

# Computational Evolutionary Embryogeny

Thesis by

Or Yogev

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy



California Institute of Technology

Pasadena, California

2009

(Submitted Defended / 12.12.08)

© 2009  
Or Yogev

All Rights Reserved

**Acknowledgments** The research described in this thesis was conducted in the Engineering Design Research Laboratory at the California Institute of Technology. Many people have been contribute to this research. First, I would like to thank my advisor - Professor Erik K. Antonsson, for his guidance and support along this research, his unbiased enthusiastic and interest in this area, all of which led to great discussions and ideas sharing, which have been matured and demonstrated in this work. I would also would like to thank my coadvisor, Professor Andrew Shapiro for his help and support and his great contribution to this work. Other people which have been part of this research and which I would like to thank are: Professor Rob Philips and Professor Chris Adami.

I would also like to gratefully acknowledge the generosity of Professor Swaminathan Krishnan at the California Institute of Technology in making his computing cluster available for our evolutionary embryogeny computations .

# Abstract

Evolution and development (Evo-Devo), are the two main processes which produce all of the different kinds of phenotypes we see in nature. Evolutionary process is responsible for eliminating the genetic information of weak phenotypes through natural selection, and also for exploring novel genotypes through genetic operations; crossover, mutation. The development process is the process of using the set of rules (codons) written in a genome, to turn a single set (zygote) into a mature phenotype. In this thesis, evolutionary and developmental processes are used to evolve the configurations of three-dimensional structures *in silico* to achieve desired performances. Although natural systems utilize the combination of both evolution and development processes to produce remarkable performance and diversity, this approach has not yet been applied extensively to the design of continuous three-dimensional load-supporting structures. Beginning with a single artificial cell containing information analogous to a DNA sequence, a structure is grown according to the rules encoded in the sequence. Each artificial cell in the structure contains the same sequence of growth and development rules, and each artificial cell is an element in a finite element mesh representing the structure of the mature individual. Rule sequences are evolved over many generations through selection and survival of individuals in a population.

Modularity and symmetry are visible in nearly every natural and engineered structure. Understanding of the evolution and expression of symmetry and modularity is emerging from recent biological research. Initial evidence of these attributes is present in the *phenotypes* that are developed from the artificial evolution, although neither characteristic is imposed nor selected for directly.

The computational evolutionary development approach presented here shows promise for synthesizing novel configurations of high-performance systems. The approach may advance system design to a new paradigm, where current design strategies have difficulty producing useful solutions. In addition to a new design approach perse, this model gives us the ability to explore the development process, from the standpoint of complex systems analysis. The phenotypes in our system have been grown under a highly stochastic environment, which serves as a triggered mechanism

for gene expression. Still, evolution was able to find solutions which are robust to these stochastic elements, both at the phenotype level (the phenotype ability to function under the environment) and the growth process itself. In addition we have also explored the effects of symmetric and non-symmetric environment over the topology of the phenotypes; we have found strong evidence that indicates a high correlation between the two. Finally we have also established a tool which enables us to understand the relationship between the environment and the degree of modularity of the phenotype.

# Contents

<b>Abstract</b>	<b>iv</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Definitions . . . . .	1
1.2 Prior Related Work . . . . .	2
<b>2 Approach</b>	<b>5</b>
2.1 Material Properties . . . . .	5
2.2 Evolutionary and Developmental Scheme . . . . .	6
2.3 Cells . . . . .	8
2.4 Development Process . . . . .	8
2.5 Genes/Rules . . . . .	9
2.5.1 Conditionals . . . . .	10
2.5.2 Cell-Type Actions . . . . .	10
2.5.2.1 Cell Division . . . . .	10
2.5.2.2 Cell Differentiation . . . . .	11
2.5.2.3 Cell Death . . . . .	11
2.5.2.4 Cell Adhesion . . . . .	11
2.5.3 Geometric Actions . . . . .	13
2.5.3.1 Growth . . . . .	14
2.5.3.2 Shear . . . . .	16
2.6 Environment . . . . .	18
2.7 Genome Structure . . . . .	19
2.7.1 Syntax Rules . . . . .	20
2.8 Metabolism and Thermodynamics . . . . .	22

2.9	Time Increments . . . . .	23
2.10	Diseases . . . . .	24
2.10.1	Repair Process . . . . .	24
2.11	Maturity . . . . .	26
2.12	Fitness Evaluation . . . . .	27
2.12.1	Fitness Description . . . . .	27
2.12.1.1	Aggregation . . . . .	28
2.13	Example . . . . .	30
2.14	Results . . . . .	32
2.14.1	Phenotype Analysis . . . . .	33
2.14.2	Development Analysis . . . . .	34
2.14.3	Genome Analysis . . . . .	36
2.14.4	Additional Runs . . . . .	39
2.14.5	Constraints . . . . .	40
2.14.6	Robustness . . . . .	40
<b>3</b>	<b>Asynchronous Scheme</b>	<b>43</b>
<b>4</b>	<b>Inhomogeneous Structures under Wind Load</b>	<b>45</b>
4.1	Wind Load . . . . .	45
4.2	Results, Inhomogeneous Structures under Wind Load . . . . .	46
<b>5</b>	<b>Robustness</b>	<b>50</b>
5.1	Results . . . . .	51
5.2	Family of Solutions . . . . .	52
5.3	Phenotype Robustness . . . . .	52
5.4	Growth Robustness . . . . .	53
<b>6</b>	<b>Word Duplication Mechanism, "WDM"</b>	<b>56</b>
6.1	The Effect of "WDM" on the Phenotypes . . . . .	57
6.2	The Relation Between "WDM" and the Landscape of Solutions . . . . .	59
6.3	Genes/Word Regulations - "WDM" and Non-"WDM" . . . . .	60
6.3.1	Words Distribution in Non-"WDM" . . . . .	60
6.3.2	Words Distribution include "WDM" . . . . .	61

6.4	Exploring the Effect of Words on the Phenotype Topology . . . . .	63
6.5	The Evolution of Simple Modules . . . . .	68
6.5.1	The Effect of a Nonsymmetric Environment . . . . .	71
<b>7</b>	<b>SelfRecovery Mechanism</b>	<b>74</b>
<b>8</b>	<b>Appendix</b>	<b>76</b>
8.1	Configuration . . . . .	76
8.2	Finite Element Scheme . . . . .	76
8.2.1	Integration scheme . . . . .	81
8.2.2	Assembly scheme . . . . .	81
<b>9</b>	<b>Cell Adhesion Scheme</b>	<b>83</b>
9.1	Cell Adhesion Scheme in Finite Element Mesh . . . . .	83
9.2	Overview of Existing Mesh Techniques . . . . .	84
9.2.1	Our Approach . . . . .	85
9.2.2	Overview . . . . .	85
9.3	The Adhesion Approach . . . . .	85
9.4	Hexahedral Linear Brick Element . . . . .	89
9.5	The Conjugate Gradient Method . . . . .	93
9.6	Results . . . . .	96
9.7	Application . . . . .	97
9.8	Conclusions . . . . .	101
9.9	Future Work . . . . .	103
	<b>Bibliography</b>	<b>104</b>



# List of Figures

2.1	The computational evolutionary development process. . . . .	7
2.2	The growth and development process. . . . .	8
2.3	The basic structural hexahedral “brick” three dimensional finite element. . . . .	9
2.4	Celldivision operation. . . . .	11
2.5	Adhesion process . . . . .	12
2.6	The four basic geometric operations observed in sub-regions of plants. . . . .	13
2.7	Hexahedral mesh vs grid mesh . . . . .	14
2.8	Face numbering - Local configuration. . . . .	15
2.9	Shear operation. . . . .	17
2.10	The amount of energy $E_c$ available to each cell during growth, as a function of the total number of cells in the growing phenotype. . . . .	23
2.11	The average usage of the repair mechanism. . . . .	25
2.12	The average usage of the repair mechanism with respect to time for one phenotype during the growth and development process. . . . .	25
2.13	A typical development process. . . . .	27
2.14	A representative preference function. . . . .	29
2.15	A simple reference solution. . . . .	31
2.16	Initial setting of the growth process . . . . .	31
2.17	The evolution of fitness values with respect to the number of generations . . . . .	32
2.18	The evolution of fitness values once the phenotype has reached the desired height. . .	33
2.19	Three stages of an early evolved phenotype . . . . .	34
2.20	Two views of a first phenotype (fitness value = 2.11), which is able to support the load, as indicated by the predominantly green color of the cells. . . . .	35
2.21	Two views of a second phenotype (fitness value = 2.07), displaying a double-helix configuration. This phenotype is also able to support the load. . . . .	35

2.22	Six stages in the growth and development of the phenotype shown in Figure 4.2. . . . .	37
2.23	Four different phenotypes, growing from the same genome . . . . .	38
2.24	The distribution of fitness values of phenotypes grown and developed from a single genotype. . . . .	38
2.25	The genome for the phenotype shown in figures 4.2 and 2.22 contains 9 different words. . . . .	39
2.26	Maximum mechanical stress, normalized to the material yield stress ( $\sigma/\sigma_y$ ) vs. angle of the load vector. . . . .	41
2.27	Fitness value vs. angle of the load vector. . . . .	41
2.28	An evolved phenotype which has been generated by additional simulation. . . . .	42
4.1	four different view of an inhomogeneous phenotype . . . . .	47
4.2	four different view of an inhomogeneous phenotype . . . . .	48
4.3	Growth process of the phenotype in Figure 4.1. . . . .	49
5.1	Three different phenotypes from three different stochastic environment experiments (SEE). . . . .	51
5.2	Phenotype evolved with an unvarying load (fixed environment). . . . .	53
5.3	A comparison between two evolutionary process . . . . .	54
5.4	A comparison between the standard deviation of two evolutionary process . . . . .	54
5.5	Fitness distribution of two genomes. . . . .	55
6.1	”Best” phenotype from both experiments . . . . .	58
6.2	A convergence comparison of both experiments. . . . .	58
6.3	The evolution of new words during evolution. . . . .	59
6.4	Non-”WDM”, the distribution of words expression. . . . .	62
6.5	The evolution of new words during evolution. . . . .	62
6.6	”WDM”,the distribution of words expression. . . . .	64
6.7	”WDM”, repeated times in the genome vs number of executions . . . . .	66
6.8	The effect of word elimination on the fitness distribution. . . . .	68
6.9	Phenotype resulting from word elimination . . . . .	69
6.10	Six stages in the growth and development of the ’tripod’ phenotype . . . . .	70
6.11	The first primitive phenotype, contains word 9. . . . .	71
6.12	The effect of the environment over the topology of the phenotype . . . . .	73

7.1	Six stages during the growth of a selfrecovered phenotype. . . . .	75
8.1	The local-global mapping. . . . .	77
8.2	The basic structural hexahedral “brick” 3-D finite element. . . . .	78
9.1	Adhesion process . . . . .	86
9.2	Pseudocode outline of the algorithm. . . . .	97
9.3	Example 1 . . . . .	98
9.4	Example 2 . . . . .	99
9.5	Example 3 . . . . .	100

## Nomenclature

$V_0$  initial volume of the cell

$h$  height of a selected cell face

$A_p$  area of a selected cell face

$R, S, T$  coordinates of a point inside a cell in local coordinates

$r, s, t$  coordinates of a point inside a cell in reference coordinates

$\mathbf{F}$  deformation tensor

$\mathbf{r}$  a vector form of  $r, s, t$

$\mathbf{R}$  a vector form of  $R, S, T$

$\alpha$  a scalar which represents change in volume

$v$  cell volume after applying geometrical operation

$V$  cell volume before applying geometrical operation

$\mathbf{J}$  jacobian

$N_i$  interpolation function

$X_i$  the X node i coordinates on the reference configuration

$Y_i$  the Y node i coordinates on the reference configuration

$Z_i$  the Z node i coordinates on the reference configuration

**dR, dS, dT** infinitesimal material vectors in the X, Y and Z direction respectively.

**a** directional vector originating in the reference configuration

$\kappa_1$  first coefficient for the shearing operation

$\kappa_2$  second coefficient for the shearing operation

$\alpha_1, \alpha_2, \alpha_3$  expansion coefficients

$\sigma$  Cauchy-Green tensor

$s$  morphogen diffusion rate

$d$  morphogen diffusion constant

$E_c$  energy consumption

$B_c$  metabolic rate

$S$  total volume of the phenotype

$N_c$  number of cells comprising the phenotype

$k$  node number

$x_m^{(k)}$  vector coordinate of the  $k^{\text{th}}$  node

$m$  dummy index

$\alpha_m$  deformation measure of a node, used as part of the repair process

$x_m$  node coordinates in the present configuration, use as part of the repair process

$f_0$  the total deformation level of the phenotype

$g_m$  degree of deformation of the cell

$e_{m,k}$  vector notation, used in the process of computing  $f_0$

$x_{m,k}$  node coordinates, used in the process of computing  $f_0$

$x$  node coordinates

$J_m$  Vector, used in the process of computing  $f_0$

$\mu_i$  preference function for the  $i^{\text{th}}$  variable

$S_i$  the  $i^{\text{th}}$  performance variable

$a, b$  preference function parameters

$\alpha$  slope parameter for preference functions

$\omega_i$  importance weight for aggregation

$\mathcal{P}_s$  aggregation function

$s$  degree of compensation for aggregation

keywords: evolution, development, embryogeny, genetic algorithm, morphogen, phenotype, genome, finite element, structure, design synthesis, modularity.

# Chapter 1

## Introduction

Natural evolution has produced systems of fantastic complexity, robustness and adaptability. Recent research has shown that it is the combination of both evolution and development processes that has produced these remarkable results [1, 2]. Evolution does not act directly on the configurations of adult phenotypes, rather it successively alters and revises the *rules* that guide the growth of a zygote into an embryo and its further development into an adult.

The computational evolutionary development approach presented here is based on an artificial evolution using indirect encoding of growth and development rules. The approach was tested on a classical structural engineering problem. The results demonstrate that artificial evolution and embryogenesis can synthesize phenotypes with novel configurations that exhibit modularity, and which meet performance goals.

The objective of this research is not simply to evolve structures that meet the desired performance requirements, but also to explore complex design environments (including functionally-graded, tailored composite materials) and to develop a fundamental understanding of the origin and development of *modularity* in design, and the design rules that give rise to modular design configurations.

### 1.1 Definitions

**Phenotypes** are

*the set of observable characteristics of an individual resulting from the interaction of its genotype with the environment”.*

**Rules**, in this context as in nature, are encoded in the genome of each individual. Genes regulate

the production of proteins using transcription factors, and hence the growth and development of the organism. The genome contains the set of instructions for the development process while the environment provides inputs that regulate the instructions [3].

Natural evolutionary processes refine the sets of rules, in the form of genes, which result in adult forms. Natural selection acts upon the phenotypes, thus rewarding sets of rules that produce fit individuals. The *indirect* character of the encoding of genetic information in natural systems, and the inter-relationship between evolution of rules and the growth and development of adult forms, has been responsible for the diversity, complexity, modularity, robustness, and adaptability in the natural world [4, 5].

**Embryogeny** or **Embryogenesis** is the process of growth by which a genotype develops into a phenotype, and is central to the emerging understanding of the relationship between evolution and development. “It should be noted that the correct term is *embryogeny*, which refers to the process, rather than the oft-misused term *embryology*, which refers to the science of studying embryos and embryogenies.” [6]

## 1.2 Prior Related Work

Artificial evolution, in the form of Genetic Algorithms (GA), has been used in a wide variety of application areas [10, 11, 12, 13]. The goal of this prior work has been to improve one or more features of a problem, and GA's have been applied in a wide variety of application areas. The key element of GA's, first stated by Holland [14], is *implicit parallelism*. The idea is that the GA scheme of actions; selection, crossover and mutation performed on  $N$  individuals, implicitly searches for an optimum in  $N^3$  space. This result is powerful, since it enables the rapid exploration of large solution spaces.

The majority of optimization problems that make use of the genetic algorithm approach have employed *direct* encoding. In direct encoding, there is a one-to-one relationship between the genetic information in an individual and the configuration of the individual. Most commonly, the genetic information contains a description of the individual, in contrast with indirect encoding, where the genetic information contains a set of rules that, when executed (and perhaps influenced by various environmental factors), guide the growth and development of a single cell into an adult.

Genetic algorithms have been previously applied as an optimization method in structural evo-

lution. Traditional structural evolutionary methods generally start by generating a fixed mesh grid (like a chess board) with a predefined volume and constraints, such as external forces and boundary conditions [15]. Every cell in the grid can have one of two states, either material is present or absent. The genetic information in this approach contains information indicating the material state in each cell of the grid. Once the initial configuration and the boundary conditions are defined (loads, constraints, *etc.*), an evolutionary process searches for the configuration exhibiting the best performance. This is an evolution using a *direct* encoding, with no embryogeny, and thus there is a one-to-one mapping between the genotype and phenotype, resulting in an optimization problem with a large but finite number of possible states [16]. This method produces structures that reflect the underlying shape of the grid [10] and is not able to create continuous or smooth structures. These grid-based building-block structures are not only unrealistic from an engineering perspective, but from a mathematical point of view they search in a limited solution space, resulting in local optima.

Indirect encoding, and the study of artificial embryogeny, has been proposed previously [6, 17, 18, 19]. In most examples the goal has been to grow and evolve a predefined target shape on a predefined grid starting from a single cell, using simple rules such as cell division and protein diffusion. Early work in this area has demonstrated the ability of indirect encoding to produce modular phenotypes in graphs and patterns [20, 21, 22].

The study of computational embryogeny has previously been conducted using a variety of approaches. Autonomous agents is one of the fields where this technique first made an impact, specifically in the field of neural networks. The notion of utilizing a collection of simple basic elements (autonomous agents) to solve a complex problem has proven to be highly efficient [23]. Vaario [24] has built an adaptive neural network that grows and adapts with respect to an external environment such that every neuron is an autonomous agent. He used an L-system concept to build a stimulus tracking system [25]. In his system, he defined agents as single neurons with basic rules for growth and development regulated by the environment. Once these rules were evolved, artificial organisms were created that were capable of adapting themselves to changes in the environment. Vaario was also one of the pioneers who understood the importance of the role of the environment in solving complex problems artificially [26]. Kitano [27] first used a context-free grammar, in the form of the L-system, to represent a neural network. He showed that this indirect representation speeds up the rate of convergence gradually. Following this work, he developed an artificial growth model with basic metabolic rules [28]. He showed that after hundreds of generations, these rules can produce complex networks that are selforganized. Following Kitano's work, Eggenberger introduced an ar-



tificial evolutionary and development system [15]. In his system cells were created with a set of genes that were to be evolved into a desired 3D shape in response to the concentration field of a morphogen [29, 30, 31, 32, 18, 20].

The work presented here builds on these ideas of growth and development, and demonstrates the unconstrained evolution of rules that produce structures comprising multiple materials, where fitness is determined only by structural performance.

## Chapter 2

# Approach

In the work reported here, an artificial embryogeny has been created for structures. The two critical foundational elements of this work are the selection of the artificial cell (the basic structural element), a collection of which constitute each individual, and the artificial genes (the rules) which are evolved into the genetic information for each individual.

The genetic information of an individual is shared by all of its cells. Each individual cell executes its rules until a mature structure is formed. Once maturity is reached, an evaluation scheme determines the fitness (performance) of the structure. Evolutionary operations (selection, crossover, and mutation) alter and refine the genetic information in a population of individuals over multiple generations. The results are structures that meet the desired performance goals.

