words, it measures how much the variance of an estimated regression coefficient increase if the independents variables are correlated.

Values of VIF between 1 and 5 are considered small multicollinearity. Table D-8 reveals that all the VIF have values between 1 and 3, which are inside the acceptable limits to consider that the correlation between the independents factors does not degrades the precision of the regression coefficient. To quantify this correlation between the different variables involved in the simulation, a Pearson correlation analysis is presented in Section D.6.3.

In general terms, the results of the multiple regression analyses showed that the overlapping between particles does not have a significant effect in the air void prediction results, with p-values close to 0.05 and the lowest t-value. On the other hand, it seems that the volume of mastic of the mixtures contribute greatly to explain the error in the simulation model. Therefore, in the next section another multiple regression analysis is presented, considering only those variables which larger statistical significance: the volume of mastic and the number of particles.

D.6.2 Multiple regression analysis with two variables

The regression analysis with three variables presented in the above section seems to indicate that the variable of overlapping between particles does not have a significance influence in the response data. Therefore, a multiple regression analysis without considering the overlapping has been done (Table D-9), to compare the the adjusted R-square (R-sq (adj)) obtained from both models.

Table D-9 – Statistical significance of the independent variables

Terms	Coefficient	Standard Error	t-value	p-value	VIF
Constant/ intercept	9.57	1.93	4.97	0.008	-
Volume of Mastic (%)	-0.1665	0.0494	-3.37	0.028	1.10
Number of particles	-0.001323	0.00436	-3.05	0.038	1.10

Model Summary	S	R-sq	R-sq (adj)
	0.84	79.82%	69.73%

In the previous section it was calculated that the regression with three variables yields an R-squared of 84.76% and an adjusted R-squared of 69.34%. From Table D-9 it can be seen that the regression model with two variables yields an R-squared of 79.82% and an adjusted R-squared of 69.73%.

Although the R-squared value of the multiple regression with 3 variables is higher than the R-squared value of the regression with 2 variables; the comparison between the R-adj between these two regression models seems to indicate that the overlapping between particles should no exert any predictive power on the error on the predicted virtual porosity. As mentioned, the adjusted R-squared looks at whether additional input variables are contributing to the model. The adjusted R-squared in regression with three variables was 69.34% compared to the adjusted R-squared in regression with two variables of 69.73%.

Therefore, the adjusted R-squared is able to identify that the independent variable of overlapping is not helpful in explaining the variance of error in the virtual porosity prediction model. In such a case, the adjusted R-squared would point the model creator to using the regression with two variables rather than regression with three variables.

Thus, by looking at the coefficient plotted in Table D-9, the obtained regression model that could explain the error in the asphalt skeleton simulation, after eliminating

the non-significant coefficients of the overlapping variable, is presented in the next equation:

$$AV\ error = 9.57 - 0.1665\ volme\ of\ mastic - 0.001323\ number\ of\ particles \quad \text{D-7}$$

The negative coefficients in the equation show how the error between the simulated air void content and the experimental one (*AV error*) decrease as the volume of mastic on the mixture and the number of virtual particles generated increase, as already explained in the previous section.

Looking at the t-statistics values of the independent variables in Table D-9, it can be noticed that both coefficients have similar influence on the variance of the output variable (both with values around -3). Besides, their p-values smaller than 0.05 prove the statistical significance of both factors.

Table D-10 – Analysis of variance ANOVA

Terms	Sum of Square
Regression	11.16
Volume of Mastic (%)	8.007
Number of particles	6.545
Residual	2.811
Total	13.981

Additionally, to quantify the amount of variation in the dependent variable attributed by each term of the regression model, the regression sum of square has been calculated through the analysis of variance options (ANOVA) in Minitab software. The larger this value is, the better the relationship explaining the error in the porosity prediction model as a function of the independent variables.

From the Table D-10 it is observed that the volume of mastic on the mixture explain around 8/13.981 (about 57.27%) of the variability in the AV error, whereas the number of particles explain around 46%. Therefore, only a 20% of the AV error variation remains unexplained.

D.6.3 Pearson Correlation between all the variables analysed in the model

For a more in-depth study of the correlation between parameters analysed during the regression model, a Pearson correlation study has been carried out.

As mentioned, the Pearson's correlation coefficient (r) varies between -1 and 1, being 0 when there is no correlation at all, 1 when a perfect correlation and -1 when a perfect inverse correlation.

Table D-11 – Pearson Correlation of the simulation model factors

Terms	AV error (%)	Number of particles	Volume of mastic (%)
AV error (%)			
Number of particles	-0.48		
Volume of mastic (%)	-0.56	-0.31	
Overlapping (mm)	0.33	-0.72	0.01

From Table D-11, it is observed that the number of particles and the volume of mastic have a negative correlation between the errors in the air void prediction model (AV error), as it was discovered by the regression analysis. As well as in the regression analysis, the Pearson correlation study reveals that the volume of mastic is the most significantly related to *AV error*, and overlapping the least.

On the other hand, the overlapping shows a strong inverse correlation with the number of particles (r=-0.72), that is, the overlapping decrease with the increase of the number of particles.

APPENDIX E. ATTACHED CODES AND CODE SOURCE

E.1 Particles samples generation with complex shapes

The scripts for creating virtual particles with realistic shapes can be seen in the script paragraph attached in Appendix E.4, which contains the subtitles: E.4.1 Particles creation with the Marching algorithm (see Section 4.3.1) and E.4.2) Particles measurement with the Raycast function (used for the virtual rocks measurement as explained in Section 4.3.2).

