Quarterly Report 1

by

Naudé Thomas Conradie

Thesis presented in partial fulfilment of the requirements for the degree of Master of Engineering (Mechatronic) in the Faculty of Engineering at Stellenbosch University

Supervisor: Dr. M. P. Venter

February 2020

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: ....2020/02/27..........

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This report serves to describe the work done on my thesis project in the first quarter of the year 2020. This report was made using the Stellenbosch University Thesis LaTeX template, as a large portion of the material will be used in or adapted for the final report.

This quarterly report consists of the literature review, a description of the materials testing process, and the pipeline for the modelling of basic elements.

1.1 Background

Manufacturing capabilities have greatly increased over the past decades with the advent of three dimensional (3D) printing and other advanced manufacturing technologies. Development of new products and systems may now be limited by design capabilities. Improving and creating new design methods may lead to innovative designs not yet seen before [2].

The field of soft robotics is particularly suited for creative and novel approaches to the design of components. Soft robotic geometries are often highly complex and free-form. Soft robotic actuators may move in imprecise and non-trivial manners [3]. Materials used in the construction of soft robotic components may behave with non-linear responses[4]. Novel soft robotic actuator designs with new and non-trivial behaviours are actively being created and implemented [5].

A computationally efficient generative design approach may be used to explore the design space of soft robotics. Generative design is a powerful automated approach to design that evaluates the performance of many different potential design solutions. Generative design is useful when performing manual design evaluations may become tedious or impossible within realistic time constraints [6].

Chapter 2

Literature Review

2.1 Soft Robotics

The field of soft robotics is relatively new and developmental. Soft robots inherently differ from traditional robots. They are constructed from compliant and pliable materials. Soft robots generally have fixed joints and locomotive actuators between joints, as opposed to traditional robots which usually have locomotive actuated joints connected by rigid sections. [3]

Soft robots offer many advantages over traditional robots. They provide a safer working environment around humans and with fragile materials, as they are much lighter, more pliable and move slower and with less force than traditional robots. They require lower tolerances for manufacture due to their inherent less precise nature. They are much more lightweight compared to traditional robots. [3]

2.1.1 Actuators

Actuators are components that cause controlled motion, generally used in robotics and machinery [7]. There are currently a few major types of soft robotic actuators in use [4].

2.1.1.1 Actuator Types

Shape Memory Alloys (SMA) are metallic alloys capable of being formed into a specific shape while above an inherent transformation temperature, as well as being formed into another shape below the transformation temperature. When the material is then heated or cooled above or below the transformation temperature, it reforms into those respective shapes. This property exists due to the transition between the martensite phase of the material below the transformation temperature, and the austenite phase above the transformation temperature. SMA actuators are heated by applying a current directly to the material. SMA actuators are small, lightweight, silent, and have a high force-to-weight ratio. When shaped straight, they can exert high forces, but only achieve small displacements relative to their length. When coiled, they can extend more, but exert smaller forces. [8]

Shape Memory Polymers (SMP) are similar to SMAs, consisting of smart polymers with the same shape memory properties, instead of metallic alloys. The initial is shape is determined during the manufacturing process. The transformed shape is obtained by cooling the SMP and shaping it as desired. SMPs use electricity or light as a heat source for transformation [9]. They have a high deformation capacity and shape recovery. They are lighter, cheaper and easier to produce than SMAs. They are limited in size due to their low recovery stresses [10; 9].

Dieelectric/Electrically-Actuated Polymers (DEAP) consist of layers of polymers interspersed with conductive material. When the conductive material receives an electrical input, a chemical reaction occurs that causes a change in volume across the layers. This causes the layers to bend in a predetermined direction. DEAPs are lightweight, silent and use little energy. They are biocompatible and functional in water. They are well-suited to mimicking real muscles. Their reactions under high voltages are not fully understood yet and accurately modelling them is highly complicated. [11]

Electro-Magnetic Actuators (EMA) make use of magnetic microparticles within a polymer matrix. The particles are manipulated to cause motion by an external magnetic field from an electromagnet. This allows for a wide range of motion by varying the orientation and magnitude of the electromagnetic field. EMAs are small, require low voltages and are efficient. They have quick response times and high dynamic ranges. They are still an emerging technology in the early stages of development. [12]

Fluid Elastomeric Actuators (FEA) use soft polymeric structures with internal geometry designed for specific types of motion when driven by fluid pressure. Fluid pressure may be obtained from pressurized containers or chemical reactions. They are simple to design, manufacture and control, and are lightweight and usually inexpensive. They are scalable to different sizes and resistant to many types of damage. [13; 14]

2.1.1.2 Actuator Shapes

Multiple types of soft robotic actuators implement linear extension. This allows for an increase in reach and activation of levers or switches. Torsional extension is a variant of linear extension. Torsionally extending actuators twist as they extend, resulting in an angular difference between the ends of the actuator. [3]

FEAs can be built to curl while contracting and straighten out while expanding, similar to natural muscles. This has a range of applications, especially when multiple of these FEAs are used in conjunction with one another. One application is as a gripper, where a number of these FEAs are arranged similarly to a hand or tentacles all curling inward. Grippers are well-suited to picking up and manipulating soft and/or irregularly shaped objects. [3]

Non-trivial actuators are manufacturable. A bimodal FEA was developed by Ellis in 2020. The actuator has a crimped paper strip acting as a strain limiter embedded within a stiff layer of Ecoflex 0030. The stiff layer is alongside multiple Mold-Star 15 cells. When pressurised by a linearly increasing single pressure source, the actuator first curls inwards, then back outwards while extending linearly. [5]

2.2 Soft Robot Modeling

To prevent having to physically construct every design of a soft body, a computationally efficient and accurate digital model can be constructed.