### 2.1 Material Properties

One of the main advantages of the approach presented here is the ability to evolve inhomogeneous structures with a high degree of internal complexity. It has been observed that the material properties of biological structures are unique and complex. A wide range of stiffnesses and strength are exhibited, and in many cases a combination of a high degree of compliance with high strength produces robust structures that are difficult or impossible to replicate with the engineered materials of today. One contributing factor to this difference is inhomogeneity: the material properties of many natural structures change from location to location.

Engineered materials with similar degrees of inhomogeneity are just now beginning to be available, through techniques such as Shape Deposition Manufacturing (SDM) [33]. SDM creates structures by adding and binding materials with different mechanical properties droplet by droplet. Design approaches to best take advantage of these new tailorable inhomogeneous engineered materials

have not yet been developed.

## 2.2 Evolutionary and Developmental Scheme

The evolutionary and developmental scheme used here is derived from a genetic algorithm, and is illustrated in figure 2.1. The growth and development process in particular is shown in figure 2.2.

The algorithm is initialized with sets of randomly generated genomes. Development begins with a single artificial cell. This cell is placed on (and attached to) a ground plane. A load vector, to be supported by the evolved and developed structure, is positioned above the initial cell. The origin of the load vector radiates a signal, analogous to a morphogen. The goal is for individuals comprising one or more cells to grow to reach the height of the morphogen and to be able to support the load.

One individual is grown from each genome, by executing the rules in the genome. Once each individual reaches maturity, its fitness is evaluated by means of a finite element analysis and additional parameters. The selection process is based on the *roulette wheel* method, where the fitness values of each individual are used to select parents to produce offspring, where a higher fitness value results in a higher probability of being selected for reproduction.

Once two parents have been selected, they produce offspring through a crossover process. In this process, the genome from each parent is cut at a randomly selected word boundary, where a word is a sequence of rules. One portion of the gene string from each parent is joined together producing a child. The remaining gene strings from each parent are joined together producing a second child.

Similar to evolution in nature, the genome is also subject to random mutation. The mutation process can erase an entire word and replace it with another, or replace a single rule within a word.

These three steps: selection, crossover and mutation are repeated, and each repetition is defined as one generation.

Due to the difficulty to evolving structures that grow to the desired height and support the load, a staged evolutionary scheme is used. This approach is similar to incremental problem complexity [34], where task complexity grows with the developing ability of evolved phenotypes. In the application of this scheme here, phenotypes are gradually challenged with increasingly difficult tasks as the evolution proceeds, *i.e.*, the height of both the morphogen and the load to be supported are increased.

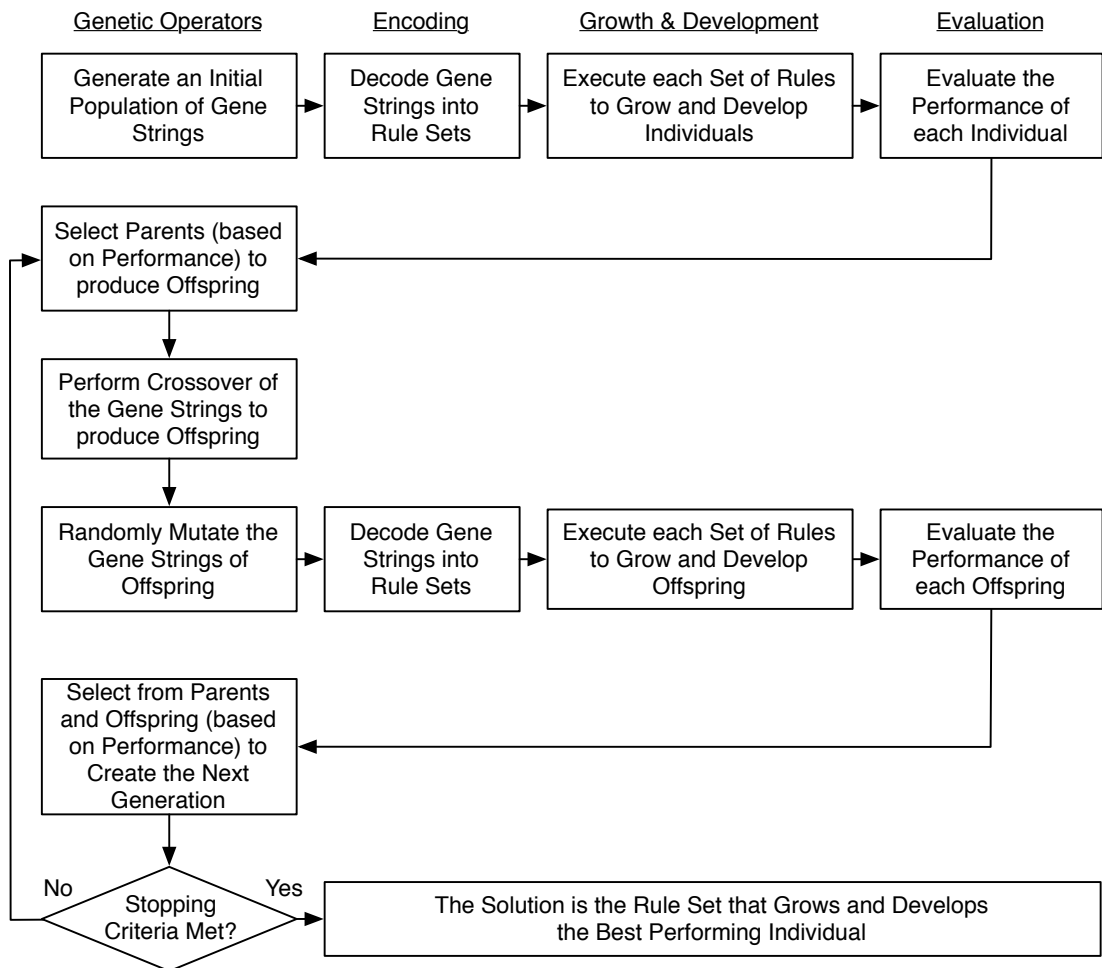


Figure 2.1: The computational evolutionary development process.

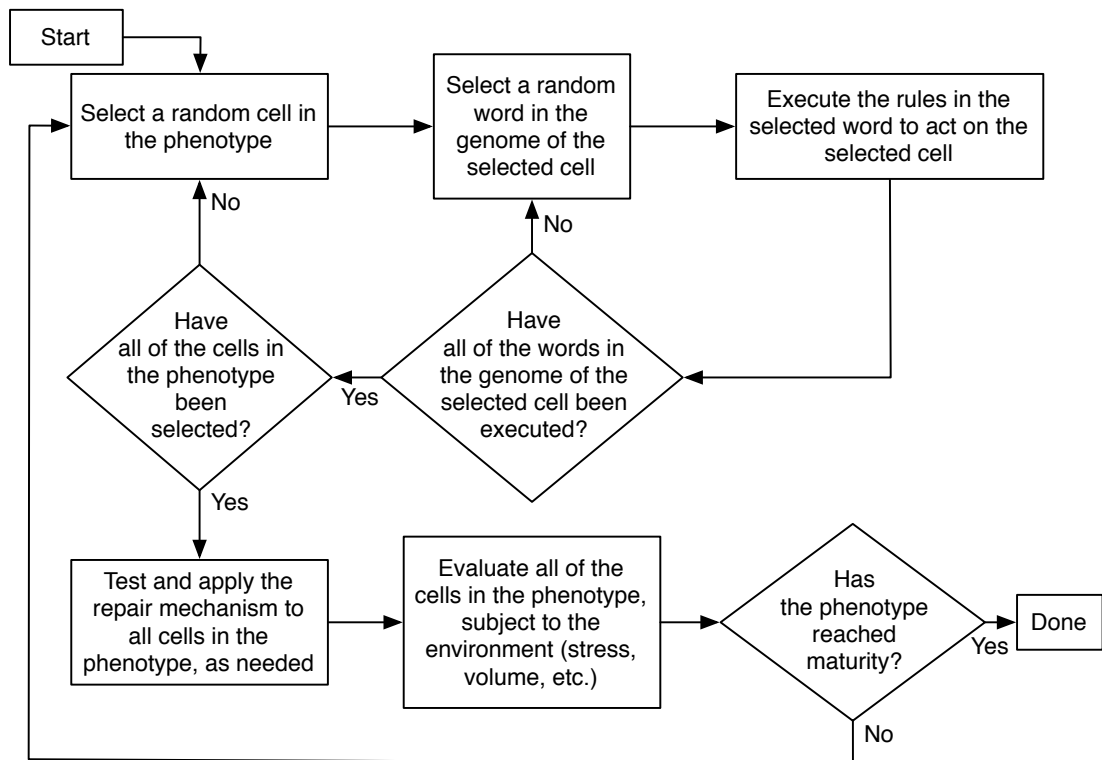


Figure 2.2: The growth and development process.

## 2.3 Cells

The model begins with a structural element that is analogous to a cell in biology. Cells in nature act as building blocks for organisms [74]. The basic structural element in the approach presented here is an extended 3-D triclinic hexahedral finite element “brick”, illustrated in Figure 2.3. Each cell-like finite element contains an identical copy of the genetic information of the individual. A collection of cells forms a finite element mesh, which defines the phenotype.

## 2.4 Development Process

In the development process a single cell grows and develops into a mature adult phenotype. The process starts with a single cell placed on the ground subjected to a gravity field. The cell is initially made of steel. A morphogen radiating from the position of the load is placed at a fixed point in space, as illustrated in Figure 2.16. The morphogen diffusion and the gravity field create an environment that is sufficient to regulate the rules in the genome. The execution of rules modifies existing cells and produces new cells that contain an identical copy of the genome. The new cells

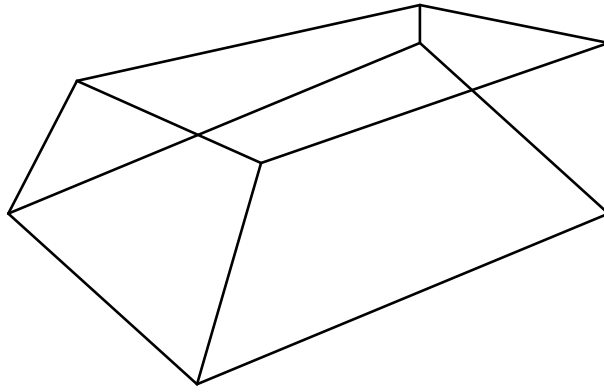


Figure 2.3: The basic structural hexahedral “brick” three dimensional finite element.

are also subjected to the global environment (gravity and the morphogen concentration) and their local environment (stresses and signals from nearby cells). Because the environment of each cell is different, the regulation mechanisms may cause different rules in different cells to be executed. This repeated process of cell production and rule/gene regulation creates a phenotype which will eventually grow and reach the load morphogen. Once a phenotype has reached the load, the load is applied to the top cells that coincide with the morphogen. This load generates a mechanical stress distribution along the phenotype which will alter the local environment on the cells, and may cause the rule/gene regulation mechanisms to alter the further growth and development of the cells. The process of evaluating the mechanical stresses on the cells at each time step is performed by solving a finite element scheme.

## 2.5 Genes/Rules

Rules are the basic instructions encoded inside the genome. Here, rules dictate the growth and development processes of each cell of the phenotype. Mimicking nature, the basic structure of genetic information is an *if-conditional then-action* rule.

Two main principles guided the creation of the rule elements (conditionals and actions) to form genes. The first principle was to create conditional rules that respond only to the local environment of the cell. The second principle was to choose action rules that will not put any constraint on the topology of the phenotype and can generally develop any 3D shape.

In addition to these two principles, rule elements were added that are observed in nature, such as cell differentiation and veto rules. All of these rules play a significant role in the development of

phenotypes.

The rules used here fall into three groups: geometric actions and cell-type actions, and conditionals in the form of veto tests. Geometric rules represent instructions which have an effect on the shape of the cell. Cell-type rules alter the type of the cell, and therefore mimic basic process observed in cell biology. Veto tests affect other instructions at the genome level, and will be discussed first.

### 2.5.1 Conditionals

Conditional artificial rules are “veto” or “suppression” rules. These rules affect other rules only at the genome level, by turning actions off or on according to whether the conditional test is satisfied or not. Veto genes that switch regulatory mechanisms on or off have been observed in biology [1]. For example tumor suppression genes that turn off other genes that produce tumor cells [36].

### 2.5.2 Cell-Type Actions

Cell-type rules are actions which mimic basic operations in cell biology. Four basic operations are used here: cell division, cell differentiation, cell death, and cell adhesion. Each of these operations was modeled as a single rule.

#### 2.5.2.1 Cell Division

The cell division rule is responsible for creating new mass and thus generates more building blocks (new cells) for the construction of the phenotype. Cells divide with respect to a given vector in the reference configuration of the cell. The vector specifies a face of the new cell. The selection of the face uses the inverse isoparametric mapping [37], and is performed by translating the vector into the cell center point and then determining the face which intersects the translated vector. Once the face has been determined, a new triclinic hexahedral cell is created perpendicular to the selected face, such that the total volume of both cells equals the volume of the parent cell prior to division, as shown in figure 2.4.  $V_0$  is defined as the initial volume of the cell prior to division. An isotropic shrink operation is applied with parameter  $\alpha = \sqrt[3]{\frac{1}{2}}$ . This operation will shrink the cell into half of its original volume. The new cell is an extension by height  $h$  of the selected parent face. The face area is identified by  $A_p$ , and then the extension height of the new cell is determined, shown in equation ( 2.1).

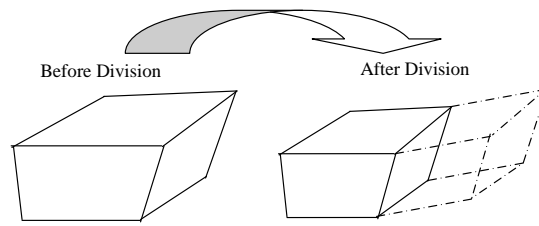


Figure 2.4: Celldivision operation.

$$h = \frac{V_0}{2A_p} \quad (2.1)$$

A cell will attempt to divide, prioritized according to the parameters governing the rules in its genome.

### 2.5.2.2 Cell Differentiation

Two different kinds of cells are modeled, one made of steel and the other made of aluminum. The cell differentiation rule simply alters the material properties of a cell from one type to the other.

### 2.5.2.3 Cell Death

Cell death is a self extermination mechanism, which kills the cell and removes it from the phenotype, and from further consideration in the embryogeny computations.

### 2.5.2.4 Cell Adhesion

During growth, two cells may intersect, causing them to adhere. In nature, this fundamental process creates the integrated structures forming the configuration of the organism. Studies have shown that the speed of the adhesion process is short [38]. Adhesion of nearby cells is modeled in a manner similar to the adherence of bubbles [39].

When two cells intersect, the merging procedure deforms the cells by applying a set of displacement fields to the nodes corresponding to the faces to be merged. The displacement fields must be physically admissible from a continuum mechanics standpoint, and to result in a minimum distortion for both cells, which thus minimizes the total strain energy of the two adhering cells.



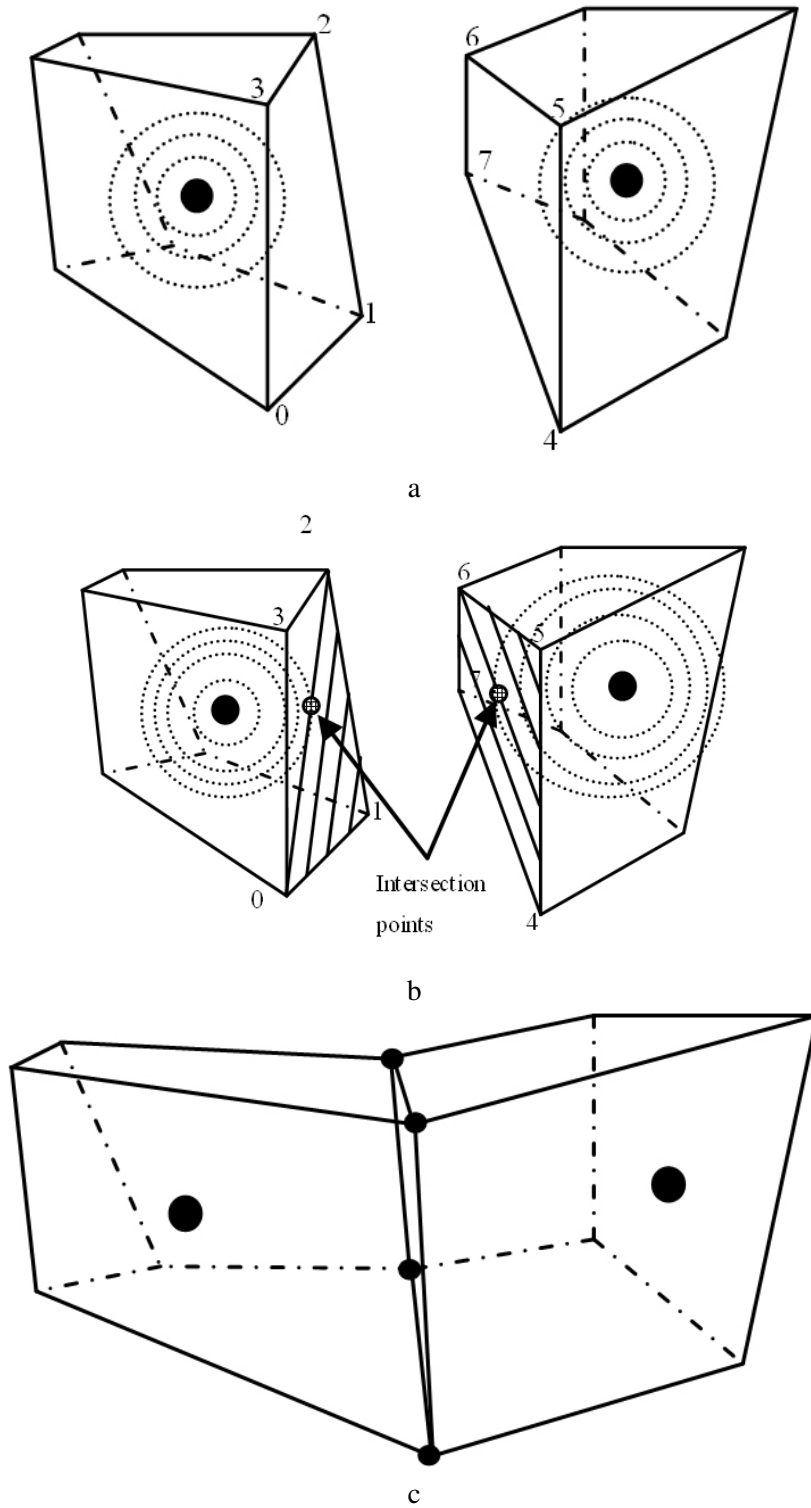


Figure 2.5: (a) Two cells prior to adhesion with radiating signals from the center of each cell. (b) The faces of the two cells to be adhered. (c) The cells after adhesion.