The virtual particle´s area measured through the triangulation mentioned in Section 4.3.1 (Figure 4-5.b) was done through the C- sharp code found in:

https://answers.unity.com/questions/291923/area-of-a-triangle-this-code-seems-to-work-why.html

The Marching Cubes algorithm in c-sharp language to create the triangular mesh of the virtual particles generated was download from:
https://graphics.stanford.edu/~mdfisher/MarchingCubes.html

E.2 Particle´s volume measurement and vibration of particles packing and

The c-sharp code for measuring the volume of the virtual particles based on their triangular mesh, in order to generate number of aggregates according to the weight of the desired sample (see Section 5.4) was found in:

https://stackoverflow.com/questions/1406029/how-to-calculate-the-volume-of-a-3d-mesh-object-the-surface-of-which-is-made-up

The script used for the particles assembly presented in Chapter. 5 is presented in Appendix E.5

E.3 Compression of the aggregate´s skeleton of porous asphalt

The classic Differential Evolution algorithm in c-sharp language, implemented in the aggregates generation model, as explained in Chapter 6, were found at:

https://github.com/mbalchanowski/Differential-Evolution/commit/f4d28a12065b51fd4dbef3cb5e2acb6b7ed4454f

The fitness function script used in the Differential Evolution algorithm (see Section 6.2.1) is presented in Appendix E.6.1.

The air void content calculation of the asphalts skeleton (see Section 6.4.1), is presented in Appendix E.6.2.

The pistons movement scripts for the aggregates skeletons compaction (see Section 6.4) is presented in Appendix E.6.3.

The discarded script which was taken into account the angle of gyration of the Superpave Gyratory Compactor is presented in Appendix E.6.4.

E.4 Particles creation. Chapter 4

E.4.1 Particle creation with the Marching Cube Algorithm

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using MathNet.Numerics.Distributions;
using MathNet.Numerics.Random;
using MathNet.Numerics;
using System.IO;
using System;
/*Rock.cs
public class rockGeneratorSimpleCompleto : MonoBehaviour
{
    public int resolucion = 5;
    public float gridSize = 1;
    public float bolitaSize = 0.1f;
    public float esfericidad = 5;
    public int noiseStep = 6; // Pasos d ruido. Cantidad de veces que se repite el proceso de añadir ruido.
    public float firstIntensityRandom = 3f;
    public float restoRadio;
    public float intensidad = 2;
    public Material material;
    public float[,,] matriz3DAleatoria;
    public float[,,] matriz3D;
    public float[,,] matrizMezclada;
    bool encendido;
    float centro;
    Vector3 centroPosition;
```

```
    public GameObject prefab3DTextAleatoria;
    public GameObject prefab3DTextMatriz3D;
    public GameObject prefabMatrizMezclada;
    PadreRockGenerator padreRockGeneratorComponent;

    [Header("Axis values")]
    [Range(0.00001f, 1f)]
    public float MagnitudeShapeX = 1f;
    [Range(0.00001f, 1f)]
    public float MagnitudeShapeZ = 1f;
    [Range(0.00001f, 1f)]
    public float MagnitudeShapeY = 1f;
    // Use this for initialization
    void Start()
    {
        //Metondo: inizializar()
        encendido = true;
        matriz3DAleatoria = new float[resolucion, resolucion, resolucion];
        matriz3D = new float[resolucion, resolucion, resolucion];
        matrizMezclada = new float[resolucion, resolucion, resolucion];
        centro = Mathf.Floor((resolucion) * 0.5f); //centro de un bloque o piedra = distancia
entre voxel x resolucion/2. Un numero entero.
        Debug.Log("center vale " + centro);
        centroPosition = Vector3.one * centro;
        MezclaDeMatrices();
        Triangular();

        //MEtodo
        MeshCollider meshc = gameObject.AddComponent(typeof(MeshCollider)) as
MeshCollider; // se le añade collider a la roca
        meshc.sharedMesh = null; // sombreado?
        GetComponent<MeshFilter>().mesh.RecalculateBounds();
```

```
    GetComponent<MeshFilter>().mesh.RecalculateNormals(); //
    GetComponent<MeshFilter>().sharedMesh.RecalculateNormals();
    GetComponent<MeshFilter>().sharedMesh.RecalculateBounds();
    meshc.sharedMesh = GetComponent<MeshFilter>().sharedMesh; // Give it your mesh here.
    Physics.defaultContactOffset = 0.000000001f;
    Physics.defaultSolverIterations = 10;
    Physics.defaultSolverVelocityIterations = 1;
  }
  float evaluarAleatorio(float x, float y, float z)
  {
    int EntX = Mathf.FloorToInt(x); // FloorToInt Devuelve el mayor entero que sea menor o igual que su argumento numérico (float en este caso
    int EntY = Mathf.FloorToInt(y);//x es el pasoX, y FloorToInt redondea esa valor.
    int EntZ = Mathf.FloorToInt(z);//coge la parte entera!!!
    float DecX = x - EntX;//sería lo mismo que pasoX-EntX. o 0-EntX. Para coger un trozo de un cuadrado
    float DecY = y - EntY;//sería lo mismo que pasoY-EntY. o 0-EntY. coge los decimales!!
    float DecZ = z - EntZ;//sería lo mismo que pasoZ-EntZ. o 0-En
    //Interpolación en X con el siguiente aleatorio
    float interpXA = matriz3DAleatoria[EntX, EntY, EntZ];//voxel actuale en X
    float interpXB = matriz3DAleatoria[EntX + 1, EntY, EntZ];//voxel siguiente en X
    float interpXF = Mathf.Lerp(interpXA, interpXB, DecX);//Se hace una interpolación lineal
    //Interpolación en Y con el siguiente aleatorio
    float interpYA = matriz3DAleatoria[EntX, EntY, EntZ];
    float interpYB = matriz3DAleatoria[EntX, EntY + 1, EntZ];
    float interpYF = Mathf.Lerp(interpYA, interpYB, DecY);
    //Interpolación en Z con el siguiente aleatorio
    float interpZA = matriz3DAleatoria[EntX, EntY, EntZ];
    float interpZB = matriz3DAleatoria[EntX, EntY, EntZ + 1];
    float interpZF = Mathf.Lerp(interpZA, interpZB, DecZ);
    return (interpZF + interpYF + interpXF) / 3f;//la media de los 3 valores,
```

