Material and Machine Computation of Designed Granular Matter

Rigid-Body dynamics simulations as a design tool for robotically-poured aggregate structures consisting of polygonal concave particles

Karola Dierichs¹, Achim Menges²
¹Institute for Computational Design, University of Stuttgart, Germany ²Institute for Computational Design, University of Stuttgart, Germany
¹http://karoladierichs.net ²http://icd.uni-stuttgart.de
¹karola.dierichs@icd.uni-stuttgart.de ²achim.menges@icd.uni-stuttgart.de

Abstract. Loose granulates are a relevant yet rarely deployed architectural material system. Their significance lies in their capacity to combine fluid-like amorphousness with solid-like rigidity, resulting in potential architectural structures capable of continuous reconfiguration. In addition aggregates allow for functional grading. Especially if custom designed concave particles are used, full-scale architectural structures can be poured using a six-axis industrial robot, combining the precise travel of the emitter-head with the self-organizational capacity of granular substances. In this context, the paper proposes Rigid-Body Dynamics (RBD) simulations as a design-tool for the robotic pouring of loose granular structures. The notions of material and machine computation are introduced and RBD is explained in greater detail. A set of small tests is conducted to investigate the advantages and disadvantages of a specific RBD software. Conclusively, further areas of research are outlined.

Keywords. Material and machine computation; aggregate architectures; designed granulates; robotic pouring; Rigid-Body Dynamics.

INTRODUCTION

Aggregate architectures
Aggregates are defined as large numbers of elements in loose contact (Cambou 1998; Duran 2000). They are commonly known from natural systems like sand or snow (Ball 2004; Nicot 2004; Bagnold 1954; Rognon, Chevoir and Coussot, 2008). In an architectural context however they are rarely deployed in their unbound state, but frequently used as an additive in concrete construction. The few architectural precedents range from the areas of building physics, where the granulate is used as an insulation material, to geo-engineering, vernacular architecture and more recent experiments in form-finding (Gaß and Otto, 1990; Houben and Guillaud, 1994; Treib 1996; Hausladen, de Saldanha and Liedl, 2006; Hensel and Menges 2006a-d; Trummer 2008; Dierichs and Menges, 2010a; Dierichs and Menges, 2012; [1]; [2]). However, a consistent architectural
approach to deploying loose granulates as an architectural material system in its own right has so far not been developed.

Aggregates have both the ability to flow like a liquid and to bear loads like a solid, rendering potential architectural structures entirely reconfigurable. They also show the capacity to be functionally graded to meet specific architectural requirements within one and the same material system (Hensel and Menges, 2008a-b; Hensel, Menges and Weinstock, 2010; Dierichs and Menges, 2012). If custom-made concave-hull particles are used in combination with a digital emitter-head such as a six-axis industrial robot, full-scale architectural structures can be poured, combining the precision of the pouring-paths with the self-organizational behavior of the granulate. Consequently, one of the fundamental differences to conventional architectural material systems lies in the process of designing with the aggregate itself. Usually, building elements have a specific place, that is a definite location and topological relation, in the overall architectural assembly that can be assigned by the developing architect. In an aggregate however, each element finds its own place and it is the role of the designer to observe and interact with this process of formation. Hence in the development of an architectural aggregate system, the tools of observation take a central role. Both material and machine computation in the form of analogue experiments and digital simulations can be used for this purpose.

In this context, the research presented here investigates the relevance of a specific machine computational model, the Rigid-Body Dynamics (RBD) simulation method, as a design tool for robotically-poured aggregate structures consisting of designed particles with a polygonal, concave hull [Figure 1].

In the first part of the paper, the notion of material and machine computation is described. Secondly, different machine computational models for loose granulates are discussed in order to lay-out the relevance of the RBD method, in the given context which is introduced in greater detail. As a case study, a simulation series using an RBD implementation software package is presented. The simulation is aimed at preparing the pouring-paths and -patterns of up to 50,000 particles using a six-axis industrial robot. To conclude, the achieved contributions are highlighted and an outline of further areas of research in this context is given.