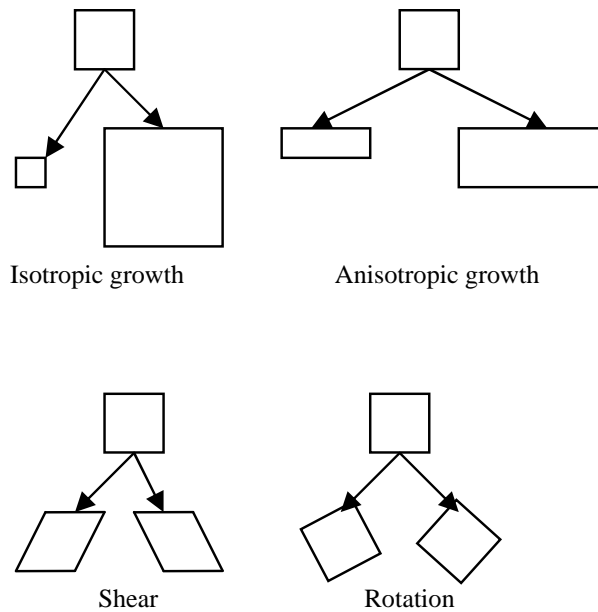


Figure 2.6: The four basic geometric operations observed in sub-regions of plants.

Three major steps are executed during the cell adhesion process; first determine the corresponding faces to be adhered; second determine the pairs of nodes to be merged, and third optimize the positions of the merged nodes, as shown in figure 9.1.

A detailed explanation of the adhesion process can be found in appendix 9.

### 2.5.3 Geometric Actions

The idea of formalizing the geometrical rules arises from studies of the development processes of plants. Plants are remarkable engineering structures that sustain high dynamic loads, for example, those generated by wind. The topological structure of plants and their material complexity provide them with the ability to sustain high mechanical stresses. During the natural embryogeny of plants, every 3D region (which is a collection of several cells) can deform according to nine different geometric operations: one for isotropic growth, two for anisotropic growth, three for shear and three for rotation [40], illustrated in figure 2.6. Every cell in the model presented here represents a 3D region that can be deformed according to these geometrical operations. Every geometric operation, excluding rotation, corresponds to a geometric rule.

These rules are the basic tools to generate any topological structure excluding arches and round objects. Nevertheless, even arcs or curved objects can be represented by linear elements with high accuracy, as shown in figure 2.7a. The last claim fails to hold for a fixed grid, as shown in Fig-

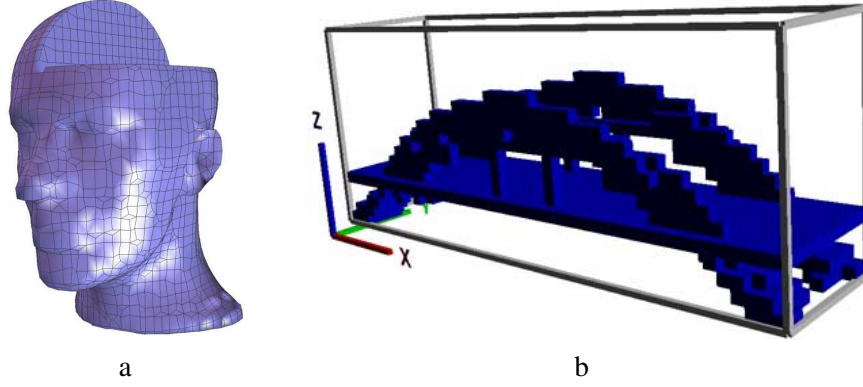


Figure 2.7: (a) 3D hexahedral linearly conformal mesh [41]. (b) 3D rectilinear grid mesh [42].

ure 2.7b. Even a simple arc has a crude representation in the fixed mesh-grid approach.

The action of a geometrical rule is always applied in the reference configuration. Since there is a one-to-one mapping between the local and the reference configuration, it will be more convenient to execute the geometrical operations locally and then map the result back to the reference configuration. In order to be consistent with the next derivations, the faces of the cell in the local configuration have been numbered according to Figure 2.8.

### 2.5.3.1 Growth

The action of this rule is to expand the cell by an amount in all directions. Assuming the cell is in the local configuration, as shown in Figure 2.3, the following mapping is defined by equation 2.2. Every point  $R, S, T$ , corresponding to a point inside the cell before applying the geometrical rule, is mapped to a point  $r, s, t$  after the geometrical rule has been applied. Both points are defined in the coordinate system of the local configuration.

$$r = \alpha_1 R ; s = \alpha_2 S ; t = \alpha_3 T \quad (2.2)$$

The coefficients  $\alpha_1, \alpha_2, \alpha_3$  representing the expansions along each of the three orthogonal axes. When  $\alpha_1 = \alpha_2 = \alpha_3$ , the growth is isotropic.

The deformation gradient tensor for this mapping is defined in equation 2.3.

$$\mathbf{F} = \frac{\partial \mathbf{r}}{\partial \mathbf{R}} = \begin{bmatrix} \alpha_1 & & \\ & \alpha_2 & \\ & & \alpha_3 \end{bmatrix} \quad (2.3)$$

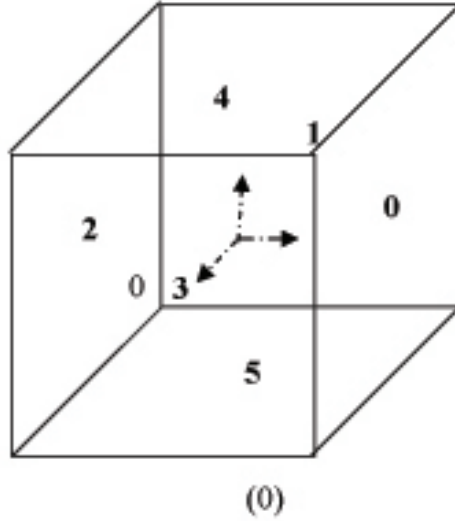


Figure 2.8: Face numbering - Local configuration.

The volume change of any infinitesimal point  $dV$  to  $dv$  can be computed according to equation 2.4.

$$\frac{dv}{dV} = \det(F) = \alpha_1 \alpha_2 \alpha_3 \quad (2.4)$$

Using equation 2.4, the volume change of the entire cell can be computed according to equation 2.5.

$$\frac{v_{cell}}{V_{cell}} = \mathbf{J} = \det(F) = \alpha_1 \alpha_2 \alpha_3 \quad (2.5)$$

The mapping in equation 2.2 is applied to the nodes of the cell in the local configuration. Since the mapping is homogeneous, mapping the nodes will set the map of the entire cell. By setting  $\alpha$  and applying this mapping, the change in volume of a cell in the local configuration will be  $\alpha_1 \alpha_2 \alpha_3$ . Define  $\tilde{V}$  to be the volume of the cell in the reference configuration prior to this mapping. A unit volume<sup>1</sup> of the cell in its reference configuration prior to the application of this mapping is given in equation 2.6. The quantities  $\mathbf{R}$ ,  $\mathbf{S}$ ,  $\mathbf{T}$  are defined as material fibers in the local configuration.

$$d\tilde{V} = N_i(d\mathbf{R}) X_i \times N_i(d\mathbf{S}) Y_i \times N_i(d\mathbf{T}) Z_i \quad (2.6)$$

<sup>1</sup>A cube with volume = 1 in the local configuration.

A unit volume of the cell in its reference configuration, after this mapping has been applied, is given in equation 2.7, where  $\mathbf{r}$ ,  $\mathbf{s}$ ,  $\mathbf{t}$  are material fibers in the local configuration after the mapping has been applied.

$$d\tilde{v} = N_i(d\mathbf{r}) X_i \times N_i(d\mathbf{s}) Y_i \times N_i(d\mathbf{t}) Z_i \quad (2.7)$$

Again using the definition of the gradient deformation tensor  $\mathbf{F}$  shown in equation 2.3, equation 2.7 can be written in a new form, shown in equation 2.8.

$$d\tilde{v} = N_i(\mathbf{F}d\mathbf{R}) X_i \times N_i(\mathbf{F}d\mathbf{S}) Y_i \times N_i(\mathbf{F}d\mathbf{T}) Z_i \quad (2.8)$$

Since  $N_i$  is a linear operator, equation 2.8 can be rewritten into the form of equation 2.9.

$$\begin{aligned} d\tilde{v} &= \mathbf{F}N_i(d\mathbf{R}) X_i \times \mathbf{F}N_i(d\mathbf{S}) Y_i \times \mathbf{F}N_i(d\mathbf{T}) Z_i \\ &= \mathbf{J}N_i(d\mathbf{R}) X_i \times N_i(d\mathbf{S}) Y_i \times N_i(d\mathbf{T}) Z_i = \mathbf{J}d\tilde{V} \end{aligned} \quad (2.9)$$

The volume ratio of the cells in the reference configuration, after the mapping in equation 2.2 has been applied, is equal to the volume ratio of the cells in the local configuration, after the mapping in equation 2.2 has been applied. This conclusion enables the growth operation to be applied in the local configuration of the cell.

### 2.5.3.2 Shear

Under the execution of a shear rule, a cell in the reference configuration will be deformed in a given direction such that its original volume will be preserved. The shearing operations involve two steps: given a directional vector originating in the reference configuration  $\mathbf{a}$ , the first step is to determine the face of the cell which first coincides with an extension of this vector. There are many ways to perform this test; the inverse isoparametric mapping method is used here. First, the direction vector is translated to the center of the cell. Next, a point along this vector, interior to the cell is chosen. By mapping the coordinates of this point to the local configuration, the inverse mapping of the vector is simply the difference between the coordinates of the inverted points, and the center of the cell in the local configuration system, which is  $(0, 0, 0)$ . The face of the cell in the local configuration coinciding with the inverted vector corresponds to the term with the maximum value of the inverted vector. The steps to compute the inverse isoparametric mapping can be found in appendix 8.2.

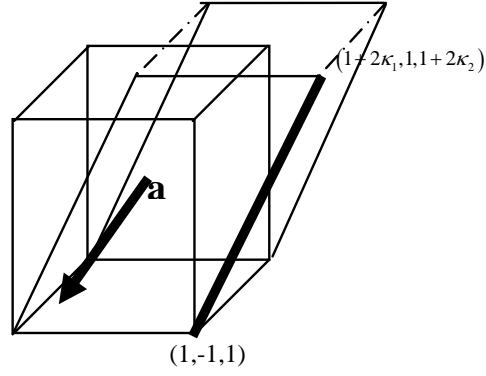


Figure 2.9: Shear operation.

The second step is to deform the cell in the local configuration such that the edges which are perpendicular to the selected face become parallel to  $\mathbf{a}$ , as shown in figure 2.9. The mapping, defined in equation (2.10) shears the cell in a predetermined direction such that the volume of the cell remains the same. The number below each box in equation 2.10 corresponds to a particular face with respect to the notation defined in Figure 2.8.

$$\begin{array}{ccc}
 \begin{array}{l} r = R + \kappa_1(S + 1) \\ s = S \\ t = T + \kappa_2(S + 1) \end{array} & \begin{array}{l} r = R + \kappa_1(T - 1) \\ s = S + \kappa_2(T - 1) \\ t = T \end{array} & \begin{array}{l} r = R \\ s = S + \kappa_1(S - 1) \\ t = T + \kappa_2(S - 1) \end{array} \\
 1 & 2 & 3 \\
 \begin{array}{l} r = R + \kappa_1(T + 1) \\ s = S + \kappa_2(T + 1) \\ t = T \end{array} & \begin{array}{l} r = R + \kappa_1(S + 1) \\ s = S \\ t = T + \kappa_2(S + 1) \end{array} & \begin{array}{l} r = R + \kappa_1(S - 1) \\ s = S \\ t = T + \kappa_2(S - 1) \end{array} \\
 4 & 5 & 6
 \end{array} \tag{2.10}$$

It is easy to show that each of the mappings in equation 2.10 is isochoric (volume preserving), by following the same derivation as in equations 2.3, 2.4, and 2.5.

The relation between the coefficients  $\kappa_1, \kappa_2$  and the target direction vector  $\mathbf{a} = \{a_1, a_2, a_3\}$  is shown in equation 2.11. As before, each face corresponds to a different relation between  $\mathbf{a}$  and  $\kappa$ .

$$\begin{array}{ccc}
\boxed{\begin{array}{l} \kappa_1 = \frac{a_2}{\sqrt{1-a_2^2-a_3^2}} \\ \kappa_2 = \frac{a_3}{\sqrt{1-a_2^2-a_3^2}} \end{array}} & \boxed{\begin{array}{l} \kappa_1 = \frac{-a_1}{\sqrt{1-a_1^2-a_2^2}} \\ \kappa_2 = \frac{-a_2}{\sqrt{1-a_1^2-a_2^2}} \end{array}} & \boxed{\begin{array}{l} \kappa_1 = \frac{-a_2}{\sqrt{1-a_2^2-a_3^2}} \\ \kappa_2 = \frac{-a_3}{\sqrt{1-a_2^2-a_3^2}} \end{array}} \\
1 & 2 & 3 \\
\boxed{\begin{array}{l} \kappa_1 = \frac{a_1}{\sqrt{1-a_1^2-a_2^2}} \\ \kappa_2 = \frac{a_2}{\sqrt{1-a_1^2-a_2^2}} \end{array}} & \boxed{\begin{array}{l} \kappa_1 = \frac{a_1}{\sqrt{1-a_1^2-a_3^2}} \\ \kappa_2 = \frac{a_3}{\sqrt{1-a_1^2-a_3^2}} \end{array}} & \boxed{\begin{array}{l} \kappa_1 = \frac{-a_1}{\sqrt{1-a_1^2-a_3^2}} \\ \kappa_2 = \frac{-a_3}{\sqrt{1-a_1^2-a_3^2}} \end{array}} \\
4 & 5 & 6
\end{array} \tag{2.11}$$

## 2.6 Environment

The environment in which the individuals are grown contains factors which every cell can sense, and which can affect the way rules are expressed. The environment can trigger the execution of rules and control their expression. Biologists studying growth and development have found that the concentration of morphogens [43] and mechanical stresses [44] are two crucial factors which influence the growth of a phenotype. Various mathematical models illuminate the role the two effects play in determining the size and shape of tissues [45, 46]. In natural systems, information from the environment is transferred to the cells through proteins known as receptors. The receptors transfer the information by generating chemicals that diffuse through the cell membrane at some level of concentration. This concentration stimulates the action of rules at the local cell level during the developmental stage.

In the simulated evolutionary embryogenesis here, as well as in nature, the relationship between the information that cells receive from the environment and the development of the phenotype is not predetermined. Rather, **conditionals** are available to the evolutionary process that sense the concentration or gradient of each morphogen. In this way, the evolutionary process establishes the relationship between information and growth and development.

The morphogens represent points, surfaces or volumes in space, with associated engineering requirements. In the artificial embryogeny presented here, two kinds of morphogens are present. The first morphogen represents an external load to be supported by the phenotype. The morphogen is located at a predetermined point and produces a chemical which continuously diffuses in space according to equation 2.12. The parameter  $s$  represents the concentration of the morphogen, while  $d$  represents the distance from the location of the morphogen. The morphogen diffuses through space impinging on the walls of each cell. The second morphogen represents the surface of the ground, to

which cells adhere when they intersect the surface.

$$s = e^{-d} \quad (2.12)$$

The phenotype is subjected to gravity effects and an external load. These two effects generate internal mechanical stresses within the cells, which can be represented by the Cauchy-Green tensor given in equation 2.13.

$$\sigma = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} \end{bmatrix} \quad (2.13)$$

By solving a finite element scheme at every time step, the mechanical stress distribution within the cells can be calculated [37]. There are six independent parameters which can be derived from the Cauchy stress tensor and added as environmental factors. The parameters are: the three principal values and the three principal directions of the Cauchy stress tensor. Since every point inside the cell has a different stress tensor, only the point which has the maximum principal stress is considered. The calculations are outlined in appendix 8.2.

Two additional environmental factors correspond to the volume and the age of the cell. The volume of a cell is simply the value of the Jacobian, defined in equation( 8.2.7) in appendix 8.2. The age of the cell corresponds to the number of time steps that have passed since the cell was created. Cells also maintain information about their distance from neighboring cells. This information is used to trigger the execution of the cell adhesion rule.

## 2.7 Genome Structure

The genome contains words which contain rules with their corresponding letters (tables 2.3 through 2.4). A word is simply a sequence of rules. The letter “Z” indicates the beginning of a set of rules within a word. A veto rule can only act on the remaining rules within a word. After every generation, a search routine looks for identical words within the same genome. These words are combined together with the letter “R” indicating the number of times the particular word will be executed in one time frame. The length of the genome can vary between different individuals, but has a predefined maximum length.



Table 2.1: Geometric Rules

ID	Name	$N$	Optional Parameters
A	Shear	1	(d,e,f,h) $\times$ fractional coefficient
B	Anisotropic growth	3	(a,b,c,g,i) $\times$ fractional coefficient
C	Isotropic growth	1	(a,b,c,g,i) $\times$ fractional coefficient
S	Isotropic shrink	1	(a,b,c,g,i) $\times$ fractional coefficient
$N =$ number of parameters			

### 2.7.1 Syntax Rules

Every rule comprises one capital letter and several lower case letters, each with a fractional coefficient. The capital letter identifies the type of a rule. The lower case letters correspond to the environmental factors which guide the action of the rule within the cell. The fractional coefficient is a number between 0 and 100 which represents the level of expression of the rule with respect to the environmental factor that follows it (a mechanism analogous to transcription factors in nature). Tables 2.1, 2.2, and 2.3 contain four columns, the first column corresponds to the identifying letter, and the second column corresponds to the name of the rule. The third column specifies how many additional parameters each rule has while the fourth column specifies the type of additional parameters, as listed in Table 2.4.

The rule execution process begins by reading the identifying letter of the rule, followed by a set of environmental factors each with its fractional coefficient. Each environmental factor is normalized to a nominal value.

aim of the normalization process is to equally scale all of the factors. The following normalization are used here: the mechanical stress is normalized to the yield stress of the material, the volume of the cell is normalized to the initial volume of the first cell, the intensity of the morphogen is normalized with the morphogen intensity impinging on the first cell at the initiation of the growth process. at the first time step, and the age of the cell is normalized with the maximum age allowed. The product of the environmental factor and the fractional coefficient specifies the level of expression of the rule. For instance, the set of letters S30g reduces the volume of the cell by 30 percent.

In a similar way, the word “R1ZC10i” corresponds to: R1 repeat once; Z word boundary indicator; C10i cause the cell to grow isotropically by 10% based on the load morphogen concentration that was measured by that cell.