```
}
void MezclaDeMatrices()
{
    //seria el antiguo void llenarAleatorio
    for (int X = 0; X < resolucion; X++)
    {
        for (int Y = 0; Y < resolucion; Y++)
        {
            for (int Z = 0; Z < resolucion; Z++)
            {
                matriz3DAleatoria[X, Y, Z] = (float)Math.Round(UnityEngine.Random.Range(-
firstIntensityRandom, firstIntensityRandom), 1);


            }
        }
    }
    //seria el antigua void LlenarMatriz3D *****POR PASOS*****
    for (int X = 0; X < resolucion; X++)//x va a valer todos los valores de indice de la matriz
de cero a resolucion. Para saber la posicion de los vovel en la matriz Como están los 3 bucles
anidados se repite
   //para cada uno de los cubos de la matriz de densidad
    {
        for (int Y = 0; Y < resolucion; Y++)
        {
            for (int Z = 0; Z < resolucion; Z++)
            {
                //3,2,1,0,-1,-2,-3
                //Distancia en Voxels del voxel actual al central
                float disx = (centro - X) * MagnitudeShapeX;//Distancia X al centro de la esfera (o
bloque de celdas. Basada en la escala de las x, q va en función de la Weibull
                float disy = (centro - Y) * MagnitudeShapeY;//Distancia Y al centro. Voxel actual
por la escala = punto que equivale al centro del voxel.
                float disz = (centro - Z) * MagnitudeShapeY;//Distancia Z al centro
```

```
            float distanceBlockCenterToVoxel = Vector3.Distance(Vector3.zero, new
Vector3(disx, disy, disz)) * gridSize;//Centro del bloque hasta centro del voxel multiplicado
por la distancia en metros

            Debug.Log("distanceBlockCentreToVoxel vale " + distanceBlockCenterToVoxel);

            float radioEnMetros = (gridSize * resolucion / 2) - restoRadio;//gridSize*1.25 es
para hacer el radio mas pequeño

            matriz3D[X, Y, Z] = (float)Math.Round((-(radioEnMetros - gridSize) +
distanceBlockCenterToVoxel) * esfericidad, 4);

            //esfericidad hace que los numeros positivos sean mas grandes (de un 30 pasa a
un 40, o de -30 a un -40), lo

            //GameObject text3D = Instantiate(prefab3DTextMatriz3D, new Vector3(X *
gridSize, Y * gridSize, Z * gridSize), UnityEngine.Quaternion.identity);

            //text3D.GetComponent<TextMesh>().text = matriz3D[X, Y, Z].ToString();

            //seria el antiguo void Mezcla de matrices


        }

      }

    }

    float paso = 1f;

    for (int pasos = 0; pasos < noiseStep; pasos++)

    {


      float pasox = 0;

      float pasoy = 0;

      float pasoz = 0;

      for (int X = 0; X < resolucion; X++)//x va a valer todos los valores de indice de la matriz
de cero a resolucion. Para saber la posicion de los vovel en la matriz Como están los 3 bucles
anidados se repite

                                  //para cada uno de los cubos de la matriz de densidad

      {

        for (int Y = 0; Y < resolucion; Y++)

        {

          for (int Z = 0; Z < resolucion; Z++)

          {
```

```
                matrizMezclada[X, Y, Z] = (float)Math.Round(matriz3D[X, Y, Z] +
evaluarAleatorio (pasox, pasoy, pasoz)* intensidad, 3);
                //GameObject text3D = Instantiate(prefab3DTextMatriz3D, new Vector3(X *
gridSize, Y * gridSize, Z * gridSize), UnityEngine.Quaternion.identity);
                //text3D.GetComponent<TextMesh>().text = matriz3D[X, Y, Z].ToString();
                //seria el antiguo void Mezcla de matrices
                pasoz = pasoz + paso;
              }
              pasoz = 0;
              pasoy += paso;
            }
            pasoy = 0;
            pasox += paso;
          }
      }
      paso *= 0.55f;
  }



  // Update is called once per frame
  void Update () {
                              }
  void Triangular()
  {
      gameObject.AddComponent<MeshFilter>();
      gameObject.AddComponent<MeshRenderer>();
      Mesh mesh = GetComponent<MeshFilter>().mesh;
      GetComponent<MeshRenderer>().material = material;
      ArrayList ALVertices = new ArrayList(); //Lista de los vértices que hay en cada voxel
(máximo 15 en cada voxel, 3V x 5 triangulos)
      ArrayList ALTriangles = new ArrayList();// lista de triangulos, sirve para mejorar el GS??
(Método 2 del word chapter1?)
```

```
ALVertices.Clear();
ALTriangles.Clear();//Borrar los buffers de los vé
  //**************** TRIANGULACIÓN DE MC**************************
float[] cellVertex = new float[8];
int actualtriangle = 0;
padreRockGeneratorComponent = GetComponent<PadreRockGenerator>();
for (int X = 0; X < (resolucion - 1); X++)
{
   for (int Y = 0; Y < (resolucion - 1); Y++)
   {
      for (int Z = 0; Z < (resolucion - 1); Z++)
      {
         //Creamos la CELDA (a partir de los centros de los voxel)

         //los valores centrales de los voxeles son los nodos del enrejado !!