MATERIAL AND MACHINE COMPUTATION

Computation is in principle information processing. This process of computation can happen through a machine such as a personal computer, but also through the material itself, where information is gathered by collecting data from the physical process. In the widest sense, machine computation encompasses all procedures that use a mathematical model as the basis of their information process; material computation denotes all procedures that use a physical substance as the source of information (Dierichs and Menges, 2010b).

Figure 1
Aggregate structure composed of 10,000 designed particles (A), detail of robotically poured aggregate structure (B), robotic pouring process using a specifically designed emitter (C).
Both material and machine computation need to be deployed in combination for the comprehensive investigation of granular substances. Whereas physical models allow for the exact observation of large granular masses, mathematical computer models render numerical results more easily, e.g. about the micro-mechanical behavior of a given granulate. Digital simulations however can also be laid out in such a way that they are sufficiently light computationally to allow for predictions about the behavior of large granular arrangements prior to an actual physical experiment.

**MACHINE COMPUTATION OF LOOSE GRANULATES**

Several machine computational methods have been developed for the simulation of granular matter, each being designed to solve a specific problem at low computational expense. The two most well-known mathematical models are the Discrete Element Method (DEM) and the Event-driven Molecular Dynamics Method (ED), the former being designed for the computation of soft, circular particles, the latter for sparse particle masses consisting of hard spheres (Cundall and Strack, 1979; Luding 1994). Other models include the Discrete Simulation Monte Carlo (DSMC) that allows for probabilistic simulations of granular arrangements at relative rest (Pöschel and Schwager, 2005).

**RIGID-BODY DYNAMICS SIMULATIONS**

Most of the above-mentioned techniques are aimed at computing granulates consisting of circular or spherical particles. Simulating polygonal and especially polygonal concave granulates however is computationally very heavy due to the complexity of contact-point determination, which is a much lighter procedure in an aggregate consisting of circles or balls. In this context, the convex hull of a set of points is defined as the minimal point-set, that allows for interior angles between its vertices to be lower than 180 degrees, in a concave hull these angles are higher (Berg et al. 2008).

The Rigid-Body Dynamics method however is developed for the computation of granulates consisting of hard, polygonal particles where static friction and fragmentation is occurring, such as for example railway ballast (Pöschel and Schwager, 2005).

The RBD method is thus, by nature of its algorithm, more suited to compute polygonal concave particles than for example the DEM method. This is due to the fact that in RBD simulations interaction forces need not be known and static friction can be calculated accurately. Whereas DEM is based on the evaluation of interaction forces, RBD is based on the contrary idea: the interaction forces are inversely computed from consistency requirements on the behavior of the particles. The RBD algorithm thus goes through the following steps of contact detection, collision treatment, collision list clear-up, formulation of the contact network and finally the computation of forces and the integration of the equations of motion (Featherstone 2008; Pöschel and Schwager, 2005).

Several known algorithms such as Dantzig's algorithm or the Runge-Kutta algorithm are used to solve sub-sets of this basic algorithm (Pöschel and Schwager, 2005). RBD is mainly deployed in gaming software applications due to its computational speed and is frequently combined with a simplified convex or concave hull-model to further accelerate the results.

**CASE STUDY: RBD SIMULATIONS OF ROBOTICALLY-POURED AGgregate ARCHITECTURES**

The case study presented here is composed of a series of four tests investigating the relevance of a Rigid-Body Dynamics gaming software as a design tool to simulate the behavior of robotically-poured particles with a polygonal, concave hull. Previous simulations with a DEM application have been conducted, to allow for a benchmarking between the two mathematical models (Dierichs, Fleissner and Menges, 2011). The aim of the simulation is to pre-evaluate robotic pouring paths and to render the exact point-to-point coordinates of the aggregate-emitter.
**Test 01: Experiment and simulation of a particle in guided fall**

The first experimental series compares the physical experiment of a single concave particle with a digital simulation using Rigid-Body Dynamics. The aim is to calibrate the simulation parameters to match the physical behavior of the particle as closely as possible. This small experiment is necessary as the software environment offers only sliders instead of measurable values for input.