2.2.1 Modeling Approaches

The Finite Element Method (FEM) is an approach to numerically solving field problems. Field problems require the determination of one or more dependent variables' distribution in space. Field problems are mathematically described with differential equations or integral expressions. Finite elements can be expressed as small parts of a larger body. A field quantity within an element may only have a simple spatial variation, such as being described by polynomial terms no higher than the second order. FEM differs from calculus as calculus uses infinitesimal elements. FEM thus delivers approximate solutions. [15]

Voxels are three dimensional (3D) pixels. Voxels are usually cubical. Realistic physical properties may be applied to voxels in specific software packages, such as the free VoxCAD software. Bodies may be constructed and undergo deformation when constructed from voxels in these software packages. It is relatively simple to construct bodies from voxels. [16; 17]

The Gaussian Mixtures representation uses a density field analogy to a level-set method. The density field is initialised as having zero density. Points of density with Gaussian falloff are added within the field. The points may have positive or negative weights. Positive weights contribute to the density, while negative weights subtract from it. If a single point was used, a solid sphere would be the thresholded result.

2.2.2 Modeling Software

Several commercial software packages capable of realistically modeling soft bodies are available.

LSDyna is a FEM software package widely used in industry. It is owned by ANSYS and maintained by LST. The software's code is based on highly non-linear and transient dynamic FEM with explicit time integration [18].

Siemens NX 12 is an integrated software package capable of performing FEM analysis. The software package has a user-friendly interface for graphical design of components [19].

MSC.Marc Mentat is a pre- and postprocessing software for the MSC.Marc FEM solver. It is focused on nonlinear material modeling and analysis. It has an extensive set of options available for post-processing [20].

2.3 Evolved Virtual Bodies

Optimization of discretely represented mathematical model problems, such as virtual bodies, is an active area of research.

2.3.1 Generative Design

An early approach to generative design used a model that encompassed the domain of all components and their possible interactions to design physical bodies with desired behaviours. This approach divided the behaviours and constructed appropriate design fragments for them. These fragments were incrementally composed until a design that obeyed the desired behaviours was obtained. [6]

A generative encoding is a type of encoding that specifies the construction of a phenotype. A genotype is a programmed representation of a potential individual or problem solution. A phenotype is a set of characteristics of an individual resulting from the composite of its genotypes [21]. It may scale well because of its inherent self-similar and hierarchical structure. [22]

Data base amplification is the generation of seemingly complex objects from very concise descriptions [23].

2.3.1.1 Evolutionary Algorithms

Evolutionary algorithms, also known as genetic algorithms, are a robust approach to solving optimization problems. They derive their name from their similarity to the evolution of biological organisms. They typically start with a randomly generated population of solutions to a given problem. Each individual solution's suitability is checked. The more suitable solutions are used to generate a new population, by "breeding" existing members of the population, and introducing some random variations. The new population is checked again, and this process is repeated for some number of generations [24]. Evolutionary algorithms are versatile. They can be applied to the optimization of functions and the evolution of complex behaviours and bodily morphologies. They have been specifically applied to the evolution of robotic bodies for many years [21; 25].

Evolutionary algorithms require a measure of the population's performance. Within the context of evolving simulated physical bodies, a realistic physically simulated goal is usually set. Examples include traversing the greatest distance within a set amount of time, jumping or climbing over an obstacle, or drawing another object closer to it. Fitness measures may also be implemented as survival criteria in testing, such as energy requirements, size, and complexity of the respective bodies. [21; 25]

Evolutionary algorithms typically use direct encodings of solutions. They may struggle to successfully design highly complex systems using direct encodings. [22]

2.3.1.2 Lindenmayer Systems (L-systems)

define L-systems

L-systems were originally conceived as a mathematical theory of plant development. They are used in theoretical biology to describe and simulate natural growth processes. They did not originally include enough detail to completely model higher-level plants. They focused on plant topology and not geometry. [26; 23]

The main component of L-systems is rewriting. Rewriting is used to define complex objects by successively replacing parts (letters) of an initial, simple object (word) according to a set of rules (grammar). Grammars are applied in parallel and simultaneously replace all letters in a given word [23]. This makes L-systems suitable for describing and generating fractal structures. Words generated by L-systems can be used as genotypes for virtual bodies [26].

Growth functions describe the number of letters in a word in terms of its derivation length. Growth functions of DOL-systems are independent of the order of the letters in a word and its derived words. [23]

There are many variations of L-systems. Deterministic and context-free L-systems (DOL-systems) use edge rewriting to replace polygon edges with figures and node rewriting to operate on polygon vertices. Stochastic L-systems implement randomization to obtain variation in productions. A context-sensitive L-system's productions' expression may depend on its predecessors' context. [23]

Partial L-systems use the notation of non-deterministic context-free L-systems (OL-systems) to define the different possible structures of a given type that can develop. They capture the main traits that characterise a structural type and provide a formal basis for their classification. [23]

Original L-systems are discrete in time and space. Model states are known only at specific time intervals and only a finite number exist. Parametric L-systems allow for infinite model states due to the assignments of continuous attributes to model components. Parametric L-systems are not limited by all values being reduced to integer multiples of a unit segment. [23]

Map L-systems allow for the formation of cycles in a production. Maps are finite sets of regions. Regions are surrounded by boundaries consisting of finite, circular sequences of edges meeting at vertices. Each edge has one or two vertices associated with it (only one if the edge forms a loop). Edges cannot cross without forming a vertex. There are no vertices not associated with an edge. Every edge is part of the boundary of a region. The set of all edges is connected. [23]

Some simulations of the branching patterns often achieved by L-systems consider the interactions among the growing features, structures and environment. This makes models more realistic and introduces some complexity. [23]

2.3.1.3 Compositional Pattern Producing Networks (CPPN)

define and discuss CPPNs

2.3.1.4 Emergent Properties

Emergent properties occur when not all components of a given property satisfy that property. An emergent property is not satisfied by the constituent components of a system, but is satisfied by the overall system. If the required condition for a specific property to exist can be determined, it is possible to construct a system satisfying that property from components that do not satisfy that property. If the target property of a system and the property of a component is known, it can be determined if the other component can have a property that will result in the system satisfying the target property. If that property exists, it can be found. [27]