         //los inputs del marching cube son los voxeles. Son los centros de cada voxel, para luego sacar los nodos (los'vértices de cada celda del cubo del enrejado)  del enrejado

         {
            cellVertex[0] = matrizMezclada[X + 0, Y + 0, Z + 0]; //convierte cada centro de voxel en un nodo del enrejado = vértices de cada celda.
            cellVertex[1] = matrizMezclada[X + 1, Y + 0, Z + 0];
            cellVertex[2] = matrizMezclada[X + 1, Y + 1, Z + 0];
            cellVertex[3] = matrizMezclada[X + 0, Y + 1, Z + 0];
            cellVertex[4] = matrizMezclada[X + 0, Y + 0, Z + 1];
            cellVertex[5] = matrizMezclada[X + 1, Y + 0, Z + 1];
            cellVertex[6] = matrizMezclada[X + 1, Y + 1, Z + 1];
            cellVertex[7] = matrizMezclada[X + 0, Y + 1, Z + 1];

            //Debug.Log("ceelV vale " + cellVertex[5]);
         }
```

```
            //Determine which cube is in front of us (actually the index in the table)

            //los nodos pueden tener valor  de densidadd positivo bit=0 o negativo bit=1. cada voxel tiene un valor float de densidad

            //el indice de los vertices son 8 bit ????

            //valores temporales son los valores centrales (su densidad) de un voxel.si es mayor que cero

            // vtable index son los bit concatenados de 0 a 255 y sabemos que triangulos generar buscando en la tabla de casos grande

            int VtableIndex = 0;

            if (cellVertex[0] > 0) { VtableIndex |= 1; }

            if (cellVertex[1] > 0) { VtableIndex |= 2; }

            if (cellVertex[2] > 0) { VtableIndex |= 4; }

            if (cellVertex[3] > 0) { VtableIndex |= 8; }

            if (cellVertex[4] > 0) { VtableIndex |= 16; }

            if (cellVertex[5] > 0) { VtableIndex |= 32; }

            if (cellVertex[6] > 0) { VtableIndex |= 64; }

            if (cellVertex[7] > 0) { VtableIndex |= 128; }


            //El operador |= se usa if you want to store 8 binary states in a single byte for example to be used in a network viewed envir

            if ((VtableIndex > 0) && (VtableIndex < 255))

            {

               int iterations = 0;

               //desde este for hasta  float v2 = VertexPositions[nIndex0, 2] * (1 - t) + VertexPositions[nIndex1, 2] * t;

               //es para buscar en las tablas


               // Cantidad de triangulos que habrá: CubeTriangles[VtableIndex, cTriangles]

               for (int cTriangles = 0; padreRockGeneratorComponent.CubeTriangles[VtableIndex, cTriangles] >= 0; cTriangles += 3)

               {

                  iterations++;// iteracion de los vertices generados para crear triangulos para saber en que posicion se encuentra (donde la densidad es igual a 0 a los largo
```

```
                    // de las aristas (superficie))
                    // PARA SABer en que posicion están los vertices de cada caso de
poligonizacion MC

                for (int ci = 0; ci < 3; ci++) //CI = CUBE INDEX
                {
                    int nEdge = padreRockGeneratorComponent.CubeTriangles[VtableIndex,
cTriangles + ci];
                    int nIndex0 = padreRockGeneratorComponent.EdgesTable[nEdge, 0];
                    int nIndex1 = padreRockGeneratorComponent.EdgesTable[nEdge, 1];

                    float t = (0 - cellVertex[nIndex0]) / (cellVertex[nIndex1] -
cellVertex[nIndex0]);

                    float v0 = padreRockGeneratorComponent.VertexPositions[nIndex0, 0] *
(1 - t) + padreRockGeneratorComponent.VertexPositions[nIndex1, 0] * t;
                    float v1 = padreRockGeneratorComponent.VertexPositions[nIndex0, 1] *
(1 - t) + padreRockGeneratorComponent.VertexPositions[nIndex1, 1] * t;
                    float v2 = padreRockGeneratorComponent.VertexPositions[nIndex0, 2] *
(1 - t) + padreRockGeneratorComponent.VertexPositions[nIndex1, 2] * t;

                    //como MC devuelve la distancia entre 0 y 1 de los vértices de cada
triángulo, lo multiplico por la distancia qeu yo quiero en metros.
                    //Son los ejes de cada triangulo multiplicado por una distancia en metros.
Se agrandan los triangulos sin que pierdan su forma.
                    //x mas vo para que no salgan todos desde el origen.
                    v0 = (X + v0) * gridSize;
                    v1 = (Y + v1) * gridSize;
                    v2 = (Z + v2) * gridSize;

                    ALVertices.Add(new Vector3(v0 - centro, v1 - centro, v2 - centro));

                }
```

```
                ALTriangles.Add(actualtriangle);
                ALTriangles.Add(actualtriangle + 1);
                ALTriangles.Add(actualtriangle + 2);
                actualtriangle += 3;

            }

        }


    }

  }

}
```

//TrimToSize: this method can be used to minimize a collection's memory overhead if no new elements will be added to the collection.