For that purpose, a cross-formed magazine is placed on top of a box measuring 110 x 110 mm. The particle is filled into the magazine and released. The process is filmed and photographed using time-lapse photography and repeated a minimum of ten times. The same test set-up is modeled within the Rigid-Body environment. The particle shows a bounce of ca. 5 mm height, ca. 1 degree rotation if the particle is released without disturbance and a falling speed of ca. 1 second [Figure 2].

The Rigid-Body Dynamics model is adjusted to match the behavior of the particle, setting the values to mass: 5.000, static friction: 0.2, dynamic friction: 0.2, bounciness: 0.05, damping: 1.0, impulse in x, y and z-direction: 0.25.

This small test can serve well to quickly adjust parameters between physical realm of the experiment and the virtual one of the simulation. It is however not meant to replace a rigorous engineering computation, where accurate values can serve as input parameters rather than sliders. Furthermore, special care needs to be taken with the impulse control, as the following tests will show. If impulse is set equal on all particles, the evolving arrangement will become more monotonous than in the physical experiment, therefore the impulse value needs to be varied between particles, e.g. from 0.25, 0.25, 0.25 in one particle to -0.25, -0.25, -0.25 in another.

**Test 02: Solver benchmarking**

Three different solvers - the Runge-Kutta, the Runge-Kutta Adaptive and the Midpoint solver - are available in the Rigid-Body Dynamics application used here. The difference mainly lies in their accuracy of contact-point detection versus speed, Midpoint being the fastest and potentially least accurate and Runge-Kutta Adaptive being the slowest and potentially most accurate.

Using the parameters established in Test 1, a solver-benchmarking is set-up within the pouring-space of the robot. It models 343 particles arranged in curvilinear, stacked pouring-paths. The Rigid-Body Dynamics simulation is run once with each solver to compare their performance with regards to the concave-hull particle model. A selective probe of each solver-run is tested for the amount of mesh intersections. For 300 frames of the same simulation set-up, the Midpoint solver needs 8:50.3 minutes with 18 mesh intersections in the probe field, the Runge-Kutta takes 10:36.8 minutes with 20 mesh intersections in the probe field and the Runge-Kutta Adaptive 15:36.3 minutes with 17 mesh intersections in the probe field.

This solver benchmarking has proven the statement by the software provider with regards to speed of the respective solvers, Midpoint being the fastest, Runge-Kutta lying in the medium range and Runge-Kutta Adaptive being the slowest. The accuracy in terms of mesh intersections does not vary as much as the processing time from solver to solver, the amount of intersections differing only slightly. This latter observation might need to be verified on larger numbers of particles.

**Test 03: Forward simulation of a linear robotic pouring path**

Test 03 is aimed at comparing an actual robotic pouring-process with the corresponding Rigid-Body Dynamics simulation. A six-axis industrial robot is...
equipped with a magazine emitter-head. The head is filled with 25 particles, tilted in the z-axis and moved along the x-axis at 75% speed. The particles fall out to form a linear arrangement.

The KRL control-points are converted into points within the coordinate space of a 3D modeling environment using a parametric software. These allow for exact modeling of the magazine emitter head rotations as a basis for the Rigid-Body dynamics simulation. The simulation is set-up such that the concave particles become active only at the point of the magazine emitter-head. This is due to the fact that simulating the movement within the tool takes too long to compute. Particles number 1, 5, 10, 15, 20 and 25 are given an axis tilt of 0.05 and -0.05 alternating in the x-direction. The resulting simulated model is close to the physical result. However the aggregate formations are more ordered than their physical counterpart [Figure 3].