Table 2.2: Cell-Type Operations

ID	Name	$N$	Optional Parameters
D	Cell division	U	(d,e,f,h)
K	Cell death	0	none
F	Cell differentiation	0	none
$N$ = number of parameters U = Unlimited number			

Table 2.3: Veto (conditional) operations

ID	Name	$N$	Optional Parameters
V	Suppress below	1	(a,b,c,g,i) $\times$ fractional coefficient
W	Suppress above	1	(a,b,c,g,i) $\times$ fractional coefficient
$N$ = number of parameters			

Table 2.4: Cell information

ID	Description
a	Maximum principal stress normalized with the yield stress
b	Middle principal stress normalized with the yield stress
c	Minimum principal stress normalized with the yield stress
d	Principal vector corresponding to the maximum principal stress
e	Principal vector corresponding to the middle principal stress
f	Principal vector corresponding to the minimum principal stress
g	Cell size
h	Load morphogen intensity
i	Load morphogen direction
t	Cell age

## 2.8 Metabolism and Thermodynamics

A thermodynamic energy model which balances the energy required to maintain the organism mass with the energy required to create a new mass [47], is incorporated in the growth and development process used here. The amount of energy  $E_c$  that each cell consumes in a given time step, is proportional to its metabolic rate  $B_c$ . Part of this energy is used for maintaining the existing phenotype while the remaining energy may be used for creating new mass, as shown in equation 2.14.

$$E_c = E_0 B_c \Delta t \quad (2.14)$$

Using Kleiber's Law [48, 49], and assuming small variation in the volume of the cells (Small variation of the volume of cells is a requirement in the fitness function), the metabolic rate  $B_c$  of each cell is proportional to the size of the phenotype  $S$  (the total volume of the phenotype) divided by the number of cells,  $N_c$ , as shown in equation 2.15. The parameter  $E_0$  is a proportionality constant which sets the energy scale.

$$B_c \propto \frac{S^{3/4}}{N_c} \quad (2.15)$$

By combining equations 2.14 and 2.15, and by setting  $E_0 = 1$ , a thermodynamic size limit can be specified for the phenotypes, as shown in equation 2.16. At the beginning of every time step, each cell contains an amount of energy  $E_c$ . This energy is utilized by the cell to execute its genome. Every rule (operation) may consume a different amount of energy.

$$E_c = \frac{S^{3/4}}{N_c} \quad (2.16)$$

The following scheme addresses the process of assigning energies to rules. First, consider a scenario where there is non-stop equally sized cell production. Assuming  $E_0 = 1$ ,  $E_c$  can be plotted, as shown in figure 2.10. The total amount of energy,  $E_c$ , is reduced as the number of cells increases since the volume of the phenotype increases. Setting the energy consumption of the cell-division rule to be above  $E_c(N)$ , forces a weak upper limit to the size of the phenotype. All the other energies are set to be less than the division energy. This enables the phenotype to change its topology without adding new mass.

The advantage of using this approach is that there is no predefined upper bound, or other limit, on the size of the phenotype. Even when the phenotype reaches the thermodynamic limit, this

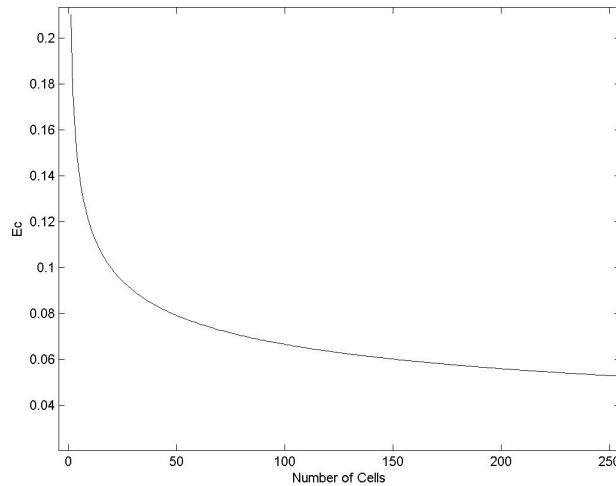


Figure 2.10: The amount of energy  $E_c$  available to each cell during growth, as a function of the total number of cells in the growing phenotype.

approach will permit new mass to be created at the expense of removing existing mass. This potentially changes the topology of the phenotype. However, the thermodynamic balance will not prevent phenomena such as unlimited cell division or extermination of the entire phenotype. These last phenomena are addressed by evolution and disease mechanisms.

## 2.9 Time Increments

The growth and development processes in nature proceed continuously. In order to simulate these processes numerically, a time step has been defined. A time step starts when the first cell executes its first rule and ends when the last cell executes its last rule.

Between the end of a time step and the beginning of the next time step, three processes are executed. The first process is an iterative process which repairs damaged cells using two optimization schemes, discussed later in Section 2.10.1. The second process is an evaluation of the phenotype by computing the mechanical stress distribution across the cells using a finite element scheme. In the third process, the internal energy of the cell is updated according to equation 2.16. In this discretization time scheme, rules are always executed in the reference configuration. In other words, even though an execution of rule might change the topological representation of a phenotype and thus effect the stresses inside the cells, these changes will only take place in the next time step.

## 2.10 Diseases

Individuals may suffer from a disease during growth and development. A disease only occurs as a consequence of a defective genome, and diseases adversely affect the growth process of the phenotype. When a developmental disease is detected in an individual, a penalty is applied to the measure of its performance. The following diseases are present.

**High cell division rate** : an upper limit to the number of divisions at one time step is set. If the number of cell divisions exceeds the threshold value, it is identified as a disease.

**Self extermination** : A defective genome might contain instructions that will eliminate all of the cells in that phenotype at some point during its development. This is identified as a disease.

**Morphology of the cell** : As part of the repair process (which is a subprocess of the growth and development process illustrated in Figure 2.2) every cell is regularly tested with equations 2.17 and 2.19 to determine whether it is “tangled” or highly deformed. An automatic repair mechanism (which is described in the following section) attempts to fix any cell that exceeds threshold values. If the repair mechanism fails to repair the cell, then the growth process is stopped. The phenotype is then evaluated, and a performance measure is assigned to the phenotype based on its current state. Since the phenotype fails to reach maturity, the performance measure is penalized (reduced by a factor). This process produces a comparative benefit for individuals that reach maturity, but will not eliminate phenotypes that have good properties but fail to reach maturity.

### 2.10.1 Repair Process

Each phenotype is a collection of cells which is also a finite element mesh. During the growth and development process, cells may be deformed significantly due to the execution of rules (especially geometric operations). In order to accurately evaluate the mechanical stress distribution, the shape of the cells must obey two restrictions: convexity and deformation. The convexity restriction states that every cell has to be convex, which is determined by computing the Jacobian in equation 8.2.7 (in appendix 8.2) and checking that it is positive. If the Jacobian is negative then one or more nodes are identified as being tangled. A repair process will then untangle these nodes, as described below. Although the repair process might alter the search landscape, experience has shown that the evolutionary scheme will evolve phenotypes such that their development is robust

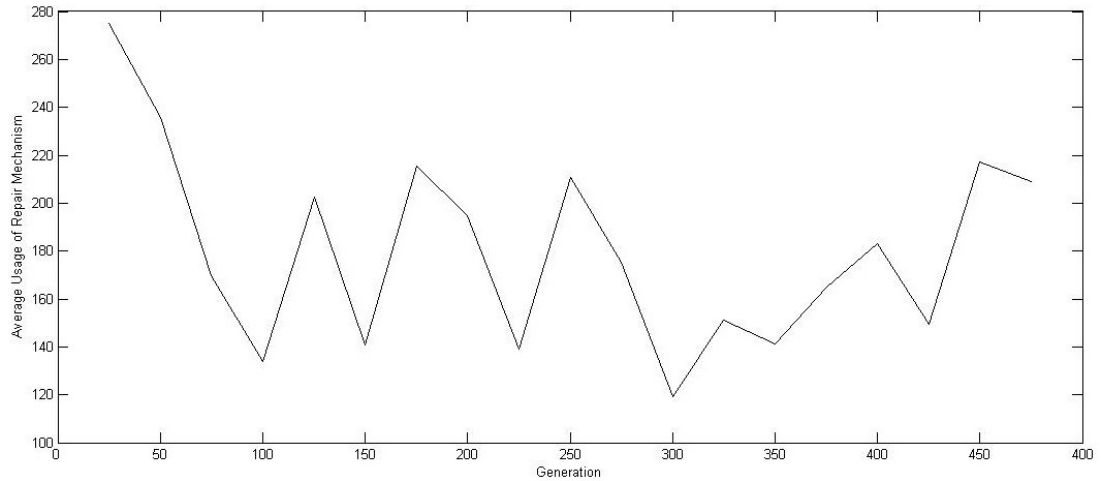


Figure 2.11: The average usage of the repair mechanism.

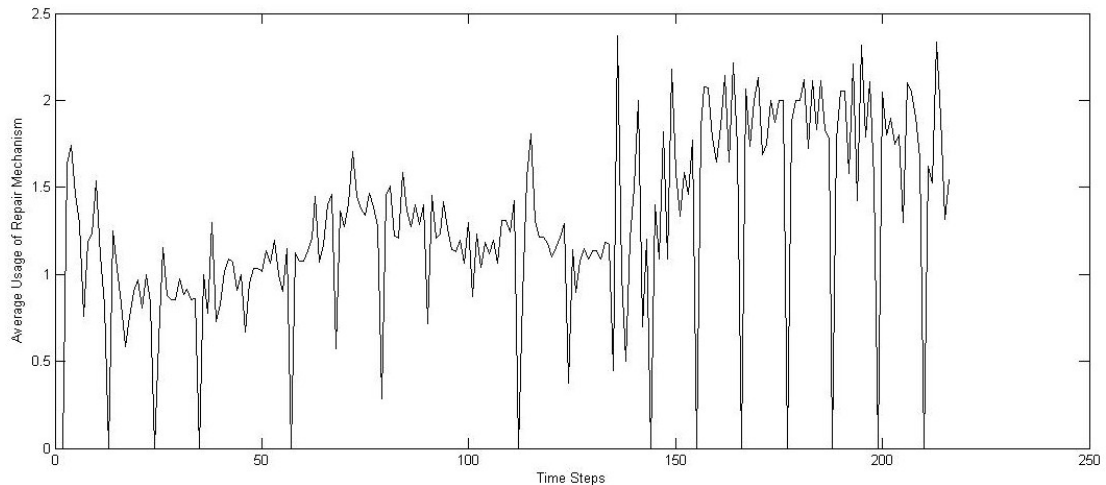


Figure 2.12: The average usage of the repair mechanism with respect to time for one phenotype during the growth and development process.

to these changes. Figure 2.11 shows the average usage of the repair mechanism during the process of evolution. The average usage is defined as the total number of times the repair mechanism is used by all cells through all of the time steps for the phenotype to reach maturity, divided by the sum of the number of cells in the phenotype at each time step. The use of the repair mechanism generally decreases in the beginning evolutionary process. Figure 2.12 shows the utilization of the repair mechanism during the growth process of one phenotype. The  $y$  axis represents the average use of the repair mechanism with respect to the cells. On average, 1.27 cells were repaired at each time step.

Following the work of Peter Knupp [50, 51, 52, 53, 54],  $\alpha_m$  is defined as a scalar according to

equation 2.17, where  $x_m^{(1)}, x_m^{(2)}, x_m^{(3)}$  correspond to the coordinates of three nodes which are adjacent to node  $x_m^{(0)}$ , where  $m$  is a dummy index, and a negative value of  $\alpha_m$  corresponds to a tangled node which needs to be untangled. The untangling process is performed using the conjugate gradient method with a global function that is to be minimized defined in equation 2.18. This process is described in detail in [53].

$$\alpha_m = \left(x_m^{(1)} - x_m^{(0)}\right) \times \left(x_m^{(2)} - x_m^{(0)}\right) \times \left(x_m^{(3)} - x_m^{(0)}\right) \quad (2.17)$$

$$f_0 = \frac{1}{2} \sum_{m=1}^M \{|\alpha_m| - \alpha_m\} \quad (2.18)$$

The second restriction corresponds to deformation level. A conditional number,  $g_m$ , is defined in equation 2.19, where  $x_{m,k}$  are the adjacent coordinates to node  $x$ ,  $g_m \geq 1$ .

$$\begin{aligned} e_{m,k} &= x_{m,k} - x \\ J_m &= [e_{m,1}, e_{m,2}, e_{m,3}] \\ g_m &= \det \left| J_m^T J_m \right| \end{aligned} \quad (2.19)$$

The value of  $g_m$  represents the degree of deformation of the cell. A high conditional number corresponds to a highly deformed cell that needs repair. The repairing mechanism is to minimize the objective function  $f_0$  defined in equation 2.20, where  $\alpha_m$  is defined in equation 2.17. Further details may be found in [52].

$$f_0 = |J_m| |Adj(J_m)| / |\alpha_m| \quad (2.20)$$

Each of the repair processes is an iterative optimization process. If either process fails to converge after a large number of iterations, a disease will be identified

## 2.11 Maturity

Once the growth process becomes stable, the phenotype is identified as being mature. The maturity stage is defined when no new mass is created or removed from the phenotype in a predefined time frame. Determining the length of this time frame is tricky since a phenotype might reach a limited stable region and then may continue to grow. In the method presented here, the following heuristic was utilized: the stability time frame was taken to be twice the previously observed period

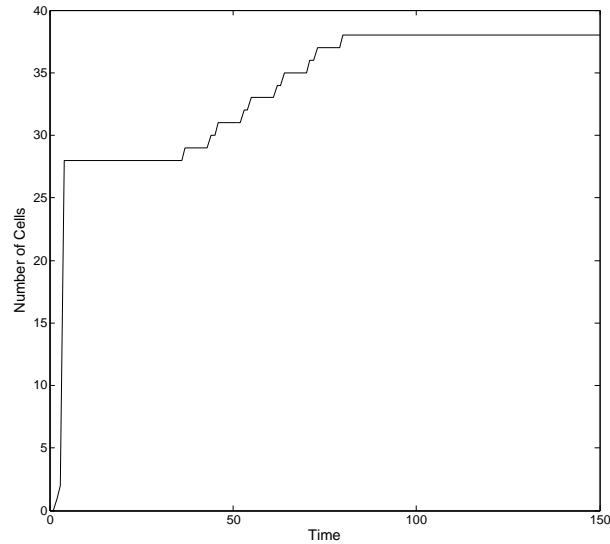


Figure 2.13: A typical development process.

of stable size.

Figure 2.13 shows a typical growth process of a representative phenotype. The  $y$ -axis marks the number of cells in the phenotype while the  $x$ -axis represents the development time. During the initial stage of the growth process, the production rate of new cells is relatively high. As time evolves, this rate decreases and stabilizes. There are two stabilization regions (plateaus) in which no new mass is created. The first plateau is unstable since the phenotype continues to produce new mass after some time. The second plateau is stable. The phenotype has reached stability such that no additional mass is created.

## 2.12 Fitness Evaluation

### 2.12.1 Fitness Description

Each phenotype is evaluated with respect to six performance attributes: mechanical stress, weight, shape of the cells, distance from the load point, age of the phenotype, and the maximum volume of the largest cell in the phenotype. The mechanical stress corresponds to the maximum value of the von Mises stress in the phenotype. The weight corresponds to the weight of the entire phenotype. The shape of the cells corresponds to the average value of the conditional number of all the nodes in the phenotype, defined in equation 2.19. The distance to the load point corresponds to the minimum distance that exists between any one of the cells to the load point. If one of the



cells intersects with the location of the source of a morphogen then this distance is zero. The age of the phenotype corresponds to the number of time steps a particular individual has been growing without developing a disease. The parameter that establishes the maximum volume of a single cell puts constraints on the size of the individual cells, preventing extremely large cells in the phenotype.

### 2.12.1.1 Aggregation

In order to establish the fitness of individuals in the population, two approaches have been used here: aggregation of the multiple values into a single scalar [55, 56], and the Pareto optimization method [57]. In both cases, the objective is to minimize the fitness value(s).

The aggregation approach builds on prior engineering design work of Scott and Antonsson [58], where both importance weighting and degree of compensation among the variables are utilized. The degree of compensation specifies how a strong value of a particular performance variable may compensate for a deficiency of another variable.

To begin, the value of each performance variable is mapped to a preference value between 0 and 1 by a preference function, where a preference value of 1 corresponds to a perfectly acceptable value of the performance variable; a preference of 0 corresponds to a completely unacceptable value of the performance variable. A preference function  $\mu_i$ , maps every variable  $S_i$  to a value on the real interval line  $[0, 1]$  using equation 2.21, as illustrated in Figure 2.14. The variable  $a$  represents the value such that any value below it will be considered to have a preference value of 1. The variable  $b$  represents the maximum value such that any value above will be considered to have a preference value of 0. Setting  $a$  and  $b$  controls the range of feasible solutions and also provides the ability to put constraints on the phenotype.

The slope of the function specifies the improvement rate which corresponds to a particular variable. One of the major concerns which must be taken into account in choosing the preference function corresponds to the volatility of the individuals within the population. A single mutation in the genome may turn an individual with high performance into a non feasible one. While this phenomena provides the ability to explore a large span of the space of possible solutions, its drawback is that it significantly penalizes individuals with poor performance. This phenomena of poor behavior is often observed at the early stages of the evolutionary process where most of the population has low measures of performance.

The solution implemented here is first to identify the particular variables in the fitness functions that have high volatility and produce evaluations of poor performance. These variables will have

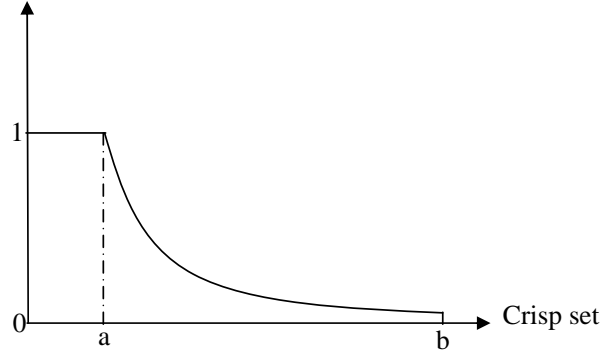


Figure 2.14: A representative preference function.

a preference mapping in the form of an exponentially decaying function, shown in equation 2.22, where  $\alpha$  sets the slope. By reducing the value of  $\alpha$ , the preference corresponding to the variable  $b$  is increased. The value of  $b$  can be increased such that a wider range of values will be considered and thus, individuals that are poorly performing can be distinguished.

$$\mu_i : S_i \rightarrow [0, 1] \quad (2.21)$$

$$\mu_i = \left\{ \begin{array}{ll} 1 & S_i < a \\ e^{-\alpha(S_i - a)} & a \leq S_i < b \\ \varepsilon & S_i \geq b \end{array} \right\} \quad (2.22)$$

Once all the preference functions have been defined, the weight and the degree of compensation of these values are used in the relation 2.23. The weight for the  $\mu_i$  performance variable is represented by  $\omega_i$ , and  $s$  represents the degree of compensation. Observing equation 2.23, it can be seen [58] that  $s = 1$  corresponds to a weighted sum, the  $\lim s \rightarrow 0$  corresponds to the weighted product  $(\mu_1^{\omega_1} \mu_2^{\omega_2} \dots \mu_n^{\omega_n})^{\frac{1}{\omega_1 + \omega_2 + \dots + \omega_n}}$ , and  $\lim s \rightarrow -\infty$  corresponds to  $\min(\mu_1, \mu_2, \dots, \mu_n)$ . Here, only compensation values  $s \leq 0$  are used.

$$\mathcal{P}_s((\mu_1, \omega_1), (\mu_2, \omega_2), \dots, (\mu_n, \omega_n)) = \left( \frac{\omega_1 \mu_1^s + \omega_2 \mu_2^s + \dots + \omega_n \mu_n^s}{\omega_1 + \omega_2 + \dots + \omega_n} \right)^{\frac{1}{s}} \quad (2.23)$$

Table 2.5: Fitness function parameters

ID	Value
$s$	-1.0
$w_{Stress}$	1.0
$w_{Mass}$	0.1
$w_{Shape}$	0.1
$w_{Distance}$	1.0
$w_{Age}$	0.5
$w_{MaximumVolume}$	1.0

## 2.13 Example

The approach described above has been applied to problems similar to one observed in engineering and nature. The configuration of a structure to be synthesized is one that is capable of supporting a highly variable load 7 meters above the ground. In addition, the structure is to be as lightweight as possible.