```
ALVertices.TrimToSize();
ALTriangles.TrimToSize();
Vector3[] tempVert = (Vector3[])ALVertices.ToArray(typeof(Vector3));
int[] tempTriangles = (int[])ALTriangles.ToArray(typeof(int));

//   Debug.Log(actualtriangle);
Debug.Log("Vertex " + tempVert);
//   Debug.Log("Index " + ALTriangles.Capacity);

mesh.Clear();
mesh.vertices = tempVert;
mesh.triangles = tempTriangles;
mesh.RecalculateBounds();
```

```
        mesh.RecalculateNormals();


    }
    private void OnDrawGizmos()
    {
        if (encendido)
        {
            for (int x = 0; x < resolucion; x++)
            {
                for (int y = 0; y < resolucion; y++)
                {
                    for (int z = 0; z < resolucion; z++)
                    {
                        float valor = 0;


                        valor = matrizMezclada[x, y, z];


                        Gizmos.color = new Color(valor, valor, valor);
                        Gizmos.DrawSphere(new Vector3(x * gridSize, y * gridSize, z * gridSize) -
centroPosition, bolitaSize);


                        Gizmos.color = new Color(((float)x / (float)resolucion), ((float)y /
(float)resolucion), ((float)z / (float)resolucion), 1f / (float)resolucion);
                        Gizmos.DrawCube(new Vector3(x * gridSize, y * gridSize, z * gridSize) -
centroPosition, Vector3.one * gridSize);
                    }
                }
            }


        }
    }
}
```

E.4.2 Particles measurement with the Raycast function

```
using System.Collections;


using System.Collections.Generic;

using UnityEngine;

public class RaycastLasr : MonoBehaviour

{

   public Transform cuboTransform;

   Vector3[] posicionesLineasEscaneoCircular; //guarda punto impacto raycas circular, laser-borde del cubo

   Vector3[] posicionesLineasEscaneoLineal;//Guarda punto de impacto del raycast, exterior del cubo (perpendicular+vector3.up) hasta borde cubo

   Vector3[] perpendicular; //Guardamos los puntos del segmento del eje (moveToward)del punto mas izquierda y mas derecha

   public int cantidadLineas = 12; //Resolucion escaneo circular

   public int cantidadPerpendiculares = 12;//resolucion escaneo lineal

   Vector3 puntoMasIzquierda; // NO LAS INICIALIZAMOS A CERO, porque los puntos de impacto pueden estar a la izquierda o a la derecha de cero.

   Vector3 puntoMasDerecha;


   // Use this for initialization

   void Start()

   {

      //Inicializar los array de vector3 a un tamaño determinado.

      posicionesLineasEscaneoCircular = new Vector3[cantidadLineas];

      perpendicular = new Vector3[cantidadPerpendiculares];//puntos sobre la linea (los segmentos)

      posicionesLineasEscaneoLineal = new Vector3[cantidadPerpendiculares]; //punto de impacto raycast desde arriba hasta borde cubo


      EscaneoCircular();

      CalculoPerimetro();
```

```
        EscaneoLineal();


        //Ejemplo donde el primer elemento es considerado el mayor y el menor, antes de seguir comparándolo
        //5,6,8,2,9
        //may=5
        //min =5
    }
    // Update is called once per frame
    void Update()
    {
        //ESCANEO CIRCULAR


            for (int i = 0; i < cantidadLineas; i++)
        {
            Debug.DrawLine(posicionesLineasEscaneoCircular[i], cuboTransform.position); // dibujo de lineas de escaneo circular (area del proyecto)
            if (i < cantidadLineas - 1)
            {
                Debug.DrawLine(posicionesLineasEscaneoCircular[i], posicionesLineasEscaneoCircular[i + 1]); //dibujo del perimetro
            }
        }
        Debug.DrawLine(posicionesLineasEscaneoCircular[0], posicionesLineasEscaneoCircular[cantidadLineas - 1]); //ultimo segmento del perimetro, hay que ponerlo a manor porque numero de array no coincide con numero lineas



        //ESCANEO LINEAL
        Debug.DrawLine(puntoMasDerecha, puntoMasIzquierda, Color.blue);// EJE MAYOR o menor  FERET


        for (int i = 0; i < cantidadPerpendiculares; i++)
```

```
    {
        Debug.DrawLine(perpendicular[i], perpendicular[i] + Vector3.up, Color.green);
        //Debug.DrawLine(perpendicular[i] + Vector3.up, posicionesLineasEscaneoLineal[i], Color.black);

    }

    //Tsk2: dibujar perpendiculares hacia arriba(y) en cada punto del array del perdicular, de 1m de largo.
  }

  void EscaneoCircular()
  {
    float grados = 360 / (float)cantidadLineas;
    for (int i = 0; i < cantidadLineas; i++)
    {
        RaycastHit hit;//es una variable (u objeto) tipo raycasthit a la que llamamos hit
        if (Physics.Raycast(transform.position, cuboTransform.position - transform.position, out hit) == true)//cuboTransform.position - transform.position s la dirccion dl vector (destino menos origen)
        {
            posicionesLineasEscaneoCircular[i] = hit.point;//aqui guardamos en el array la info del impacto: con hit.point guardamos la posicion que es lo que nos interesa ahora.