**Figure 3**
Rotations of the robotic magazine emitter-head converted from KRL using a parametric software tool (K. Dierichs and T. Schwinn) (A) and Rigid-Body Dynamics simulation result (B).
Figure 4
Pouring pattern for the comparative simulation. The pouring pattern is embedded in the robot's operating field (A). The operating space measures ca. 4.80m in diameter (B). Six rows of particles are inserted into this space (C).

Figure 5
The resultant formations after the first run of 02:24:30.5hrs (A), the second run of 1:52:08.3hrs (B), and the third run of 1:57:37.9 hrs (C).

Figure 6
Close-up of the outer circle in the fourth quadrant of the robotic pouring-space showing the first run with 12 particles (A), the second one with 11 (B) and the last one with 10 (C). The patterns vary on this scale of observation.
This can be further calibrated using a higher degree of variation in the impulse direction of the individual particles. Further tests using 3D scanning as a basis for comparison are currently being conducted. Conclusively this test shows the need for a flow-controlled emitter-head in context with digital simulations, which does not work on gravity but emits one particle at each point in time as well as for a more carefully orchestrated disruption of the particle impulse direction.

**Test 04: Comparative simulation of a single curvilinear pouring path**

Using the Runge-Kutta Adaptive solver as well as the particle parameters set-out in Test 01, a specific curvilinear pouring pattern is tested three times with the exact same settings [Figure 4].

This is to investigate, if patterns remain the same or vary within the simulation under the same conditions. For 1000 frames, the first run lasted 02:24:30 hrs, the second 1:52:08.3 hrs and the third 01:57:37.9 hrs [Figure 5, Figure 6].

The three formations are very similar but not entirely the same. Variation happens mainly on a smaller scale: for example in the fourth quadrant of the outer circle, there are 12, 11 and 10 particles respectively and their formation is varying from run to run. The overall formation however is self-similar with the top and bottom end fanning out and the middle part packing tightly.

**Summary of the test series**

The series presented has shown four tests investigating different aspects of Rigid-Body simulations for robotic pouring of designed granulates with a non-convex hull. The first one served to calibrate simulation values of a single particle, the second one benchmarked different Rigid-Body Dynamics solvers. Test 03 is the first validation of an actual robotic pouring process and a simulation using KRL input points as the basis for the simulation itself. Test 04 compared three sets of the same simulation.

It has been shown, that variation of the impulse-direction of each particle needs to be carefully orchestrated to simulate the actual behavior of the granulate. Furthermore, the need of a flow-controlled emitter-head has become clear to allow for a closer match between a computationally light simulation and the robotic experiment. Solver methods can be chosen to accelerate results respectively.

**CONTRIBUTIONS**

Within the context of Aggregate Architecture the notion of material and machine computation has been introduced and Rigid-Body Dynamics software from the gaming environment has been presented as a possible design-tool for robotically poured aggregate structures consisting of particles with a concave hull. Several tests have been performed to investigate a specific software package as a viable design tool, including parameter calibration with physical experiments, solver benchmarking and repetitive pattern observation.

**FURTHER RESEARCH**

Further research will be mainly conducted into an even closer streaming of data between the robotic pouring process and the Rigid-Body Dynamics simulation environment. The initial focus will be on a flow-controlled emitter head for the six-axis industrial robot that renders the pouring process more similar to what can be simulated at low computational expense, in the following phase, the results from the pouring process and the Rigid-Body Dynamics simulation will be more accurately compared using 3D scanning as well as pattern recognition.

**ACKNOWLEDGEMENTS**

The authors would like to thank Tobias Schwinn, Institute for Computational Design, University of Stuttgart and Michael Preisack, RoboLab, University of Stuttgart for their support with the robotic pouring process as well as Florian Fleissner, Institute of Engineering and Computational Mechanics, University of Stuttgart for DEM simulations in Pasimodo.
REFERENCES
Rognon, P G, Chevoir, F and Coussot, P 2008, ‘Rheology