In order to provide a basis for comparison, a simple solution has been constructed to serve as a reference, illustrated in Figure 2.15. The solution is a straight vertical beam with a 1 meter by 1 meter cross section. The solution in Figure 2.15 corresponds to a fitness value of 6.2054, by using equation 2.23, with the weights  $w$  and the degree of compensation  $s$  set according to Table 2.5.

Equation 2.23 provides an aggregate measure of performance between 0 and 1, such that better performance corresponds to higher fitness such that the goal is to minimize the aggregated function. The inverse of equation 2.23 is used here, shown in equation 2.24 for fitness  $\mathcal{F}_s$ .

$$\mathcal{F}_s((\mu_1, \omega_1), (\mu_2, \omega_2), \dots, (\mu_n, \omega_n)) = \frac{1}{\left(\frac{\omega_1 \mu_1^s + \omega_2 \mu_2^s + \dots + \omega_n \mu_n^s}{\omega_1 + \omega_2 + \dots + \omega_n}\right)^{\frac{1}{s}}} \quad (2.24)$$

Nature has evolved phenotypes that address this problem in different ways. Trees, for instance, in addition to other functions, support loads generated by their own structure and the wind. Bones support gravity and muscle loads.

As described above, two morphogens are present in the environment: one represents the load to be supported; the other represents the ground, illustrated in Figure 2.16. To introduce variation into the environment, the direction of the load changes at each time step during development of the individual phenotype, as shown by the multiple load vectors in Figure 2.16 and the pink sector in Figures 2.19 through 2.23 and 2.28. Prior work has shown that environmental changes can promote robustness [59] and modularity [22].

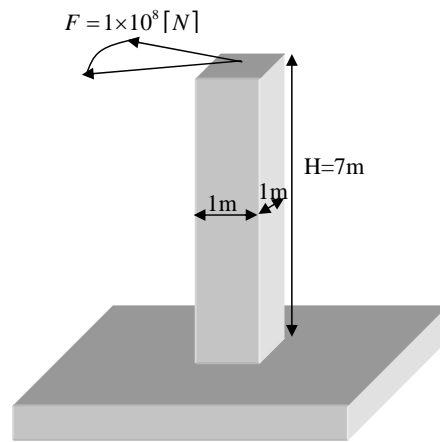


Figure 2.15: A simple reference solution.

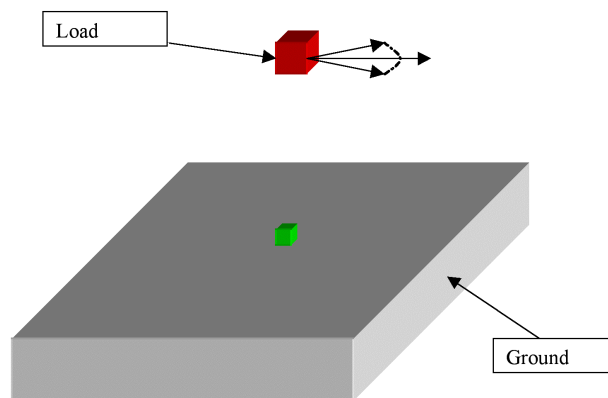


Figure 2.16: Initial setting of the growth process. The gray box represents the ground. The small green cube is the initial cell of an individual. The red cube represents the load. The load is 20,000 N and its direction is randomly selected in the range  $\pm 45$  degrees at each time step during development of the individual phenotype. The final distance between the ground and the load is 10 meters.

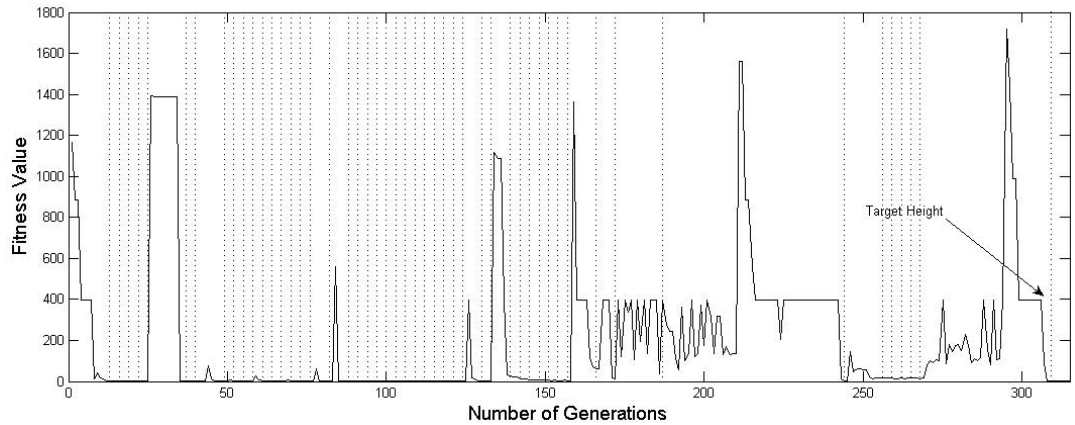


Figure 2.17: The evolution of fitness values with respect to the number of generations. The vertical dashed lines show when the height of the load was increased.

As mentioned previously, the height of load morphogen is periodically increased during the evolutionary process.

## 2.14 Results

The simulated evolution was run in parallel on 70 processors for 24 hours. The initial population contained 400 individuals, each one starting with a random genome. Figure 2.17 shows the fitness value of the best phenotype in the population with respect to the number of generations, the dashed lines correspond to times during the evolution when the height of the load was increased. It can be seen that the fitness value oscillates during the evolutionary process. This phenomena corresponds to the increasing increments of the height of the load. After every increase, it takes several generations to evolve the phenotypes with respect to the new location of the load and thus to minimize fitness.

Figure 2.18 shows the fitness evaluation of the best phenotype once the load has reached the desired height of 7 meters. The dashed line corresponds to the fitness value of the reference solution. The evolved phenotype is approximately 3 times better in performance than the reference solution.

The fitness values in Figure 2.18 exhibit small oscillations. This phenomenal has been observed many times during the evolutionary process and is a consequence of the variability in the environment. This kind of variability can result in different phenotypes grown from the same genotype through the regulation of genes/rules during the growth process.

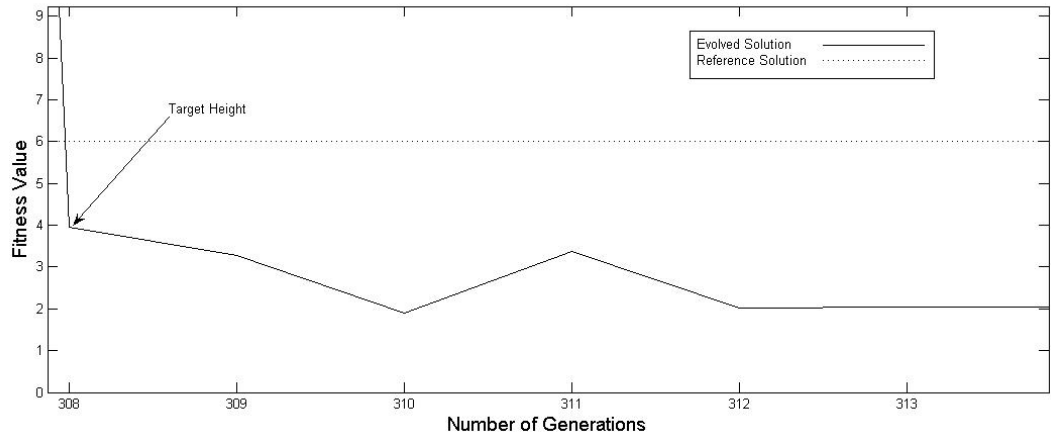


Figure 2.18: The evolution of fitness values once the phenotype has reached the desired height.

### 2.14.1 Phenotype Analysis

A typical developmental process of an evolved phenotype is shown in Figure 2.19. The development begins with a single cell and two morphogens (Figure 2.19a). The load is simulated by a vector force that randomly changes direction within the pink sector. Prior to the phenotype growing to reach the height of the load, it is only subjected to gravity (Figure 2.19b). Once the phenotype grows to reach the final height of the source of the load morphogen, the phenotype is subjected to the full load. The colors of the cells represent mechanical stress. Green corresponds to low stresses (below the yield stress), and red corresponds to high stresses (above the yield stress) (Figure 2.19c).

The topology is similar to a pyramid, but unfortunately, almost all of the cells in the phenotype are over stressed (and are therefore colored red) indicating that this phenotype does not perform well in supporting the load (fitness = 102.54). As a result, this genome is unlikely to be selected for crossover, and therefore its genetic information is likely to be eliminated from the evolution.

Figures 4.1 and 4.2 show two phenotypes that have been evolved after 400 generations. Figure 4.1 shows two different views of one phenotype and Figure 4.2 contains two views of a second. Both phenotypes are able to support the load, indicated by the color of the cells. However, the configurations of the two are different. The first phenotype is composed of two major modules (structural elements) that are perpendicular to each other. One of the modules is primarily vertical, while the other extends diagonally, which increases the ability of this phenotype to support the varying load. The two modules are connected to each other by means of cells that transfer the load from one module to the other, shown in Figure 4.1b. The second phenotype has the configuration of a

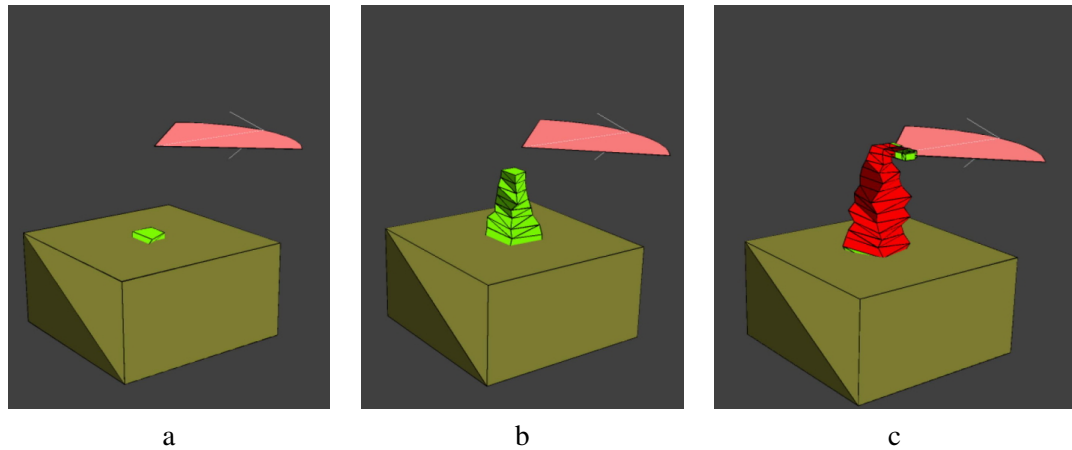


Figure 2.19: Three stages during the developmental process of a single evolved phenotype. The colors represent mechanical stress. Green corresponds to low stresses (below the yield stress), and red corresponds to high stresses (above the yield stress). The white arrow represents the magnitude and principal direction of the load. The pink sector represents the range of variation of the direction of the load.

double helix (Figures 4.2a and b). This phenotype contains two primary modules (distinct structural elements) that are wrapped together and connected at multiple locations. The helix has a unique structural topology which makes it able to support loads that vary in direction.

Characteristics of modularity (*i.e.*, distinct structural elements) can be observed in the evolved phenotypes. At this stage of the research the degree modularity of each phenotype cannot be quantified, and therefore no measure is calculated or reported. However, both of the modular characteristics shown here have spontaneously emerged without directly imposing them in the fitness function or constraining the configuration of the phenotype.

### 2.14.2 Development Analysis

Figure 2.22 illustrates the embryogenesis of the helically shaped phenotype shown in Figure 4.2. Beginning with a single cell attached to the ground, the evolved set of rules guide the growth and development of the phenotype, as described below.

**Figure 2.22a.** Initially, cell division occurs along the axis of the original cell facing towards the load morphogen, creating a vertical stack of 3 cells. Then the top cell in the stack divides laterally.

**Figure 2.22b.** Next, the lateral cell divides normal to the surface of the cell facing towards and away from the load morphogen. This creates the starting point for the two principal modular elements of the double helix of the structure.

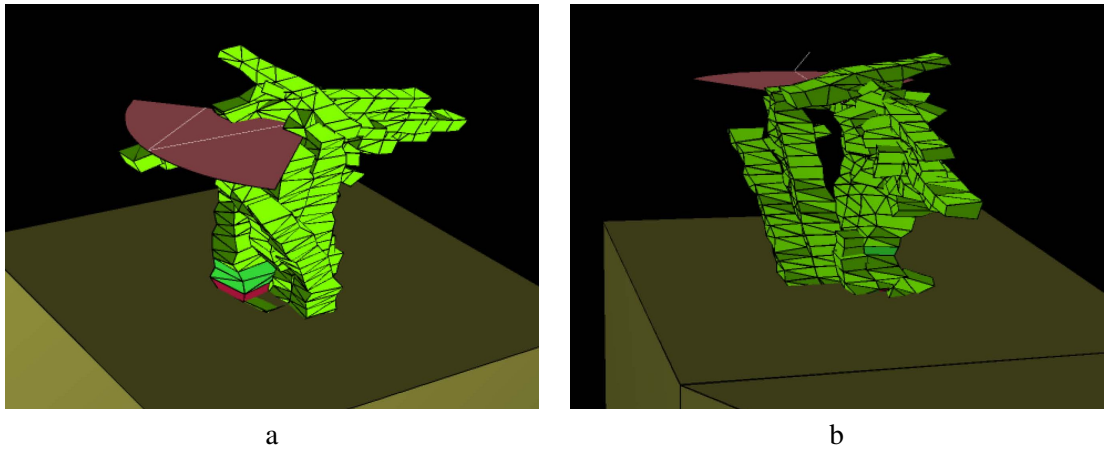


Figure 2.20: Two views of a first phenotype (fitness value = 2.11), which is able to support the load, as indicated by the predominantly green color of the cells.

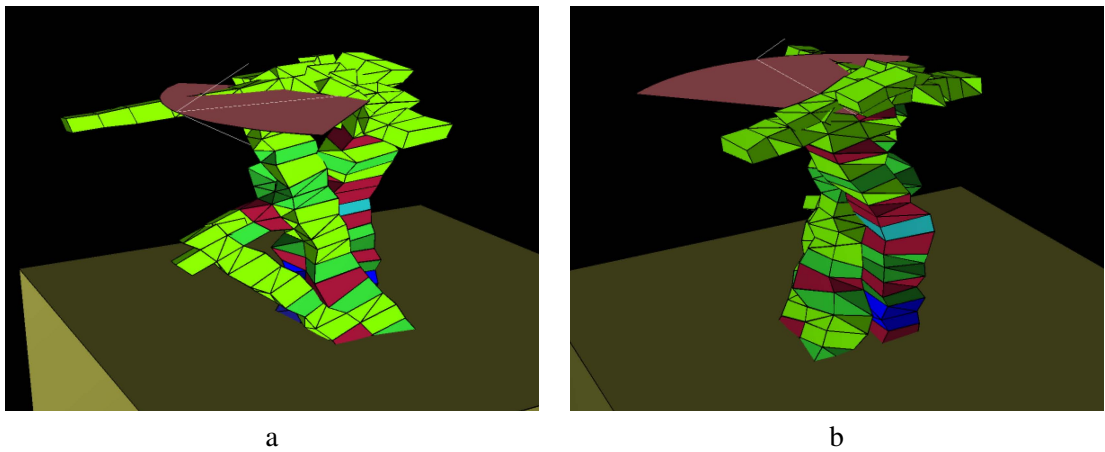


Figure 2.21: Two views of a second phenotype (fitness value = 2.07), displaying a double-helix configuration. This phenotype is also able to support the load.



**Figure 2.22c.** Growth of both elements continues. The 3<sup>rd</sup> cell from the top of the right-hand element divides laterally, creating the starting point for the 3<sup>rd</sup> modular element.

**Figure 2.22d.** Growth of all three modular elements continues, approaching the load morphogen. Note that all cells are colored green, indicating that they are lightly stressed. A fourth modular element begins to grow diagonally into the ground near the left-hand side of the base of the structure.

**Figure 2.22e.** Growth of the composite structure continues and reaches the load morphogen, inducing load onto the structure, as reflected in the red color of many of the cells. The bright red color indicates cells that are carrying stress above the yield stress of the material. The second and third modular elements have merged (through the action of cell adhesion). The fourth modular element continues to grow diagonally upwardly and can be seen as the green cells extending to the right of the structure, near its base.

**Figure 2.22f.** Additional growth and adhesion of cells near the top of the two principal modular elements, results in reduction of the stress in all of the cells to levels below the yield stress of the material.

Figure 2.23 shows four different phenotypes which have been developed from the same genotype. The first three have similar fitness values and similar topology with some degree of variation. The fourth phenotype has different topology, since it has not completely developed. Most of the cells are over stressed, which results in a higher fitness evaluation (and therefore lower performance).

Figure 2.24 shows the fitness evaluations of hundreds of phenotypes which were developed from the same genotype. Two peaks of the fitness values can be observed, with nearby distributions similar to a Gaussian distribution. Phenotypes with fitness values near the lower peak have topology similar to the phenotypes in Figures 2.23a, b, and c. The second group corresponds to phenotypes that perform more poorly (and hence have a higher fitness value) and are similar to the phenotype shown in Figure 2.23d. Due to variability in the environment, some phenotypes cannot complete their development process properly at some time step. The phenotypes which are able to continue developing after this time step usually continue to develop into successful mature phenotypes similar to those shown in Figure 2.23a, b, and c.