            //posicionesLineas [i]= transform.position; //en lugar de guardar la posicion del objeto, como en script laser, tenemos que guardar la posicion del impacto
            //ahora guardamos los puntos mas a la derecha y mas a la izquierda del cubo.
            if (i == 0)
            {
                puntoMasIzquierda = posicionesLineasEscaneoCircular[0];
                puntoMasDerecha = posicionesLineasEscaneoCircular[0];
            }
            else
```

```
            {
                if (posicionesLineasEscaneoCircular[i].x < puntoMasIzquierda.x)
                {
                    puntoMasIzquierda = posicionesLineasEscaneoCircular[i];
                }

                if (posicionesLineasEscaneoCircular[i].x > puntoMasDerecha.x)
                {
                    puntoMasDerecha = posicionesLineasEscaneoCircular[i];
                }
            }
        }
        transform.RotateAround(cuboTransform.position, Vector3.up, grados);
        Debug.Log("l hit" + i + "sta n" + posicionesLineasEscaneoCircular[i]);
    }
}
void CalculoPerimetro()
{
    // SACAR LA MAGNITUD DEL PERÍMETRO
    float sumaPerimetroTotal = 0;
    for (int i = 0; i < cantidadLineas - 1; i++)
    {
        Vector3 segmento = posicionesLineasEscaneoCircular[i + 1] -
posicionesLineasEscaneoCircular[i];//para sacar un vector de otros dos es vector destino-
vector origen. Dentro del bucle para que lo haga con todos
        sumaPerimetroTotal += segmento.magnitude;
    }
    Vector3 ultimoSegmento = posicionesLineasEscaneoCircular[cantidadLineas - 1] -
posicionesLineasEscaneoCircular[0];
    sumaPerimetroTotal += ultimoSegmento.magnitude;// . magnitude porque no le
posdemos sumar a un float un vector3, sino su magnitude.
```

```
    print(" el perímetro es " + sumaPerimetroTotal);
    print("el punto mas a la izquierda es " + puntoMasIzquierda);
  }
  void EscaneoLineal()
  {
    //******PRIMERA PARTE******
    //Primer paso del escaneo lineal es: con los puntos sacados del escameo circular encontrar los puntos mas a la izqd y derecha.
    //hacemos un moveToward para que vaya recorriendo a segmentos el eje entre punto mas a izquierda y mas a derecha, para luego tirar perpendiculares hacia arriba
    Vector3 eje = puntoMasDerecha - puntoMasIzquierda;
    float tamanoSegmento = eje.magnitude / (float)cantidadPerpendiculares;
    Vector3 puntoQueSeMueve = puntoMasIzquierda;
    for (int i = 0; i < cantidadPerpendiculares; i++)

    {
      puntoQueSeMueve = Vector3.MoveTowards(puntoQueSeMueve, puntoMasDerecha, tamanoSegmento);
        perpendicular[i] = puntoQueSeMueve;
      //***********2º PARTE*****
      //Una vez sacadas el array de puntos del segmento, el cual sirve para dibujar perpendiculares hacia arriba en este caso, en el update
      //las lineas verdes dibujadas en update solo para comprobar que el punto de origen de raycast es correcto (solo visual)
      //tenemos el punto de origen desde el cual lanzamos el raycast.
      RaycastHit hit;
      if (Physics.Raycast(perpendicular[i] + Vector3.up, -Vector3.up, out hit))
      {
        posicionesLineasEscaneoLineal[i] = hit.point;
      }

    }
```

```
    }
    void OnDrawGizmos()
    {
        Gizmos.DrawWireSphere(puntoMasDerecha, 0.1f);
        Gizmos.DrawWireSphere(puntoMasIzquierda, 0.1f);
        //Task1: Spherita wired en cada punto de perpendicular. nuevo for!
        /*for (int x = 0; x < cantidadPerpendiculares; x++)
                {
                        Gizmos.DrawWireSphere (perpendicular [x], 0.1f); //las variables perpendicula y cantidadPerpendicular están definidas ocmo variables globales, podemos las podemos usar aqui
                }*/
    }
}
```

E.5 Particles assembly. Chapter 5

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class shakepro : MonoBehaviour {

	public float amplitudx = 0.004f;
	public float amplitudy = 0.004f;
	public float amplitudz= 0.0006f;
	public float velocityx = 50f;
	public float velocityy=50f;
	public float velocityz = 0f;
	Vector3 initialPosition; //(actualposition es el vector3 inicial de mi probeta
	float timex=0;
	float timey=0;
	float timez=0;

	// Use this for initialization
	void Start () {
		initialPosition = transform.position; // para que la posición actual vaya cambiando. El problema es que se teletransporta.
	}

	// Update is called once per frame
	void Update () {
		//acumula en x y z la cantidad de tiempo que hay transcurrido por la velocidad. PARA QUE PROBETA VAYA SUAVE
		timex += Time.deltaTime * velocityx; // Time delta time para que elcubo no se teletransporte (en vez de Time.time)
```

```
                //(le vamos sumando fragmentos de tiempo que hay entre cada fotograma
multiplicado por la velocidad.) Ver ejemplo inferior en gris.

                timey += Time.deltaTime * velocityy;

                timez += Time.deltaTime * velocityz;


                /*float tiempoy = Time.time * velocidady;

    float tiempox = Time.time * velocidadx; //Time, es general donde están los elementos
del tiempo. time (minúscula) es la hora.

    float tiempoz = Time.time* velocidadz;*/


                transform.position = initialPosition + new Vector3 ( Mathf.Cos (timex) *
amplitudx * 0.5f,

                                        Mathf.Cos (timey) * amplitudy * 0.5f,

                                        Mathf.Cos (timez) * amplitudz * 0.5f);

        }

}
```