Reactive systems consist of interconnected sub-components that are a part of structural links defining communication methods. These systems may exhibit emergent properties that are unpredictable even when complete knowledge of the systems is accessible. This implies the systems are complex in such a manner that they cannot be simplified to rules based on inferences from their properties. Knowledge of the rules of interactions between the sub-components is also necessary. [28]

Emergent properties are sometimes encountered with generative design. Emergent properties may be complex behaviours that are difficult to predict [28] and challenging to understand initially, that arise from the combination of the simple elements and rules used to construct the generative design algorithms. For example, virtually evolved bodies may end up being complexly constructed in such a way that the methods of completing their objectives are not initially obvious [29]. These emergent properties are desirable, as one advantage of generative design processes is that they may arrive at original and unique designs that may be extremely difficult for a human to arrive at [21].

2.3.2 Previous Work

In 1994, Karl Sims introduced the concept of evolving three dimensional virtual bodies with the aid of evolutionary algorithms. Four simulations with different fitness evaluation functions were run. The four functions were swimming as far as possible within a limited time period, moving across a flat surface as far as possible within a limited time period, jumping as high as possible from a stationary position, and following a light source. The bodies were simply modelled. Nodes describe the rigid parts of the body, the joints between parts and their parent parts, and the set of connections they have to other nodes. Rigid parts have specific dimension. Joints have different types and limits of motion. Node connections describe a part's relationship to its parent. The bodies are described as a set of nodes. These bodies are evolved in three dimensions according to their performance against one of the simulation fitness functions using evolutionary algorithms for a number of generations. The best-performing models were inspected at the end [21]. Further tests were done where two bodies were evolved in the same environment. The bodies had to compete with each other to keep a cube as close to themselves and as far away from the other body as possible. This study investigated the effect of competition between organisms on evolution and optimisation [25].

In 2012, Rieffel et al applied evolutionary algorithms to soft bodies. They used NVidia's PhysX engine to model soft tetrahedral meshes. The physics engine allows for the manipulation of a material's stiffness and damping coefficient. Three properties of the bodies were used to control the evolution. The properties were the body shape, body motion, and material properties. Three different simulations were run. For each one, one of the three properties were fixed, while the other two were allowed to evolve. The mesh resolution was also manipulated. Higher resolutions allowed for smoother surfaces at the cost of computing power. The final bodies are 3D printable, but lack a simple actuation mechanism. [30]

Also in 2012, Hiller and Lipson modelled soft amorphous bodies using the Gaussian mixtures representation. Gaussian points are listed in a 3D workspace. Densities and falloff ranges are associated with the points. This results in smooth, free-form shapes. Material properties are stored in the points and distributed accordingly between points. The Gaussian mixtures representation is a computationally efficient method of storing and representing complex smooth shapes. Shapes were evolved using evolutionary algorithms. The Gaussian points' coordinates, densities, falloff ranges and material indices were used as parameters of interest during evolution.[31]

In 2013, Cheney et al used voxels to model soft bodies in VoxCAD. VoxCAD is free voxel-modelling software. Soft bodies were evolved using CPPN-NEAT. CPPN morphologies were found to appear natural and produce interesting and varying results. Three materials were used for the construction of the soft bodies. The three materials differed in hardness, being compliant, partially compliant and stiff. Compliant and partially compliant voxels acted as the actuators. Soft bodies were tasked either with traversing along a linear path or squeezing through a tight space. Simulations were run multiple times with different penalty functions. Penalty functions included costs for actuated voxels, voxel connections and the total number of voxels. It was found that with differing penalty functions, different bodies evolved and performed better. Final bodies are 3D printable. They are actuated by placing them within a pressure chamber where the pressure is sinusoidally varied. This limits their practical application. [16; 17]

Chapter 3

Material Testing

3.1 Motivation

3.1.1 Material

Mold Star 15 SLOW is selected as the main material to be digitally modelled for the purposes of the project. Mold Star 15 is selected because of its availability and properties. Mold Star 15 is deliverable to the premises where testing is to be done and available from a registered supplier. Additional properties are listed in 3.1 below.

Table 3.1: Given Material Properties [1]

Material	Cost/kg	Pot life (min)	Cure time (hr)	Tensile strength (MPa)
Mold Star 15 SLOW	R332.00	50	4	2.7579

Mold Star 15's long pot life allows for adequate time to prepare specimens thoroughly. The two components of the material need to be mixed according to a 1:1 ratio and stirred until completely mixed. The material then needs to be degassed, poured into the moulds, degassed again, and levelled.

Mold Star 15 has a hyper-elastic non-linear response. It is suitable for inflation while being capable of supporting itself at the relevant scale of construction.

3.1.2 Testing

The necessary properties to correctly model Mold Star 15 are not available from the supplier. Testing is done according to ISO 37 and ISO 7743 standards for tensile and compression testing respectively to obtain data to construct an accurate Ogden model for the material [32; 33].

3.2 Testing Procedure

Chapter 4

Software

4.1 Motivation

Siemens NX 12 is used to create the geometry of the soft elements. NX 12 is very user friendly with simple and easy-to-use interfaces for drawing. NX 12 is capable of exporting geometric files to formats that other FEM software are capable of importing.

MSC.Marc Mentat 2019 is used to model the materials and their behaviour. Marc is capable of advanced non-linear analysis.

Python 3.6 is used to construct a modelling pipeline. Packages are available that allow for the integration of Python, NX 12 and Marc. Many advanced numerical analysis packages are also available for Python.

4.2 Approach

4.2.1 Method

A 5x5 grid of square elements is drawn in NX 12. The geometry is exported to Marc Mentat, where boundary conditions, material properties, and other necessary settings are applied. Boundary conditions are selected as a fixed left and bottom side and a negative pressure ramp exerted on the other surfaces. Figure 4.1 displays the grid and boundary conditions.