### 2.14.3 Genome Analysis

The genome that gives rise to the phenotype depicted in Figures 4.2 and 2.22 is shown in Figure 2.25. There are 7 different words in the genome with one word repeated 1,113 times. This word

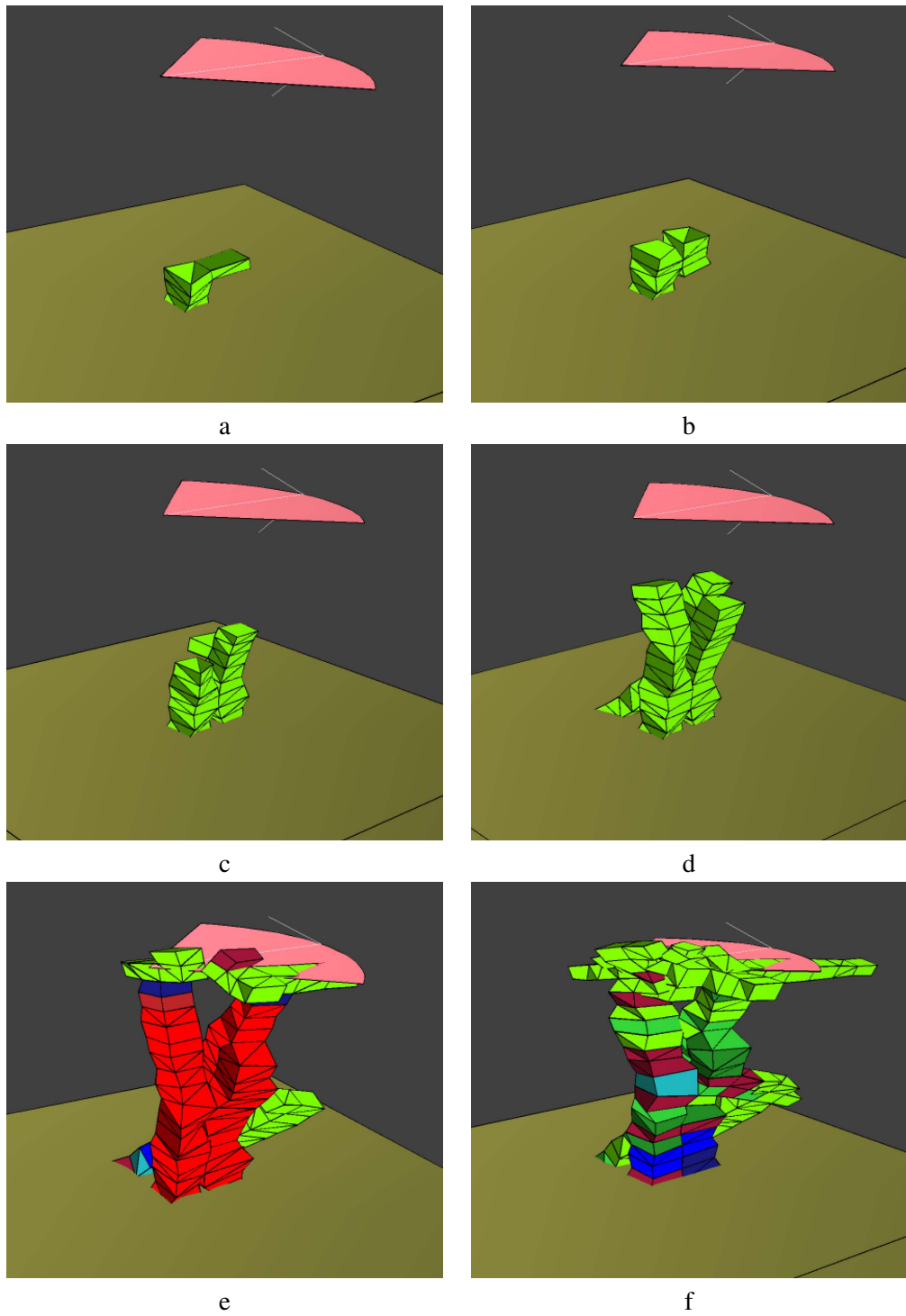


Figure 2.22: Six stages in the growth and development of the phenotype shown in Figure 4.2.

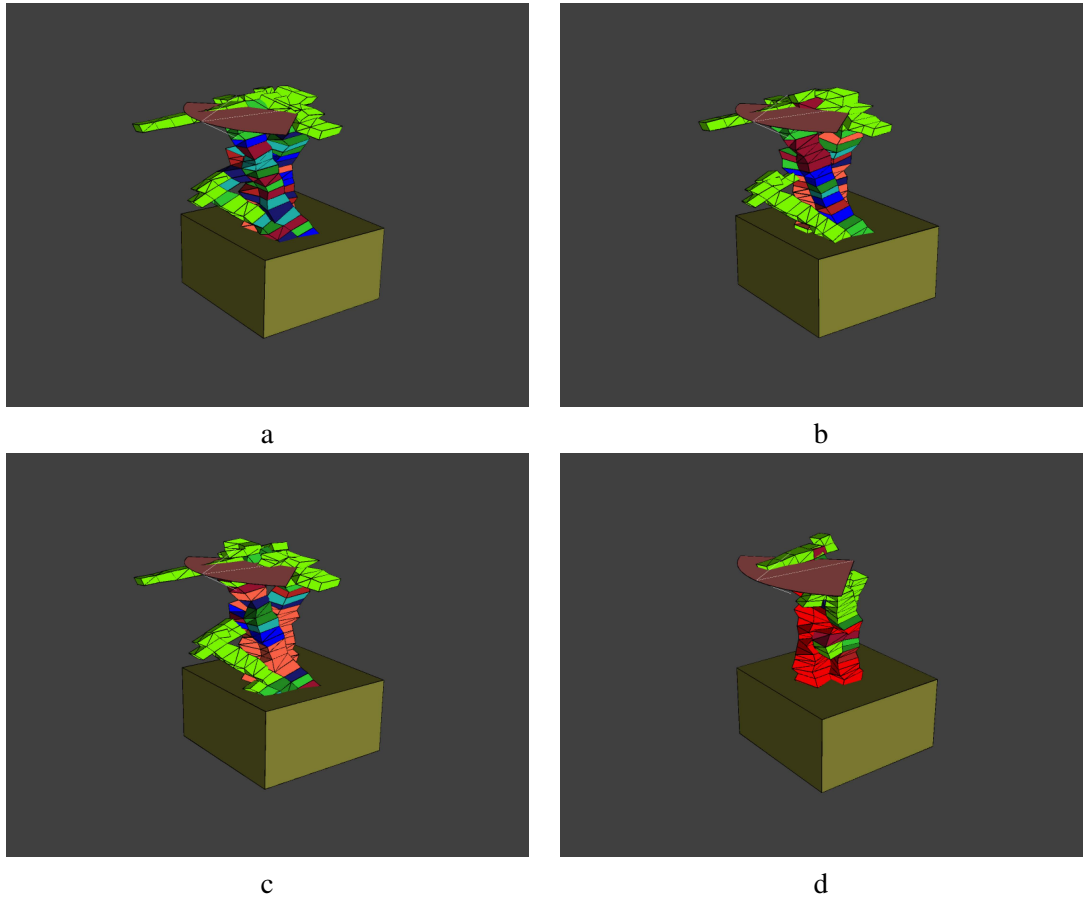


Figure 2.23: Four different phenotypes, growing from the same genome. The topologies of phenotypes a, b, and c, are almost identical, aside from small variations which result from the environment. The phenotype d has a visibly distinct topology.

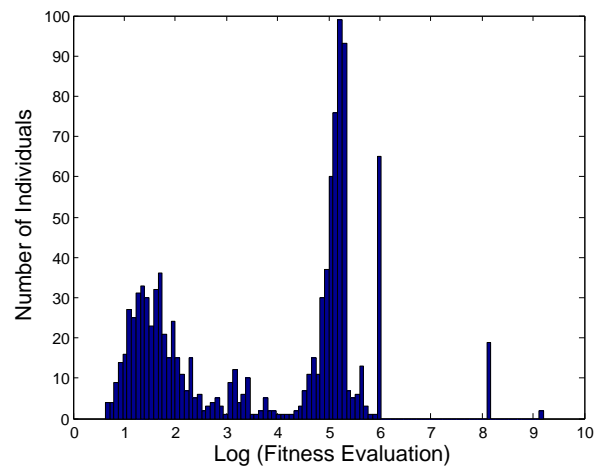


Figure 2.24: The distribution of fitness values of phenotypes grown and developed from a single genotype.

```

R1Z0S3hC7aV95gB9b6c6hC5gC0cB1h9b6cB2c4b0gDidiS5cDi
R1113Z2W158tV72gB3b2c3bDiiiK
R1Z2W110bK
R1Z2A1dC0gB1g2c8gV72gB3b2c3bDiii
R1Z0C5hV95gDifS2cC2hK
R2Z1W5tV102aS5a
R1Z1W135gKC9g

```

Figure 2.25: The genome for the phenotype shown in figures 4.2 and 2.22 contains 9 different words.

contains four major characteristics: veto rules, anisotropic growth, cell division and cell death. The veto rules suppress the execution of this word when the phenotype reaches a certain age, which means that this word is executed frequently at the beginning of the developmental stage and produces a large number of cells. The cell death rule separates a group of cells and promotes the creation of two struts in the phenotype. The shearing operation turns the struts into two helices which wrap around each other.

The mechanism of rapid cell production at the initial stage of development has also been observed in nature. Embryo cells divide much more frequently at the beginning of development than at maturity. The repetition of rules in the genome may be one of the factors that produces a degree of modularity in the phenotype [1].

#### 2.14.4 Additional Runs

One example of a phenotype produced by the combined processes of evolution and development which exhibits characteristics of modularity is presented above. Additional evolutions synthesized new phenotypes. One of these phenotypes is shown in Figure 2.28. Even though the presented phenotype appears to be much less modular than the previous one, it has the capability to support the load, yet remains relatively light weight. This phenotype has a narrow element in the main direction of the load. This element has two effects: it increases the strength of the phenotype in the main direction of the load and it is relatively light. The base element in the phenotype is rather wide, which dramatically reduces the bending stresses on the base of the phenotype, where the bending stresses are largest. This result demonstrates that there are many alternative solutions to this type of problem. Each run of the evolutionary process synthesizes a new solution.

### 2.14.5 Constraints

As in practical engineering, where constraints play a major role during the design process, several physical constraints are active here, including thermodynamics and cell intersections, *etc.* An internal constraint to not grow ill-conditioned mesh configurations is imposed through the disease mechanism, which repairs or eliminates solutions with high numerical error. Additional constraints could be incorporated in the fitness function, for example phenotypes that exceed predetermined geometric limits can be given low fitness values. The advantage of not including such external constraints is that the evolutionary process is able to freely explore the design space, and synthesize solutions that provide high levels of performance without being constrained to a predetermined configuration.

### 2.14.6 Robustness

Prior results have shown that including expected variations in the evaluation of the fitness of individuals during evolution produces phenotypes that are robust to the variations [59]. In order to evaluate the robustness of the phenotype shown in Figure 4.2, two plots are presented. Figure 2.26 shows the maximum stress in the phenotype (normalized to the yield stress of the material) as a function of the angle of the load vector. Figure 2.27 shows the fitness value of the phenotype as function of the angle of the load vector. The angle of the load vector at the center of the pink sector in Figure 4.2 corresponds to the value 0. As can be seen in the data, the lowest fitness value (1.456, which corresponds to the highest performance) is achieved when the load angle is zero, and increases in a generally symmetrical pattern between  $\pm 45$  up to a fitness value of 2.07. This corresponds to a 30% change over the entire range of load vector angles, which indicates robust ability of the phenotype to successfully support the load over the full range of variation in the angle of the load vector.

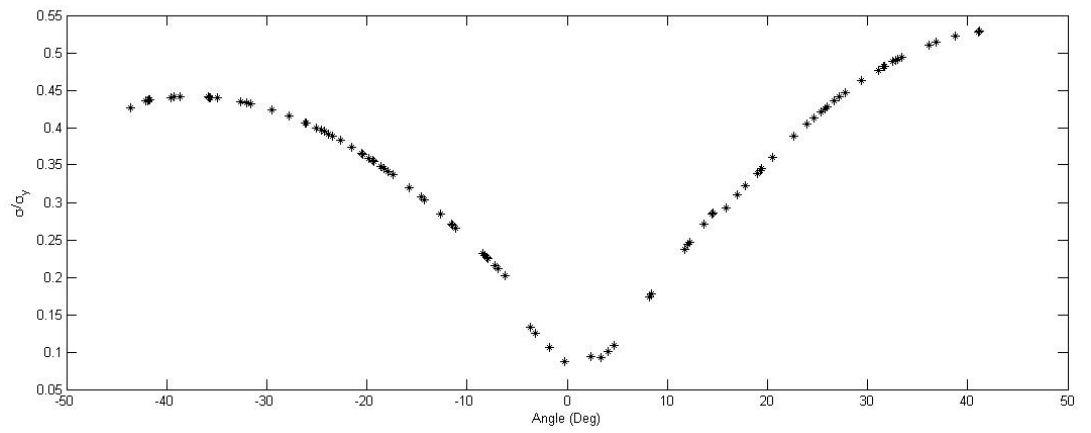


Figure 2.26: Maximum mechanical stress, normalized to the material yield stress ( $\sigma/\sigma_y$ ) vs. angle of the load vector.

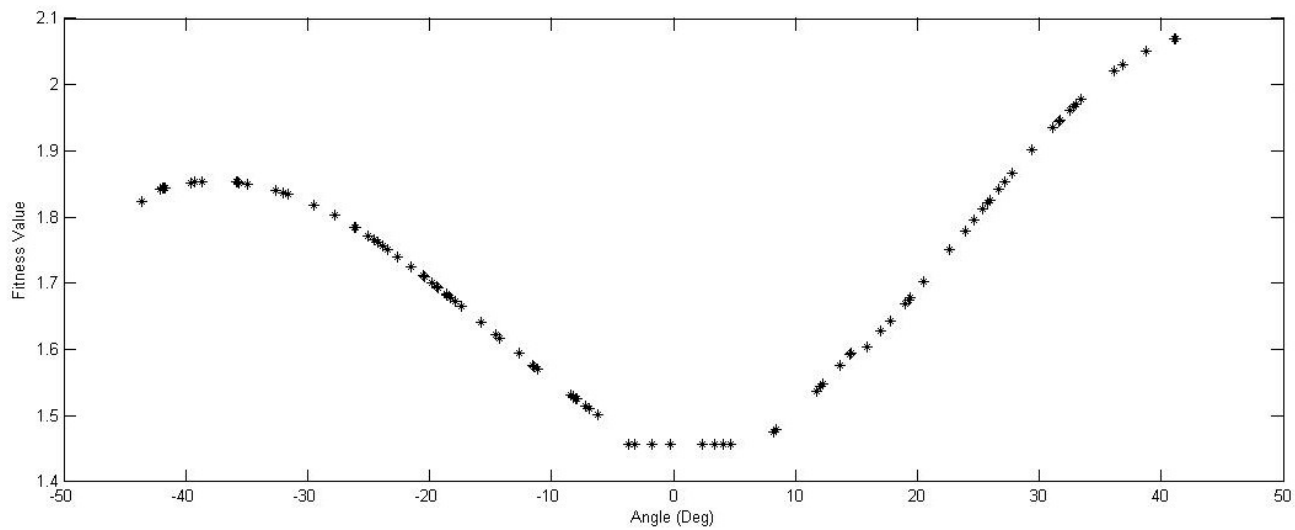


Figure 2.27: Fitness value vs. angle of the load vector.

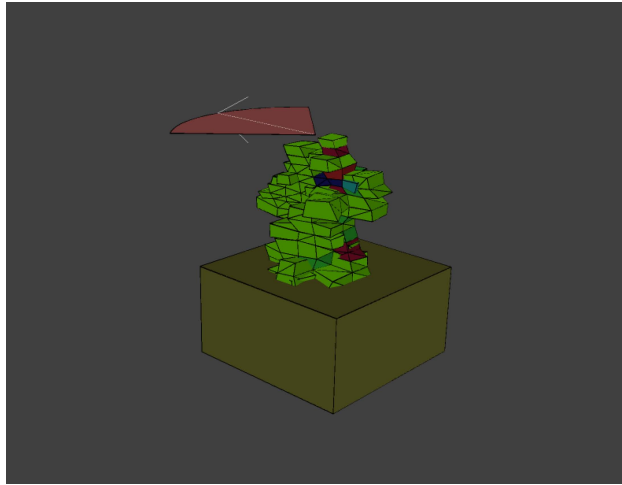


Figure 2.28: An evolved phenotype which has been generated by additional simulation.

## Chapter 3

# Asynchronous Scheme

The genetic algorithm in general uses a synchronize selection where the selection process is performed only after all the individuals are been evaluated prior to the mating scheme. In the direct encoding approach no development process is done and thus the variation in terms of evaluation between individuals is relatively small. In our model, the development process of every individual is highly computationally intensive. In addition, there is a high variation in the development-computational time between different individuals within the population. Using a synchronization scheme in our model resulted in a low computational efficiency since the individual with the slowest development process set the time of a single generation. Instead we have chosen to use an asynchronous evolutionary scheme which turns out to be much more computationally efficient and also increases the variability of the individuals during evolution, which eventually results in higher convergence rate. Unlike the synchronize scheme we do not wait until the last individual has been developed and evaluated. Instead, our only concern is to keep the nodes at a maximum capacity all the time. The number of nodes has to be smaller than the population size. For each time increment, an individual is been developed and evaluate on a single node, its fitness value is sent back to the master node. The master node will then send a new genotype for evaluation to the slave node. This process stops when there are no more genotypes to evaluate in the population. At this point the total number of individuals is given by,  $PopulationSize - NumberofNodes$ , where  $PopulationSize$  is the individuals that have been developed and evaluated and  $NumberofNodes$  is the individuals in the evaluation process. The synchronous scheme will keep waiting for the evaluation of the last individual among the remaining unevaluated individuals before producing a new offsprings and thus going through the next generation. In our asynchronous scheme we produce new offsprings from the subset of  $(PopulationSize - NumberofNodes)$  individuals, in what we defined as a "mating season". In this way, we generate a new population such that by the time one of the nodes



has finished evaluating an individual from the previous "season", a new individual from the current "mating season" will be sent to it for a new evaluation. The selection scheme during each "mating season" remains the same. This means that individuals with lower fitness values have higher a probability of being selected to produce offspring (roulette wheel).

## Chapter 4

# Inhomogeneous Structures under Wind Load

Inhomogeneous structures can be useful in many areas including optics, mechanics and thermal management, etc. In optics, for instance, several layers of thin films create an optical filter. Each layer has different properties which make the design of such filters highly complex. Different methods have been developed for the synthesis of such filters. J. Skaar [7] has shown a way to synthesize optical thin-film filters with inhomogeneous properties such that each layer in the film has a different number of reflectors. Yang and Kao [8] introduced a way to evolve the structure of a thin film with inhomogeneous optical coatings such that the evolved structure has the functionality of a beam splitter and a narrow-band reflector.

In addition to the large variety of applications, new techniques have been introduced with the ability to fabricate highly inhomogeneous structures. These methods include molding and micro-molding techniques [9] which enable inhomogeneity to be produced at the scale of a single micro-drop.

Our model provide the evolutionary process a tool to create inhomogeneous phenotypes, using the cell differentiation mechanism. In the following example, cells can be either made from steel or from aluminum.

### 4.1 Wind Load

Another important application for this method is to synthesize structures which are subjected to a wind load. From a computational standpoint we have try to decrease the computational resources and thus we have not modeled a full wind field. Instead we have used a rough approximation

which describes the behavior of a structure subjected to a wind as shown in equation 4.1. The parameter  $F_i$  represents the load on node  $i$ ,  $F_0$  is the total maximum load on the entire structure,  $S$  is the structure's volume and  $N$  is the number of cells. Equation 4.1 shows that an increase in the total volume will result in an increase of the total load in an exponential manner, this behavior roughly describes a structure which is subjected to a wind. All the other parameters such as, gravity evolutionary scheme remains the same.

$$F_i = \frac{F_0 \left(1 - \frac{1}{S^{1/3}}\right)}{N} \quad (4.1)$$

## 4.2 Results, Inhomogeneous Structures under Wind Load

Figure 4.1 and Figure 4.2 show four different views of two phenotypes that have been evolved for several hundreds of generations. The colors of the cells in Figure 4.1a and 4.2a represent the distribution of mechanical stresses inside the cells. Green represents low stress, graduates to red which represents high stresses. We can see that none of the phenotypes are over stressed. figures 4.1b and figure 4.2b show the cell's materials distribution in the phenotypes. We can see that both phenotypes are inhomogeneous such that there exists regions of adjacent aluminum cells and regions of adjacent steel cells. A distinction between these regions are presents in figure 4.1c, figure 4.1d and figures 4.2c, and figures 4.2d.