E.6 Asphalt skeleton compression. Chapter 6

E.6.1 Fitness Function of the Differential Evolution algorithm

```
using System;
using System.Collections;
using UnityEngine;
using Utilities;
namespace Optimisation
{
   class FitnessFunctions
   {
      public static float[] realRockData;
      public static float[] rockMeasurements;
      public static float[] p50Actual;
      public static int numRocksToGenerate;
      public static float percentile;
      public static float fitnessOutput;
      /************************ Fitness function & helpers ************************/
      //
      /// <summary>
      ///     We analize chromosome quality and asing it to fitnessOutput value of each
individual chromosome in current population
      /// </summary>
      ///
      public static IEnumerator FitnessFunction(Chromosome individual)
      {
         SpeedTest.StartTest(false);
         yield return FitnessFunction(individual.genes, 0);
         SpeedTest.EndTest("Full population analisis: ", false);
      }
```

```
    /// <summary>
    ///     The overall fitness function: generate rocks, take measurements,
    ///     calculate percentiles and compare with the real rock data
    /// </summary>
    ///
    public static IEnumerator FitnessFunction(float[] individual, int id)
    {
        // initialise the allRockMeasurements array
        float[][] allRockMeasurements = new float[Settings.NUM_MEASUREMENTS][];
        float[] rockMeasurementsAnalizer = new float[Settings.NUM_MEASUREMENTS];
        for (int i = 0; i < Settings.NUM_MEASUREMENTS; i++)
            allRockMeasurements[i] = new float[numRocksToGenerate];


        // generate a number of rocks, storing their measurements
        for (int i = 0; i < numRocksToGenerate; i++)
        {
            //RockBuilder.GenerateRock(individual, true);
            RockOptimiserOnline.instance.rockSpawners[i].SetGenes(individual);
            RockOptimiserOnline.instance.rockSpawners[i].WeibulReScale();
            RockOptimiserOnline.instance.rockSpawners[i].CreateRandomPoints();
            RockOptimiserOnline.instance.rockSpawners[i].Preview();
            float[] results = RockOptimiserOnline.instance.rockSpawners[i].Analize();
            // after ProcessRock(), this.rockMeasurements contains measurements for the
current rock
            // get the centre of the rock using MeshCollider.bounds.center
            //yield return RockBuilder.ProcessRock();
            //float[] results = RockBuilder.measurementResults;
            for (int j = 0; j < Settings.NUM_MEASUREMENTS; j++)
            {
                allRockMeasurements[j][i] = results[j];
                //Debug.Log(((MeasurementIndex)j).ToString() + " - GenerateCut: " + results[j] + "
/ GenerarCorte: " + rockMeasurementsAnalizer[j]);
```

```
            }
        }
        yield return null;//??????
        // Debug.Break();
        // calculate fitness by comparing percentiles of the virtual and real rocks
        float fitness = EvaluateFitness(allRockMeasurements, realRockData);
        //Debug.Log("Single fitness: "+Math.Round(fitness, 4));
        //Debug.Log("Fitness w/out round: "+fitness);
        //float zxRatio = individual[(int)ChromosomeIndex.MAGNITUDE_SHAPE_Z] / individual[(int)ChromosomeIndex.MAGNITUDE_SHAPE_X];
        // penalise if ratio between magnitudeShapeX and magnitudeShapeZ is not valid
        if (float.IsNaN(fitness) /*|| zxRatio < 0.5 || zxRatio > maxZXRatio*/)
            // if the fitness is returned as NaN, it means the rocks are very invalid
            // therefore we consider these inputs to be bad, and so just set fitness to infinity
            fitness = float.PositiveInfinity;
        // TODO: 0.5 and 0.7 as parameters
        //float oldFitness = fitness; // debug
        //fitness *= 5; // penalty is multiplying fitness value by 5
        //Debug.Log("WRONG Z/X RATIO | X = "+magnShapeX+", Z = "+magnShapeZ+" | original fitness = "+oldFitness+" | new fitness = "+fitness);
        fitnessOutput = fitness;
    }
    /// <summary>
    ///     Returns the fitness value of rocks generated based on some input parameters
    ///     Call once rocks have been generated and measurements taken
    /// </summary>
    /// <param name="measurements"> D*N array where D is number of attributes and N is number of rocks generated </param>
    /// <param name="realRockPercentiles"> Array of P50s for the real rock data, for comparison </param>
    // D = 7 (perimeter, area, aspect ratio, minor axis, major axis, sphericity, roundness)
    static float EvaluateFitness(float[][] measurements, float[] realRockPercentiles)
```

```
    {
        float errorMagnitude = 0;
        //float p50spht=float.PositiveInfinity; // debug
        //float p50roundness=float.PositiveInfinity;
    //    Debug.Log("TIRADA " + numeroTirada);
      //  Debug.Log("Cantidad Rocas: " + measurements[0].Length);
        //Array initialization
        p50Actual = new float[realRockPercentiles.Length];
        // calculate P50 for each attribute and calculate squared difference between that and the real rock P50
        for (int i = 0; i < Settings.NUM_MEASUREMENTS_OP; i++)
        {
            float p50 = CalcUtil.CalcPercentileAttr(measurements[i], percentile);
            //float sqDiff = Mathf.Pow(p50 - realRockPercentiles[i], 2);
            float sqDiff = (realRockPercentiles[i] / p50 ); //El diferencial ahora es cuanto se aleja de 1
            p50Actual[i] = p50;
           // Debug.Log(((MeasurementIndex)i).ToString() + " - Calculado: " + p50 + " / Real: " + realRockPercentiles[i] + " / Diferencia: " + sqDiff);
            errorMagnitude += Mathf.Abs(1-sqDiff); //Acumulamos las diferencias a 1(que es el valor ideal)
        }
       // Debug.Log("RESULTADO TIRADA " + numeroTirada++ + ": "+errorMagnitude);
        return errorMagnitude; // return the sum squared error as the fitness value
    }
    static int numeroTirada = 1;
  }
}
```