A random internal element is selected for deletion. The job is run and the results compiled for inspection.

4.2.2 Pipeline

Figure 4.2 displays a flowchart describing the overall pipeline for the creation of the basic soft elements.

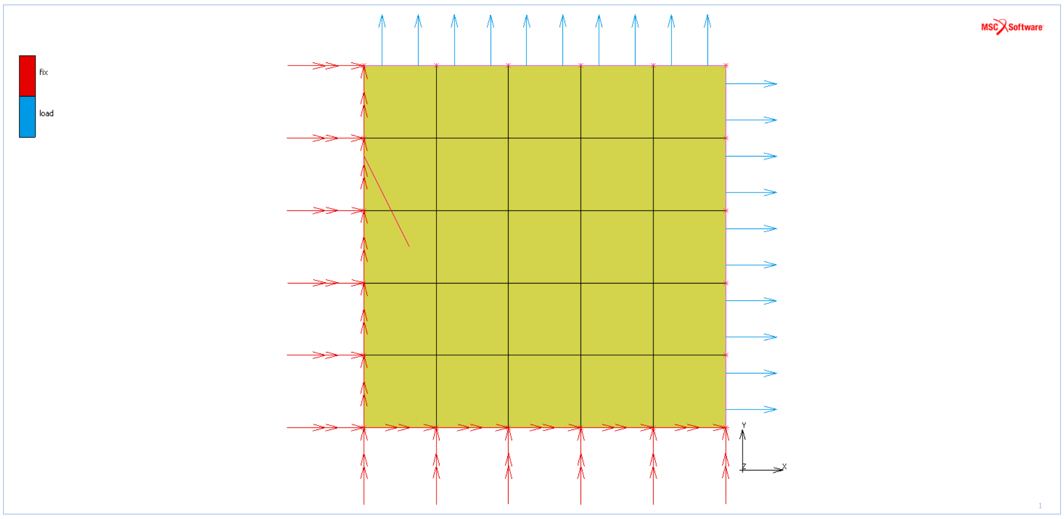

Figure 4.1: Boundary conditions applied to grid of elements

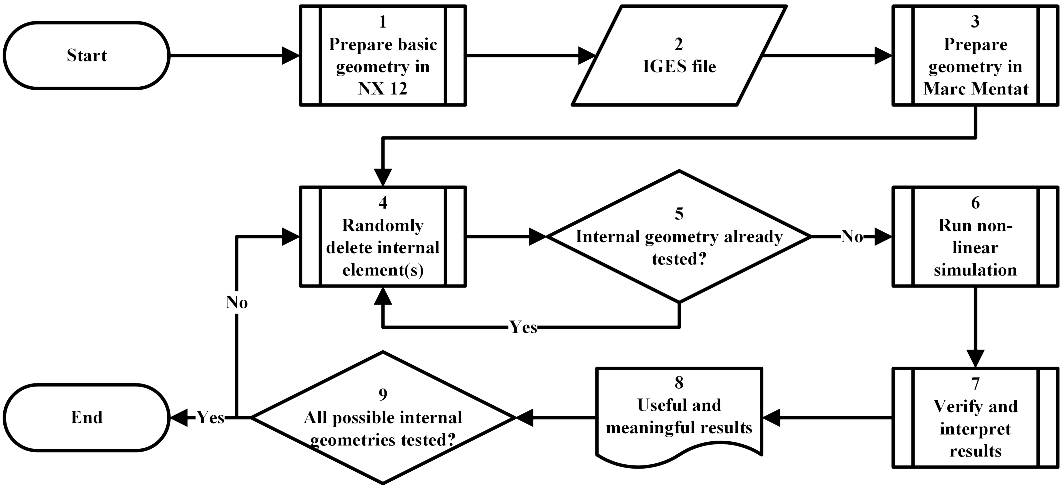

Figure 4.2: Main pipeline flowchart

Figures 4.3 through 4.7 display the flowcharts elaborating on parts of the pipeline. Tables detailing the exact commands and inputs used in Marc follow the relevant figures.