The difference between both phenotypes in terms of performances is small. Both phenotypes have lower stresses and are relatively light. They also grew to the desired height. From a topological view, both phenotypes are completely different. The phenotype in figure 4.1 looks similar to a bar. The bar contain cross-sections with areas that ranged from high to low, from bottom to the top, respectively. We can also see that the inner part of the bar is made from aluminum Figure 4.1c, while the outer part made from steel Figure 4.1c. The second phenotype in Figure 4.2 has a topology which is similar to a two piece arc. The arc has two regions, upper and lower. The upper region is made from aluminum Figure 4.2c and the lower region is made from steel Figure 4.2d.

The fact that both phenotypes contain regions of different materials instead of a random distribution is very encouraging. Phenotypes in nature share this type of property. Bone for instance has regions of materials with different mechanical properties, trabeculae in the center part and osteon in the outer part of the bone. A similar structure has also been seen in the phenotype in Figure 4.1.

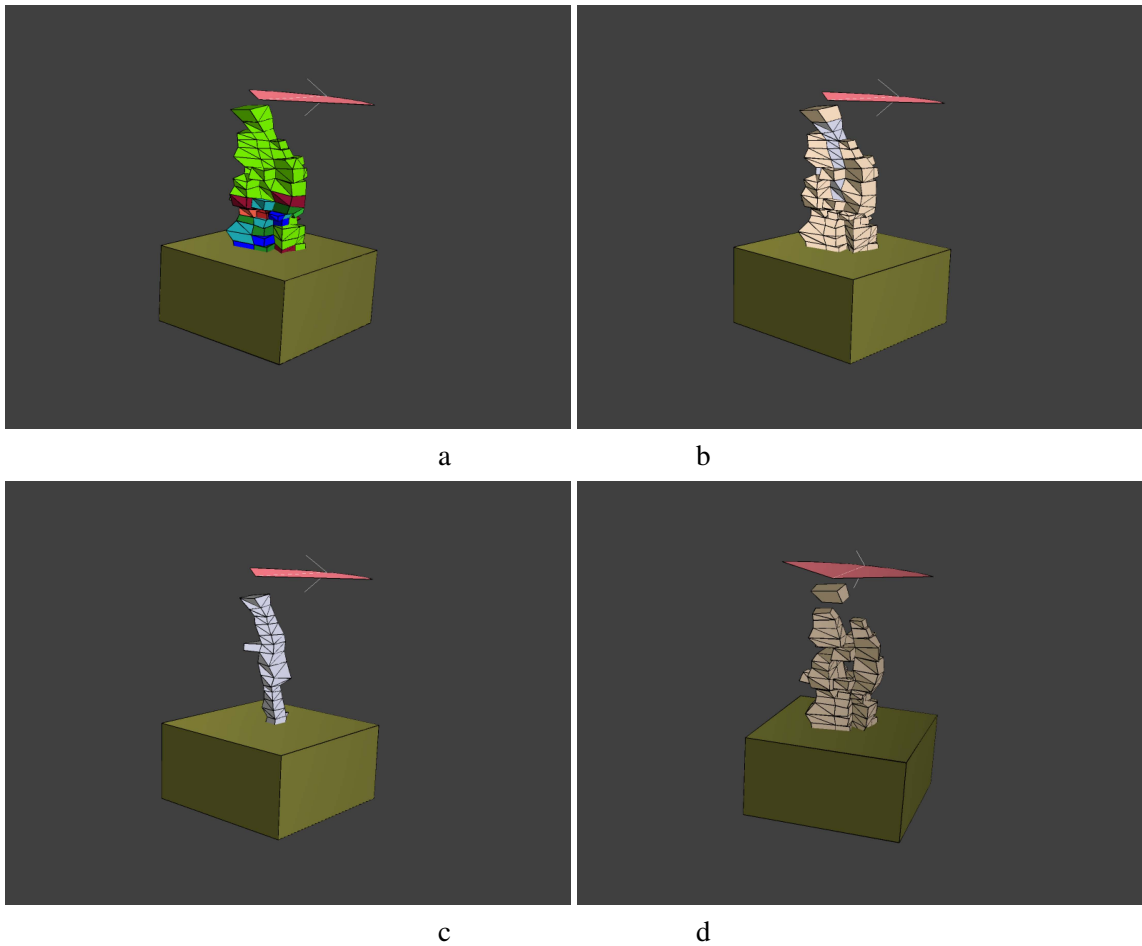


Figure 4.1: Inhomogeneous Structure, four different view of the phenotype. (a) colors represent mechanical stresses. (b) colors represents material, brown for steel , gray for aluminum. (c,d) The portion of cells that made from aluminum and steel respectively.

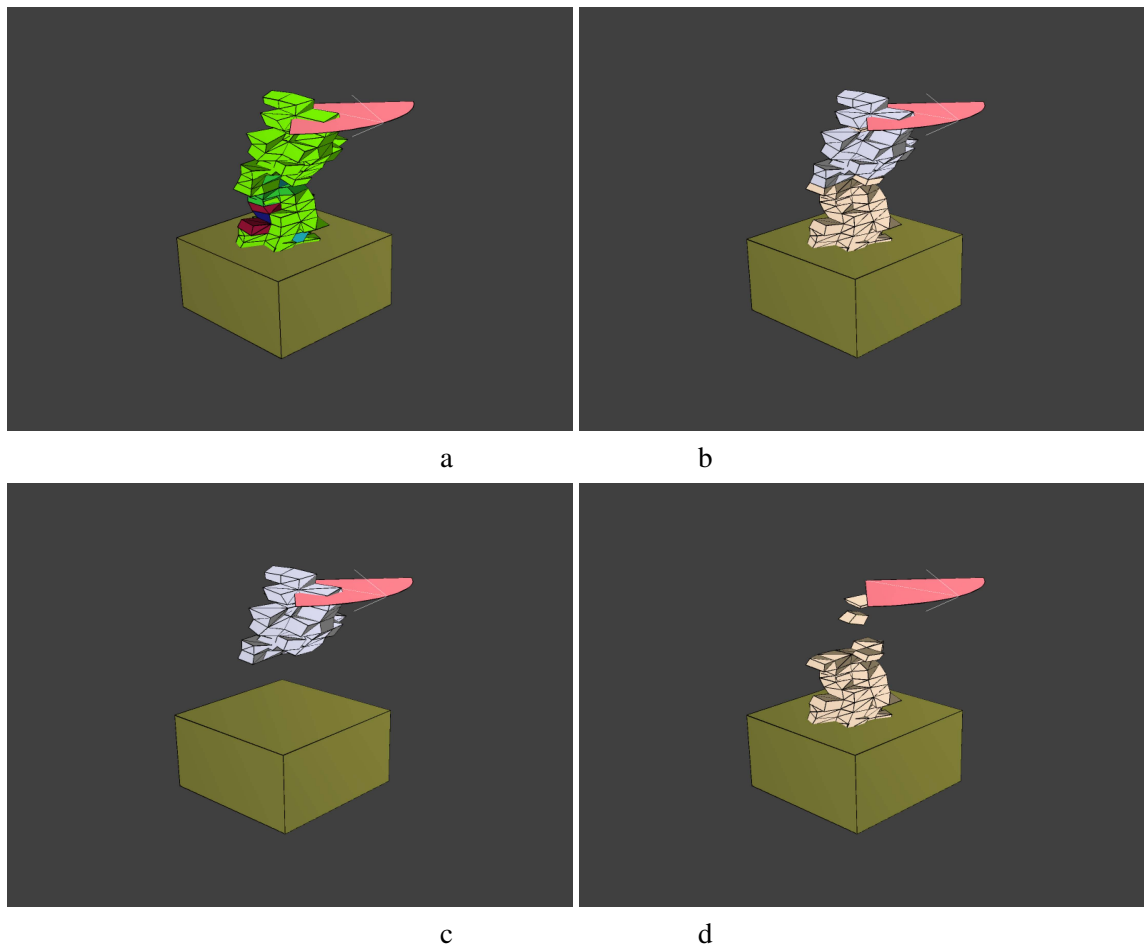


Figure 4.2: An Inhomogeneous Structure - four different view of the phenotype. (a) colors represent mechanical stresses; (b) colors represent material, brown for steel and gray for aluminum; (c) and (d) the portion of cells made from aluminum and steel respectively.

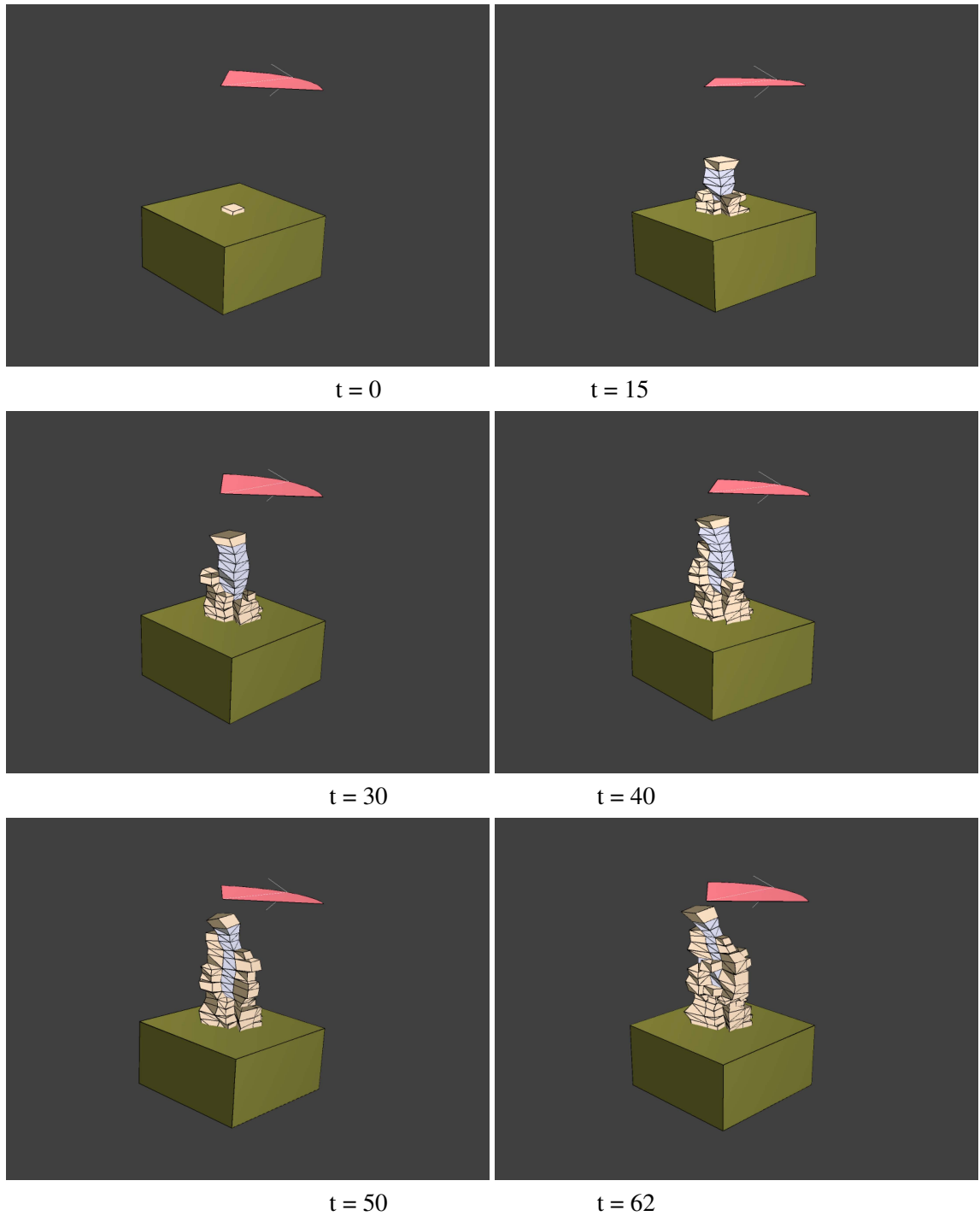


Figure 4.3: Growth process of the phenotype in Figure 4.1.

## Chapter 5

# Robustness

We present two different types of robustness mechanisms which have been spontaneously evolved *in silico* in a model of artificial embryogenesis. The first type provides robustness at the phenotype level, which is the ability of the phenotype to function in the presence of uncontrolled variations in the environment. The second type provides robustness of the growth process itself, which is the ability to grow a successful phenotype from a genome subject to environmental changes.

In our model, phenotypes grow in a stochastically changing environment from a simple set of rules (encoded in its genome) which are triggered by the environment, and have been evolved with evolutionary selection determined only by the fitness of the phenotypes to function in this environment. We show that our evolutionary process is capable of evolving genomes which are robust in both ways. These types of robustness are critical for evolutionary synthesis of novel phenotype configurations for two reasons. The environment, which serves as the trigger mechanism for regulation of the growth rules, changes significantly during the growth process. Additionally, the order of execution of the growth rules in a genome is random in our model.

The goal in this chapter has been to explore and understand the role of robustness in both the genotype and phenotype and their relationship to variations in the environment. In this section, we explore the correlation between the configuration of the phenotype and the environment. From a genotype standpoint we also investigate the action of genes during growth and present two kinds of experiments using the artificial evolution and embryogenesis described above. Each experiment simulates the evolution of a continuous structure using the described model. Each experiment evolved for approximately the same number of generations with the same environmental effects.

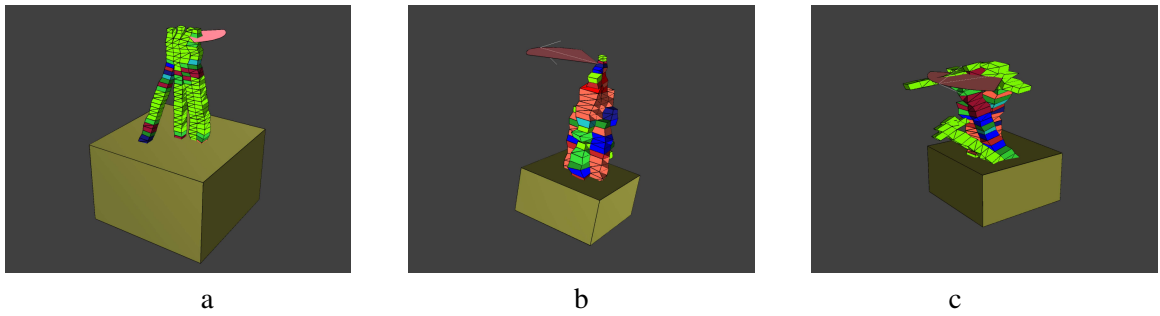


Figure 5.1: Three different phenotypes from three different stochastic environment experiments (SEE).

## 5.1 Results

The asynchronous evolutionary scheme was run with an initial population of 400 individuals which corresponds to 400 different randomly generated genotypes. All genetic operations (selection, crossover, and mutation) have been performed at the genotype level at the end of every generation or “mating season”.

In the first three experiments the environment had high variability. We call this kind of experiment a “Stochastic Environment Experiment” (SEE). We define the second kind of experiments as “Fixed Environment Experiment” (FEE). In this experiment the environment (the external load) is held fixed. In all the SEE experiments, the same highly variable environmental effects were applied, and the same evaluation scheme was used. In all experiments, the initial location of the load morphogen was 6 meters above the ground and the load was translated and scaled to the phenotype during growth, as described above.

Each simulation was run on 240 processors, taking several days for each run. The average number of generations was 500. Figure 5.1 shows three different phenotypes that were synthesized in the three SEE experiments under the same environmental conditions. The colors of the cells represents the mechanical stress, red for high stress through green for low stress. The pink region corresponds to the direction of the load variation. All phenotypes have a similarly low fitness values, representing high performance. Every simulation starts with a different randomly generated population of genotypes.



## 5.2 Family of Solutions

By observing the phenotypes from all SEE experiments, although the phenotypes themselves are topologically different, nevertheless we can still observe a “family resemblance” between them in terms of their configuration. Note that all phenotypes have a collection of cells at the top. These cells share the load between the legs, such that the load will be spatially distributed on the upper part of the structure. Additionally, a collection of supports (or legs), are present in the all four phenotypes. Such supporting legs are common in engineering structures because of their light weight and their ability to support variable loads.

## 5.3 Phenotype Robustness

The phenotype robustness is a direct outcome of the evaluation process which has been used in the evolutionary process. Every individual is evaluated only in its mature state. The evaluation is done with respect to several scenarios (in our model several type of loads) that are randomly generated. All three phenotypes in Figure 5.1 are robust to the environment since each is capable of supporting a high variance in the load, as indicates by the color of the cells. One of the major characteristics, common in their configuration, is support modules (distinct structural elements) which contribute to their robustness.

For instance, the phenotype in Figure 5.1a has four modules in the shape legs, three in-plane and one lateral. This configuration is good for two reasons: it maintains a light-weight structure, and supports a variable load. Although the other two phenotypes are different, both of them also have “leglike” structures, providing them with the same characteristics.

Next step is to consider whether these type of structural modules are a direct consequence of the evaluation scheme. A second experiment was performed, where the phenotypes were only challenged by an unvarying load in a single direction (FEE type), resulting in an environment which was held fixed at all times. The execution process however, remained the same, and thus produced a high degree of variability in the genome execution process.

Figure 5.2 shows the configuration of the best performing phenotype generated by the FEE experiment. It has one module in the shape of a thin structure oriented towards the load. We can see that the configuration of this phenotype is much more regular than the phenotypes presented in Figure 5.1. This indicates a strong correlation between the complexity of the configuration of

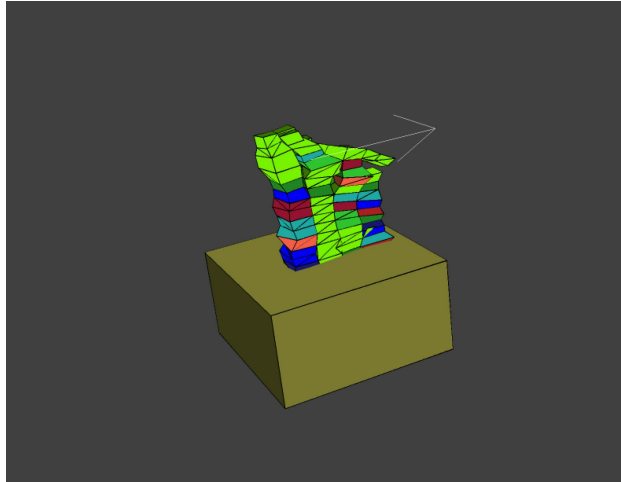


Figure 5.2: Phenotype evolved with an unvarying load (fixed environment).

the of phenotype and the required functionality. Phenotypes which were subjected to a higher load variability were much more complex and much more modular than those subjected to a fixed load.