E.6.2 Calculation of the air voids content

```
public class AirVoidCalculation
{
   float rocksVol = RockGenerator.instance.GetVolumeOfAllRocks();
   float bitumenPercent = float.Parse(uiCompresionPanel.percentBitumenText.text) / 100f;
   float bitumenVol = (uiCompresionPanel.totalMass * bitumenPercent) / 1.04f;
   float fillerVol = float.Parse(LibraryController.instance.uiController.fillerText.text) / 2.7f;
   float totalMassSample;
   float bitPercent;
   float dustPercent;
   float bitumenDensity = 1.03f;
   float RockDensity = 2.7f;
   float rocksVol = RockGenerator.instance.GetVolumeOfAllRocks();// volumen de las piedras
generadas
   float bitumenVol= (totalMass * bitumenPercent100) / (100f - bitumenPercent100) /
bitumenDensity;
   float theoricalRockMass= totalMass - bitumenMass;
   float realSimulatedMass = rocksVol * RockDensity;
   float noSimulatedRockMass= theoricalRockMass - realSimulatedMass;// volumen del dust
otra forma
   float noGeneratedRockVol = CalcVolMass.noSimulatedRockMass /
CalcVolMass.genericRockDensity;
   float simulatedSampleVol = CalcVolMass.totalRocksVolByMesh
                  + noGeneratedRockVol
                  + bitumenVol;
   //Volumen cilindro
   float radio = 10 / 2;
   float area = (rad * rad) * Mathf.PI;
   float volumeCilinder = area * CompresionSkeleton.currentDistance();
     //Air Vol Percent
    float airCont= (totalVol - simulatedSampleVol) / volumeCilinder;
}
```

E.6.3 Pistons movement

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

//*************************************************
//Compresión de la muestra a una presión determinada
//*************************************************
public class CompresionSkeleton : MonoBehaviour
{
   public Transform bottomTransform;
   public Transform topTransform;
   float initialForce = 0.01f;
   float currentForce = 0.01f;
   float voidAirVol = AirVoidCalculation.airCont;
   private Vector3 initialUpPosition;
   private Vector3 initialDownPosition;
   private float initialDistance = 0;
   private float pressHeigth = 0.01f;
   private float currentDistance = 0;
   private float realHeigthSample = 0;
   private bool moveByForce;
   private bool stopByPenetrationFlag = false;

     void Start()
   {
      Initialize();
   }

   //Corrutina de espera de tiempos para movimiento por velocidad y por preseión
```

```
IEnumerator CompressionProcess()
{
    float pressHeigth = 0.001f;
    float speedPressMovement = 0.011f;
    float currentDistance = Vector3.Distance(bottomTransform.position,
topTransform.position) - pressHeigth;
    Vector3 finalPressPosition = bottomTransform.position + (Vector3.up *
currentDistance);

    do
    {
        currentDistance = Vector3.Distance(bottomTransform.position,
topTransform.position) - pressHeigth;

        //Movimiento de la prensa inferior por velocidad
        bottomTransform.position = Vector3.MoveTowards(bottomTransform.position,
finalPressPosition, speedPressMovement * Time.deltaTime);
        //Movimiento de la prensa superior por velocidad
        topTransform.position = Vector3.MoveTowards(topTransform.position,
finalPressPosition, speedPressMovement * Time.deltaTime);

        //esperamos
        yield return null;

    }
    while (voidAirVol < 35);

    //Movimiento por fuerza

    if (voidAirVol < 35)
    {

        if (currentForce < 3.5f)
```

```
            currentForce += 1.2f * speedForceMultiplier;
        else
        {
            compressing = false;
            currentForce = initialForce;

        }
    }

}
private void Initialize()
{

    initialUpPosition = upPressObject.transform.position;
    initialUpRotation = upPressObject.transform.rotation;
    initialDownPosition = downPressObject.transform.position;
    initialDownRotation = downPressObject.transform.rotation;
    initialSpeedPressMovement = speedPressMovement;

}

internal float CalculateOverlaping()
{
    float totalOverlaping = 0;

    ShowPenetration[] penetrationArray = FindObjectsOfType<ShowPenetration>();
    for (int i = 0; i < penetrationArray.Length; i++)
    {
        totalOverlaping += penetrationArray[i].CalculatePenetration();
    }
```

```
        if (penetrationArray.Length == 0)
            return 0;
        totalOverlaping /= penetrationArray.Length;
        return totalOverlaping;
    }
    internal float GetHigth()
    {
        if (currentDistance == 0)
            currentDistance = Vector3.Distance(downPressObject.transform.position,
upPressObject.transform.position) - (pressHeigth);

        return currentDistance * 100;
    }
}
}
```

E.6.4 Discarded script of the pistons movement with rotation and angle of gyration

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PrensaRotation : MonoBehaviour
{
   //public float velocidadRotacion = 0;
   //public float anguloDeGiroEjeZ= 0;
   //public float velocidadEjeZ = 10;
   //   public float thrust;
   //   public Rigidbody rb;

   // Use this for initialization
   //void Start ()
   //   {
   //      rb = GetComponent<Rigidbody>();
   //}
   // Update is called once per frame
   //      void Update () {
   //       rb.AddForce(0,thrust, 0,ForceMode.Acceleration);

   //                float zValue = Mathf.PingPong (Time.time*velocidadEjeZ, anguloDeGiroEjeZ
* 2) - anguloDeGiroEjeZ;

   //                //transform.Rotate (xValue*Time.deltaTime, velocidadRotacion *
Time.deltaTime, zValue*Time.deltaTime);
   //                //transform.rotation=Quaternion.Euler (0,velocidadRotacion*Time.time,
zValue);

   //                transform.localRotation = Quaternion.Euler (0, velocidadRotacion *
Time.time, zValue);
```

```
    //                  Debug.Log ("z value" + zValue);

    //      Physics.gravity;

    //                          }
    //
}
```