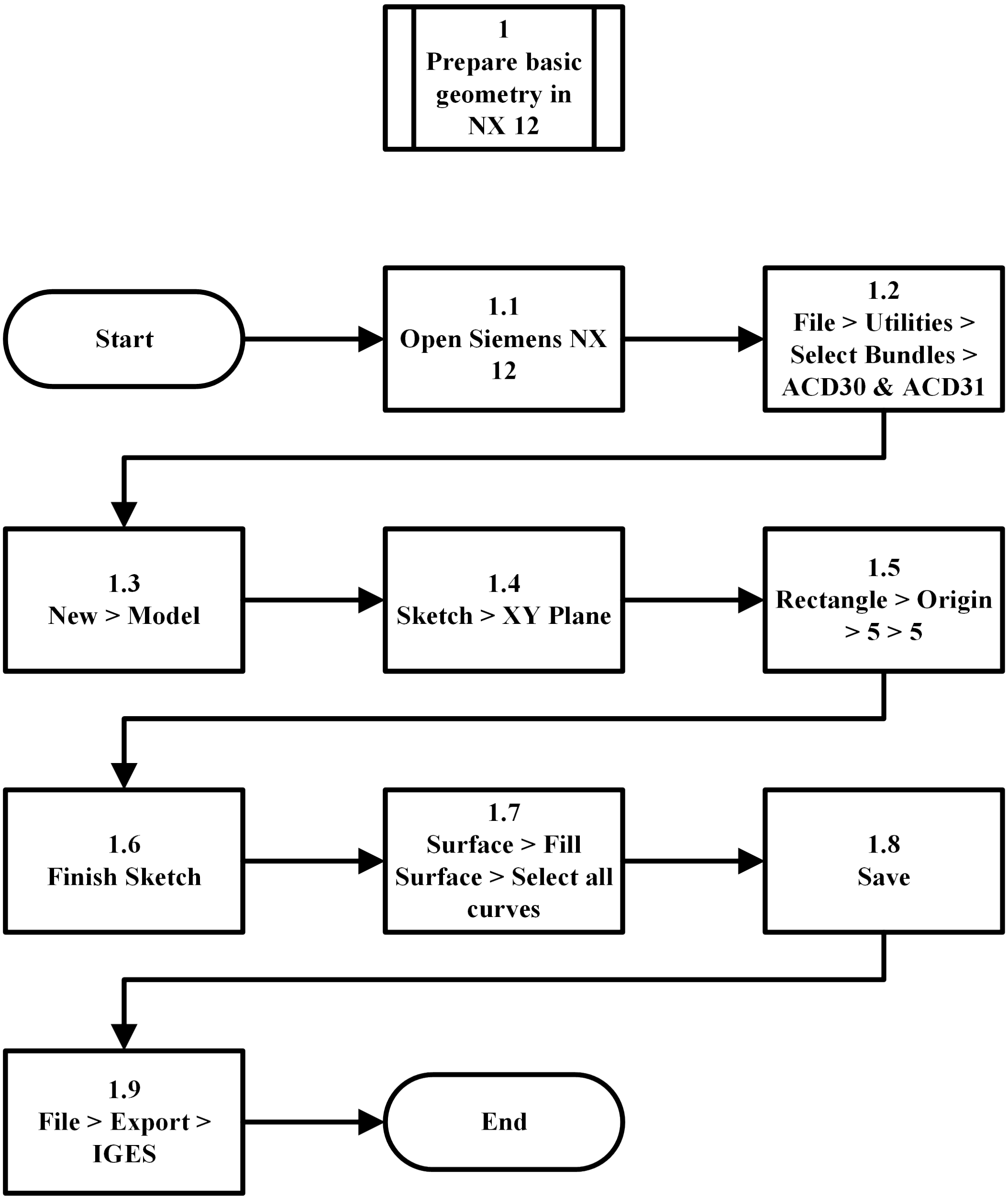

Figure 4.3: Preparation of basic geometry in NX 12 flowchart

Table 4.1: Pre-Automesh (3.6)

Ribbon	Tab	Command	Input	Target Length	Apply Curve Divisions
Geometry & Mesh	Pre-Automesh	Curve Divisions	Target Length	1	1 2 3 4

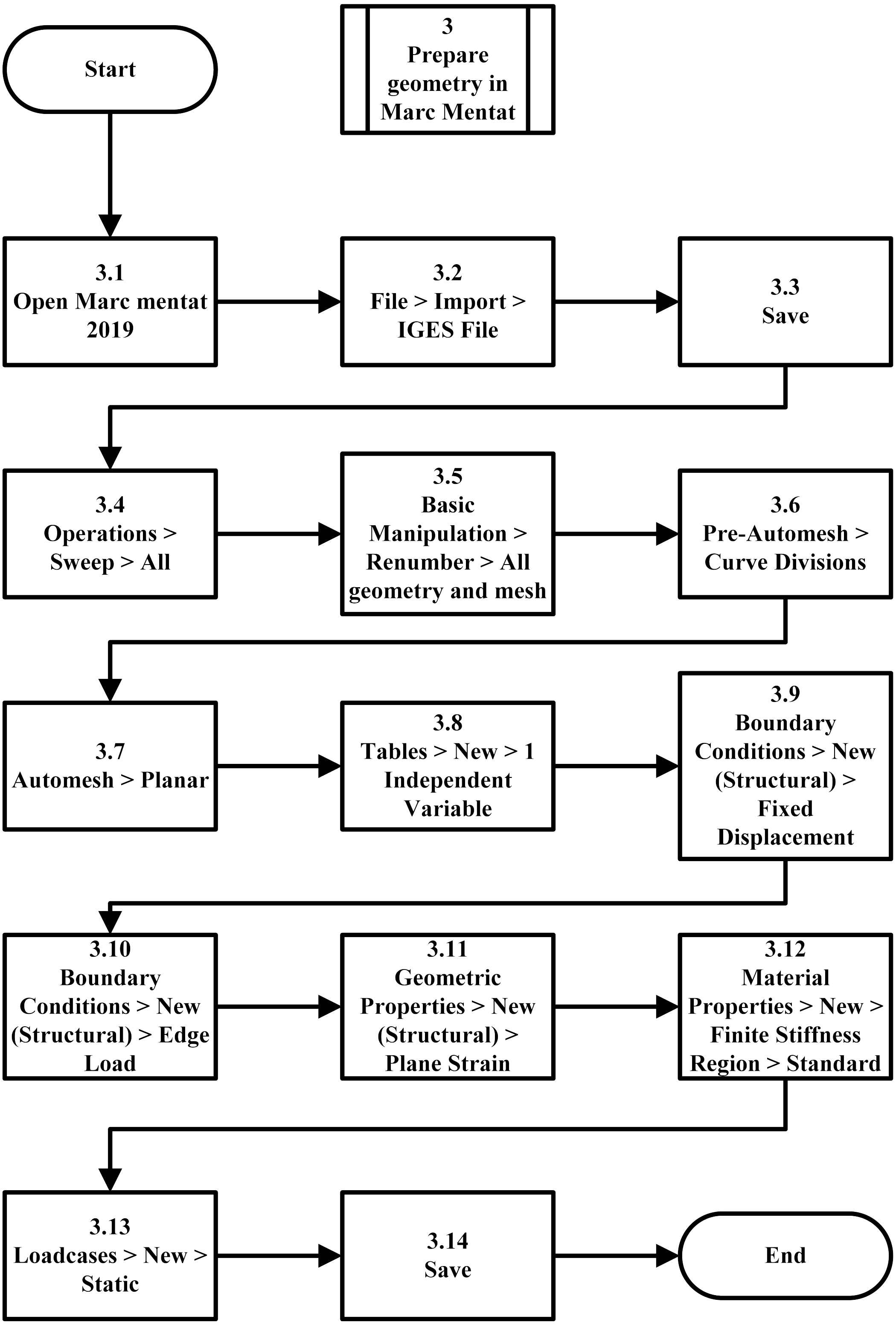

Figure 4.4: Preparation of geometry in Marc Mentat flowchart

Table 4.2: Automesh (3.7)

Ribbon	Tab	Command	Quadrilaterals (Adv Frnt)	Enter curve list
Geometry & Mesh	Automesh	Planar	Quad Mesh!	1 2 3 4

Table 4.3: Tables (3.8)

Ribbon	Tab	Command	Name	Independent Variable V1		Function Value F
				Type	Steps	Steps
Tables & Coord. Syst.	Tables	New >1 Independent Variable	ramp	time	1	1

Table 4.4: Boundary Conditions - Displacement (3.9)

Ribbon	Tab	Command	Name	Displacement X	Displacement Y	Displacement Z	Rotation X	Rotation Y	Rotation Z	Add Nodes
Boundary Conditions	Boundary Conditions	New (Structural) >Fixed Displacement	fix	0	0	0	0	0	0	1 2 3 4 5 6 16 17 18 19 20

Table 4.5: Boundary Conditions - Load (3.10)

Ribbon	Tab	Command	Name	Pressure	Table	Add Curves
Boundary Conditions	Boundary Conditions	New (Structural) >Edge Load	load	-10	ramp	2:0 3:0

Table 4.6: Geometric Properties (3.11)

Ribbon	Tab	Command	Name	Add Elements
Geometric Properties	Geometric Properties	New (Structural) >Plane Strain	straingeom	all

Table 4.7: Material Properties (3.12)

Ribbon	Tab	Command	Name	Type	C10	C01	Add Elements
Material Properties	Material Properties	New >Finite Stiffness Region >Standard	rubber	Mooney	20.3	5.8	all

Table 4.8: Loadcases (3.13)

Ribbon	Tab	Command	Name	# Steps
Loadcases	Loadcases	New >Static	pressureload	100

Table 4.9: Element removal (4.1)

Ribbon	Tab	Command	Rem Elements
Geometry & Mesh	Basic Manipulation	Geometry & Mesh	7 8 9 12 13 14 17 18 19

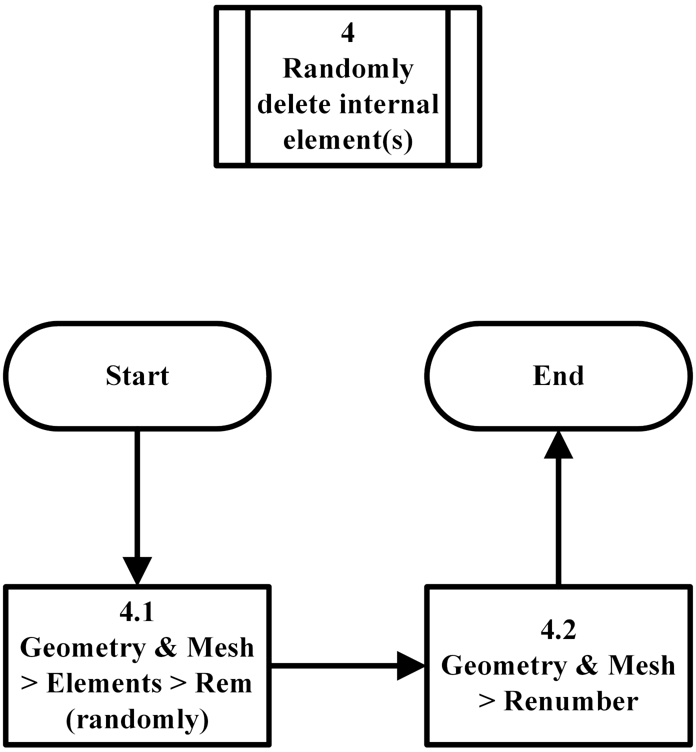

Figure 4.5: Random deletion of internal element(s) flowchart

Table 4.10: Renumber elements (4.2)

Ribbon	Tab	Command	Command
Geometry & Mesh	Basic Manipulation	Renumber	All Geometry & Mesh

Table 4.11: Jobs (6.2)

Ribbon	Tab	Command	Name	Selected	Analysis Options		Job Results	
							Available Element Scalars	
Jobs	Jobs	New >Structural	job_<deleted_element(s)>	pressureload	Large Strain	Follower Force	Equivalent Von Mises Stress	Total Strain Energy

Table 4.12: Model Plot (7.1)

Ribbon	Command	Style	Scalar Plot	
			Style	Scalar
Results	Model Plot	Deformed & Original	Contour Bands	Total Strain Energy

Table 4.13: Movies (7.2)

Ribbon	Command	Base File Name	Command
Results	Movies	baseelement_<deleted_element(s)>	Make GIF Movie

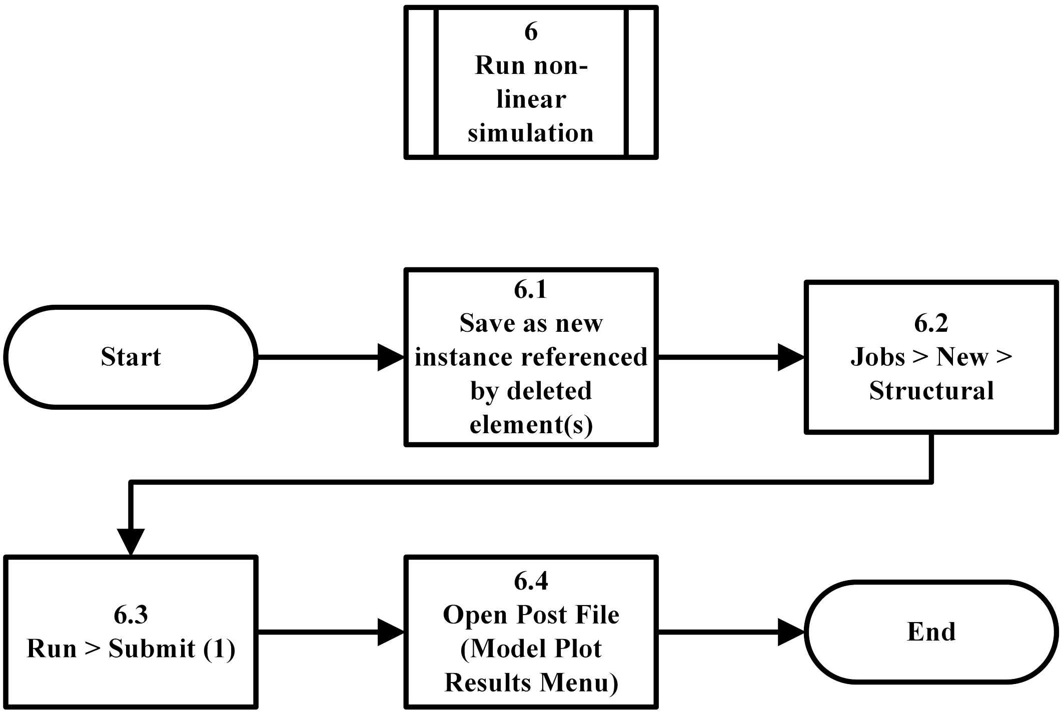

Figure 4.6: Non-linear simulation flowchart

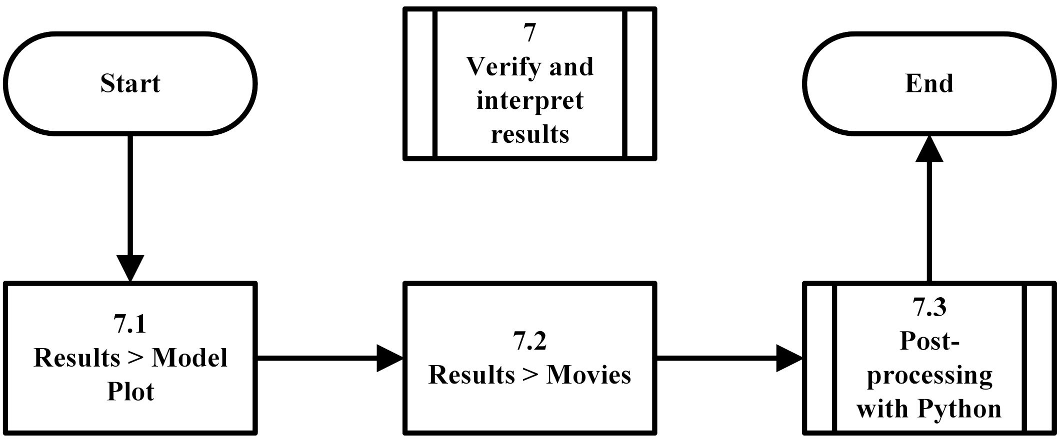

Figure 4.7: Verification and interpretation of results flowchart

Chapter 5

To-Do List

5.1 Document

5.1.1 Introduction

Define aim and objectives properly
-implement generative design process to construct basic elements
-implement generative design process to construct soft bodies from basic elements
-use generative design process to design soft actuator meeting some goal
-compare results to previous work
Define scope and assumptions properly
-hyper-elastic non-linear materials being inflated
-two dimensions

5.1.2 Literature Review

-define L-systems appropriately
-discuss and define CPPNs
-add illustrative diagrams where necessary

5.1.3 Material Testing

List other materials and given properties if applicable
-Smooth Sil 950
-Ecoflex 0030
Describe testing process in detail
-specimen preparation
–wear nitrile gloves
–sanitise workspace
–mix materials in 1:1 ratio
–mix until no streaks

–degas
–pour into tensile specimen and compression specimen mould
–degas
–even out surface
–leave to set for 4 hours
-specimen testing
–describe ISO standards appropriately
–use Instron machine
–100 kN load cell
–clamp grips vs roller grips
–long travel extensometer vs DIC

5.1.4 Software

Discuss coding approach in more detail
Discuss analysis of results in more detail
Refine layout and diagram quality
Eliminate unnecessary diagrams/translate to writing

Bibliography

[1] Mold Star 15, 16 and 30.
Available at: http://www.smooth-on.com/tb/files/MOLD{_}STAR{_}15{_}16{_}30{_}TB.pdf

[2] Shea, K., Aish, R. and Gourtovaia, M.: Towards integrated performance-driven generative design tools. *Automation in Construction*, vol. 14, no. 2 SPEC. ISS., pp. 253–264, 2005. ISSN 09265805.

[3] Whitesides, G.M.: Soft Robotics. *Angewandte Chemie - International Edition*, vol. 57, no. 16, pp. 4258–4273, 2018. ISSN 15213773.

[4] Boyraz, P., Runge, G. and Raatz, A.: An Overview of Novel Actuators for Soft Robotics. *Actuators*, vol. 7, no. 3, p. 48, 2018. ISSN 2076-0825.

[5] Ellis, D.R.: *Generative Design Procedure for Embedding Complex Behaviour in Pneumatic Soft Robots*. Ph.D. thesis, Stellenbosch University, 2020.

[6] Brose, P.: *Compositional Model-Based Design: A Generative Approach To The Conceptual Design Of Physical Systems*. Ph.D. thesis, University Of Southern California, 1993.

[7] Sekhar, P. and Uwizeye, V.: Review of sensor and actuator mechanisms for bioMEMS. *MEMS for Biomedical Applications*, pp. 46–77, jan 2012.
Available at: https://www.sciencedirect.com/science/article/pii/B978085709129150002X

[8] Villoslada, A., Flores, A., Copaci, D., Blanco, D. and Moreno, L.: High-displacement flexible Shape Memory Alloy actuator for soft wearable robots. *Robotics and Autonomous Systems*, vol. 73, no. October 2017, pp. 91–101, 2015. ISSN 09218890.

[9] Behl, M. and Lendlein, A.: Shape-memory polymers. *Materials Today*, vol. 10, no. 4, pp. 20–28, 2007. ISSN 13697021.
Available at: http://dx.doi.org/10.1016/S1369-7021(07)70047-0

[10] Rodriguez, J.N., Zhu, C., Duoss, E.B., Wilson, T.S., Spadaccini, C.M. and Lewicki, J.P.: Shape-morphing composites with designed micro-architectures. *Scientific Reports*, vol. 6, no. June, pp. 1–10, 2016. ISSN 20452322.
Available at: www.nature.com/scientificreports/http://dx.doi.org/10.1038/srep27933

[11] Mutlu, R., Alici, G., Xiang, X. and Li, W.: Electro-mechanical modelling and identification of electroactive polymer actuators as smart robotic manipulators. *Mechatronics*, vol. 24, no. 3, pp. 241–251, 2014. ISSN 09574158.

[12] Do, T.N., Phan, H., Nguyen, T.-Q.Q. and Visell, Y.: Miniature Soft Electromagnetic Actuators for Robotic Applications. *Advanced Functional Materials*, vol. 28, no. 18, p. 1870116, 2018. ISSN 16163028.

[13] Shepherd, R.F., Ilievski, F., Choi, W., Morin, S.A., Stokes, A.A., Mazzeo, A.D., Chen, X., Wang, M. and Whitesides, G.M.: Multigait soft robot. *Proceedings of the National Academy of Sciences of the United States of America*, vol. 108, no. 51, pp. 20400–20403, 2011. ISSN 00278424.

[14] Onal, C.D., Chen, X., Whitesides, G.M. and Rus, D.: Soft mobile robots with on-board chemical pressure generation. *Springer Tracts in Advanced Robotics*, vol. 100, pp. 525–540, 2017. ISSN 1610742X.

[15] Cook, R.D., Malkus, D.S., Plesha, M.E. and Witt, R.J.: *Concepts and Applications of Finite Element Analysis.* 4th edn. Wiley, Madison, 2002. ISBN 978-0-471-35605-9.

[16] Cheney, N., MacCurdy, R., Clune, J. and Lipson, H.: Unshackling evolution: Evolving soft robots with multiple materials and a powerful generative encoding. *GECCO 2013 - Proceedings of the 2013 Genetic and Evolutionary Computation Conference*, pp. 167–174, 2013.

[17] Cheney, N., Bongard, J. and Lipson, H.: Evolving Soft Robots in Tight Spaces. pp. 935–942, 2015.

[18] LS-DYNA | Livermore Software Technology Corp.
Available at: `https://www.lstc.com/products/ls-dyna`

[19] Announcing the Next Generation Design Platform: NX 12 | NX Design.
Available at: `https://blogs.sw.siemens.com/nx-design/announcing-the-next-generation-design-platform-nx-12/`

[20] MSC.Marc Mentat Datasheet. 2003.

[21] Sims, K.: Evolving virtual creatures. *ACM*, pp. 15–22, 1994.
Available at: `http://www.karlsims.com/papers/siggraph94.pdf`

[22] Hornby, G.S. and Pollack, J.B.: The advantages of generative grammatical encodings for physical design. *Proceedings of the IEEE Conference on Evolutionary Computation, ICEC*, vol. 1, no. December, pp. 600–607, 2001.

[23] Prusinkiewicz, P., Lindenmayer, A., Hanan, J.S., Fracchia, F.D., Fowler, D., de Boer, M.J.M. and Mercer, L.: *The algorithmic beauty of plants.* 1. 2004.

[24] Groenwold, A.A., Stander, N. and Snyman, J.A.: A regional genetic algorithm for the discrete optimal design of truss structures. *International Journal for Numerical Methods in Engineering*, vol. 44, no. 6, pp. 749–766, 1999. ISSN 00295981.

[25] Sims, K.: Evolving 3D Morphology and Behavior by Competition. *Artificial Life IV Proceedings*, pp. 28—-39, 1994.

[26] Kolodziej, J.: Modeling hierarchical genetic strategy as a Lindenmayer system. *Proceedings. International Conference on Parallel Computing in Electrical Engineering*, pp. 409–414, 2002.
Available at: `http://ieeexplore.ieee.org/document/1115312/`

[27] Zakinthinos, A. and Lee, E.S.: Composing secure systems that have emergent properties. *Proceedings of the Computer Security Foundations Workshop*, pp. 117–122, 1998. ISSN 10636900.

[28] Aiguier, M., Gall, P.L. and Mabrouki, M.: Emergent properties in reactive systems. *Neonatal, Paediatric and Child Health Nursing*, , no. January 2008, pp. 273–280, 2008. ISSN 14416638.

[29] Damper, R.I.: Emergence and levels of abstraction. *International Journal of Systems Science*, vol. 31, no. 7, pp. 811–818, 2000. ISSN 14645319.

[30] Rieffel, J., Knox, D., Smith, S. and Trimmer, B.: Growing and Evolving Soft Robots. *Artificial Life*, vol. 20, no. 1, pp. 143–162, 2013. ISSN 1064-5462.

[31] Hiller, J. and Lipson, H.: Automatic design and manufacture of soft robots. *IEEE Transactions on Robotics*, vol. 28, no. 2, pp. 457–466, 2012. ISSN 15523098.

[32] ISO 37 - Rubber, vulcanized or thermoplastic - Determination of tensile stress-strain properties. 2017.

[33] ISO 7743 - Rubber, vulcanized or thermoplastic - Determination of compression stress-strain properties. 2017.