In Figure 5.3 we show a comparison between the fitness convergence of both experiments. The  $y$  axis represents the fitness value of the best performing individual in a single generation while the  $x$  axis corresponds to the number of generations. The dotted lines represents the fitness running average of the fitness value. Based on this graph we see that the SEE converged after 250 generations while the FEE converges after 60 generations. It is also clear that the fitness values of the best performing individuals in the SEE experiment oscillate much more than in the FEE. figure 5.4 plots the standard deviation of both experiments with respect to their running mean value. Based on this plot we can see that the standard deviation of the SEE decays much more slowly than the FEE experiment, which indicates slower convergence. Both of these observations correlate strongly with the environment, which is the only distinction between the two experiments. Nevertheless, in both experiments a solution has been found which produces a low fitness value (high performance) through the rest of the evolutionary process, as shown by the stdv level. This results indicates that the evolutionary process synthesized solutions which overcome the high degree of stochastic variation in both the environment and the gene execution process.

## 5.4 Growth Robustness

As mentioned above, growth robustness corresponds to the ability of a phenotype to overcome variations in the environment during its growth process. In order to explore this behavior, the

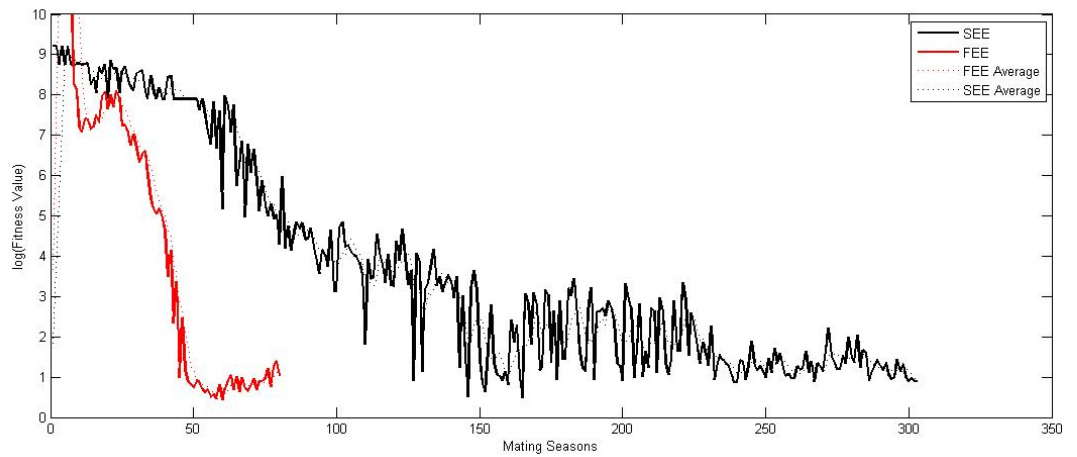


Figure 5.3: A comparison between two evolutionary process, stochastic (SEE) and fixed environment experiments (FEE), the  $y$  axis is the log of the fitness value, the  $x$  axis is the number of generations.

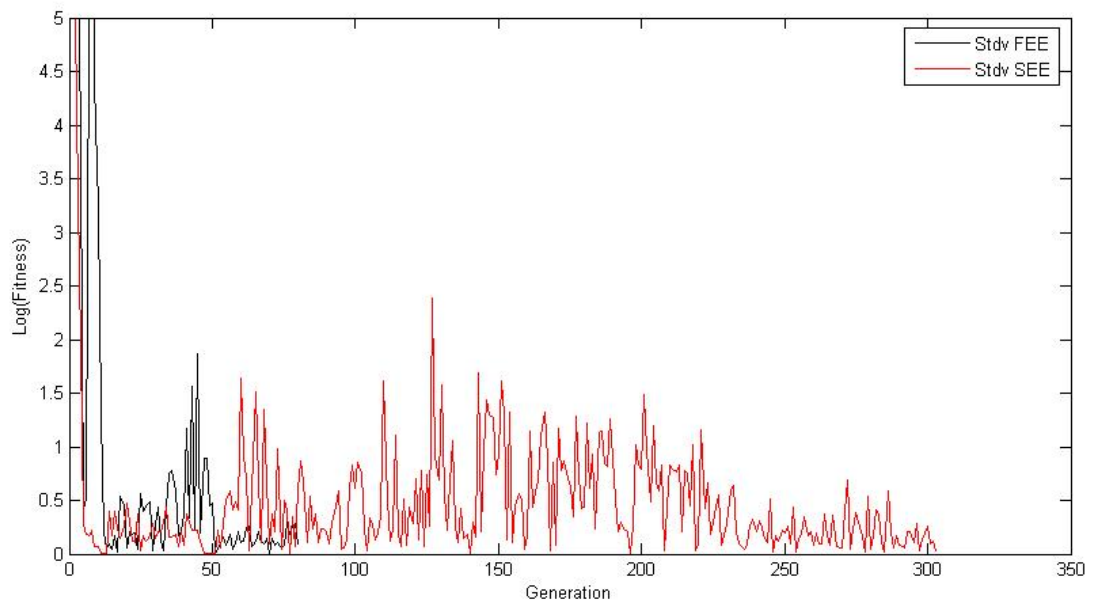


Figure 5.4: A comparison between the standard deviation of two evolutionary process in terms of standard deviation. Stochastic (SEE) and fixed environment experiments (FEE), the  $y$  axis is the stdv of the fitness value, the  $x$  axis is the number of generations.

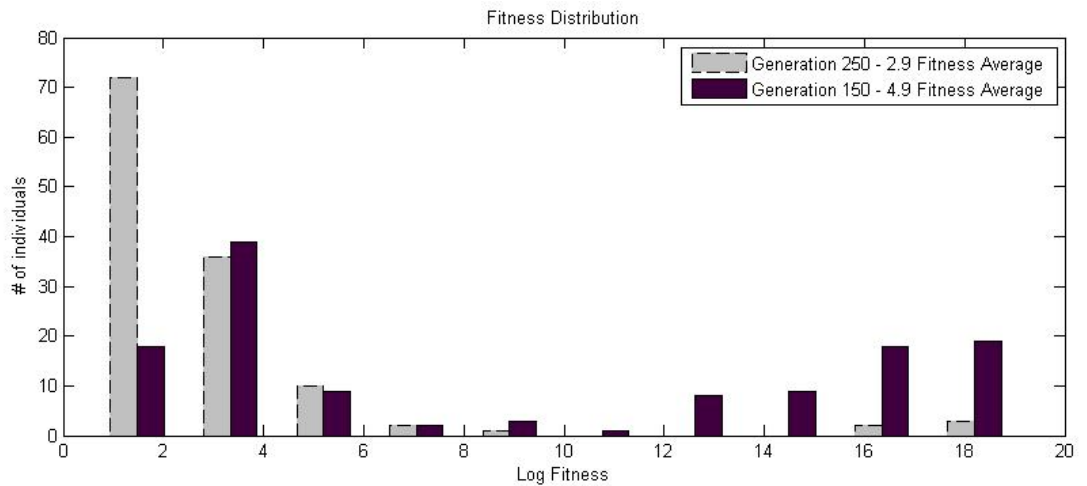


Figure 5.5: Fitness distribution of two genomes.

same genome, and thus the same set of rules, have been used to grow multiple phenotypes. The distribution of the the fitness values of all the phenotypes which have been grown from the same genome represents a measure of growth robustness.

For this purpose two genomes have been used, once from the midstage of evolution (generation 150) and one from the late stage of evolution (generation 250). Both genomes were taken from the SEE experiment. Figure 5.5 represents a plot of a fitness distribution of the two genomes. We can see that in late stages of evolution more highperformance phenotypes (low fitness value) have been developed from the same genome, changed from an average of 2.9 to 4.9 on the  $\log(\text{fitness})$  scale. Based these results, it is clear that the evolutionary process produces genomes which are robust to both the variations in the environment and to the variations in the gene/rule execution process.

## Chapter 6

# Word Duplication Mechanism, "WDM"

Every word in our model is a collection of genes. A gene cannot be changed through the course of evolution. Words on the other hand, are evolving through the kind and number of genes they are containing. A careful examination of the type of the words which evolve after hundreds of generation reveal that some of the words are repeated many times in the genome. This phenomenon is occurring spontaneously, due to the genetic operations of selection and cross over. If we assume that some words are much more viable to a phenotype than other words, then it is acceptable to think that some words will spread easily in the population through natural selection. This means that there is a high probability that two parents which contain identical words in their genomes, produce an offspring with two copies of the same words. In every generation, there is a probability to have genomes with more copies of a dominant words. In nature, the gene duplication mechanism has been associate with genetic robustness [70]. When several copies of the same gene exists in the same genome, this genome is more robust to mutation effects, since the last will only effect one copy of the gene. Nevertheless, this effect is only a secondorder effect, where the effect of gene duplication over the phenotype level is consider to be the first order. In other words, gene duplication may serve two purposes; it effects the growth process. This alters the topology and the functionality of the phenotype. It also makes the genome more robust to mutation effects. This argument stating that the effect of gene duplication at the phenotype level is much more crucial than at the genome level. Phenotypes that are affected by gene duplication in such a way will become more robust to the environment. They will tend to dominated other phenotypes which do not share the same robustness, regardless of their robustness to mutation. The same argument can go backwards, meaning phenotypes which are more robust to the environmental effects, which do not have a repeated mechanism will dominate phenotypes which are less robust to the environment and which contain duplicated genes. Early work by Susumu Ohno [71] argued that early vertebrates have a

complete genome duplication which transforms them into a modern vertebrate. His argument is that this mechanism has critically affected the phenotype, much more than the mutation effects. Another explanation which proceeded Ohno's hypothesis, correlates gene duplication with the degree of modularity of the phenotype. Wagner has proposed that a repeated execution of certain genes results in different modules at the phenotype level. He also argues that a high degree of modularity is related to the robustness of the phenotype [72].

## 6.1 The Effect of "WDM" on the Phenotypes

In order to examine this hypothesis two experiments were run. In both experiments the environment was the same uniform distribution of the load around at the load morphogen. The first experiment run with the word duplication mechanism ("WDM"). In the second experiment the "WDM" has been suppressed in the following way; by the end of every mating season, every identical copy of the same word has been removed. By the end of this process all genomes contains only a single copy of a particular word. Figure 6.1 shows both phenotypes from both experiments have converged. The convergence criterion was a solution that was close to '1' and which was stable through number of generations. The 'fourlegs' in figure 6.1a has evolved with the "WDM" and has a fitness value 2.15. The 'tripod' phenotype in figure 6.1b has been evolved without the "WDM" and has a fitness value of 1.47. In Figure 6.2 we show a comparison between the two experiments in terms of their convergence to the best solution. Based on this figure we see that the experiment with the non-"WDM" has converge significantly faster then the experiment which have included the "WDM". If we examine both phenotypes in figure 6.1 we can see a clear similarity between them. Both phenotypes have a large collection of cells at their upper part which enables them to reduce the load by distributing it over a large region. In addition to that both phenotypes have "legs" like structure which are surprisingly symmetric along the center point of the morphogen. The legs extend from the upper part of the phenotype to the ground in different planes, which help the phenotype to support a variational load. The legs are also relatively light as fitness function which required the evolved structures to be as light as possible. The conclusion from figure 6.1 and figure 6.2 is that the "WDM" did not have an effect on the topology of the phenotype, since we see similar patterns on both phenotype. From a convergence standpoint the "WDM" actually have slowed the convergence rate.

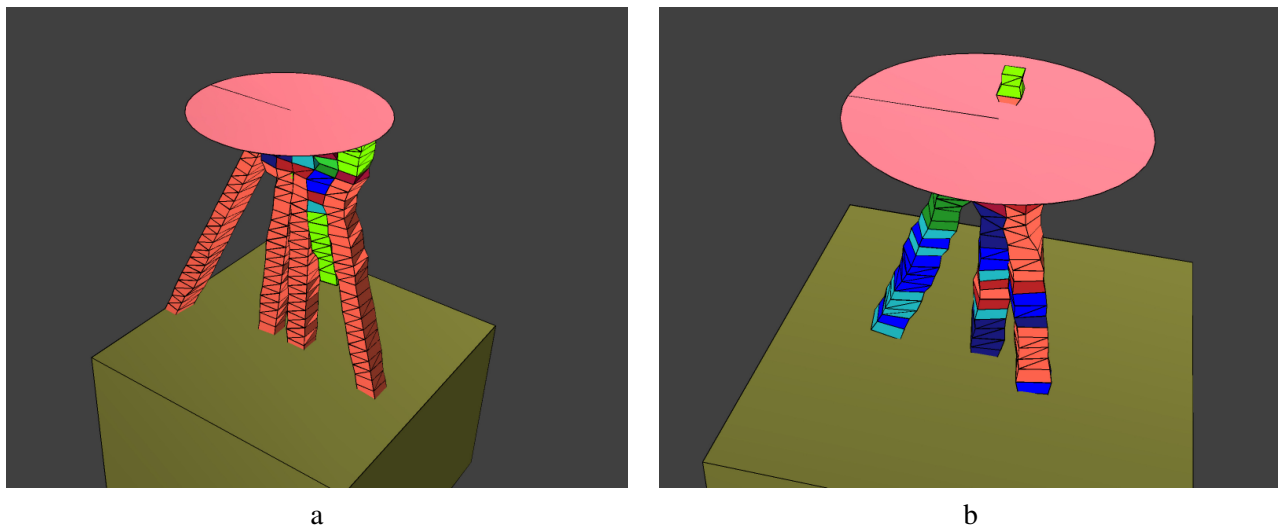


Figure 6.1: "Best" Phenotype from both experiments. (a) 'fourlegs' phenotype include "WDM". (b). 'tripod' phenotype with non-"WDM" .

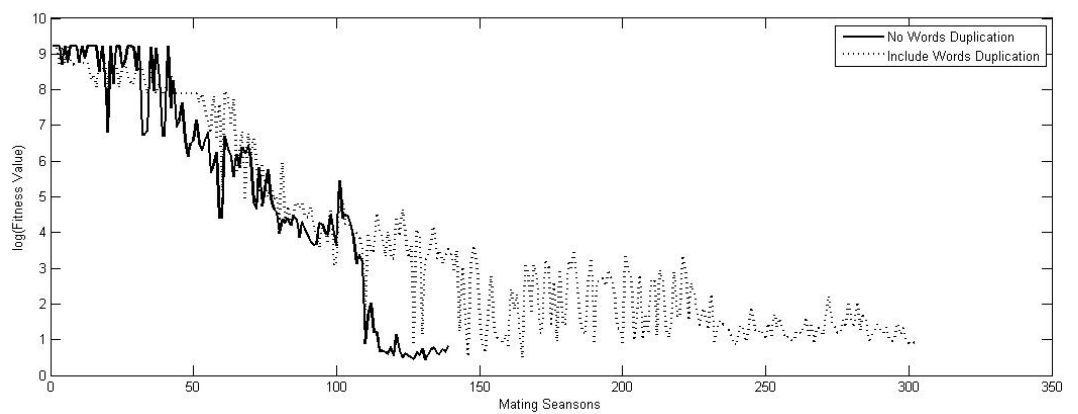


Figure 6.2: A convergence comparison of both experiments.

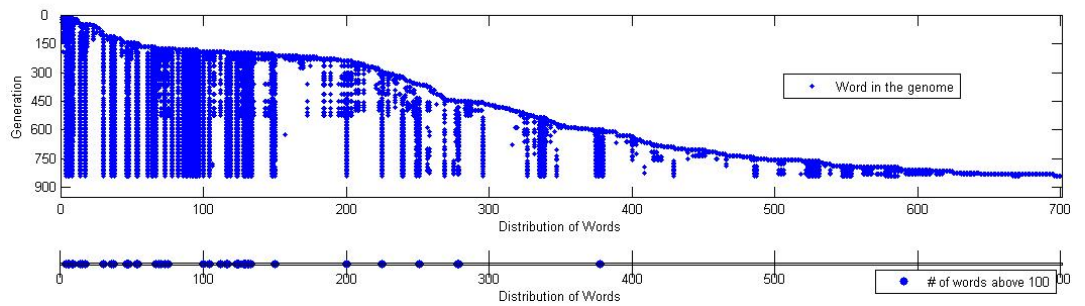


Figure 6.3: The evolution of new words during evolution.

## 6.2 The Relation Between "WDM" and the Landscape of Solutions

Figure 6.3 shows the evolution of new words during the evolutionary process of the phenotype in figure 6.1a. Every point on this graph represents a word. The x axis represents the different words where the scale of this axes is just an index number (i.e. through the course of evolution there were only 700 different words on all phenotypes which dominated their generation). The y axis corresponds to the generation. Every line, parallel to the x axis in figure 6.3, corresponds to a genome of a certain generation. Every vertical line represents a word that has been preserved through evolution. The starting point and the end point of this line correspond to the first generation where this word had been produced and the last generation where no copies of it remain in the population. Therefore, words which were kept through the entire course of evolution exist on the dominant phenotype in the final generation. The second part of this graph represents words which have more than 100 copies of those in the genome. We can see that most of the initial words, which have been produced in the early stage of evolution and which have been preserved in the next generation, are the ones which have the most number of copies in the genome. This fact supports our first hypothesis; words which were contained in dominant phenotypes, which had higher probability of selection, produce offspring which have several copies of identical words through the crossover operation.

Our hypothesis is that as the number of repeated words in the genome increase, the landscape of new explored solutions is in fact decreased, which explains redundant convergence behavior in Figure 6.2. From a mathematical standpoint we can argue that indeed a highly repeated number of



words decreases the probability of exploring new solutions, although it makes the current solution very robust to mutation. The mutation operation can mutate a single word with equal probability between all the words in the genome. If a particular word has many identical copies of it in the genome, there is a high probability that this word will be mutated instead of other words. Mutation of this word will have a low chance of damaging the phenotype since there are still many copies of it as backup. The only chance it will drastically effect the phenotype, is if it will create a different regulation effect which will affect most of the other words in the genome. The chances for this to occur decrease with the number of generations, since the phenotypes have developed a strong robustness mechanism against mutation by having many copies of identical words.

### **6.3 Genes/Word Regulations - "WDM" and Non-"WDM"**

The next step is to explore the expression of words during the growth process and to compare the way words are being expressed in both the "WDM" and the non-"WDM". Although the set rules which are encoded in the genome is relatively simple, the regulatory network which represents the cross-effects of genes over other genes, is extremely complicated. The reason is that genes may affect both other genes within the same cell, and also genes in adjacent cells. In addition, the environment, which is changing randomly, is one of the trigger mechanisms of genes. Since words themselves are an evolvable unit which contains a sequence of rules, and since some words are much more dominant than other words, we established a mechanism which enabled us to monitor word expression during the growth process. Most of the words contain control/switch genes, which serve as precondition of the sequence of operations, written inside the word. A very common scenario will be a word that never gets to be expressed since the preconditions are never satisfied. Thus, this word can be removed from the genome and it will not effect the growth process. In our wordmonitoring scheme, we count the total number of times a particular word has been expressed for an operation, i.e. cell division, protein generation, cell differentiation etc. All the cells at all time frames during the growth process. This method provide us with an ability to compare words that have been expressed many time to words that have been expressed fewer times.

#### **6.3.1 Words Distribution in Non-"WDM"**

We first analyze the genome of the 'tripod' phenotype, shown in figure 6.1a which has been evolved with non-"WDM". In order to have conclusive confidence about the distribution word ex-

pression, we always run the growth process twice to make sure same words have been used during the growth process. The reason is that the environment is constantly changing, and thus one can imagine two growth process, originating from the same genome, having a different word distribution. Of course we have shown strong evidence of the growth robustness which overcomes these stochastic effects, and thus it will be clear that no significant changes can be observed between two growth processes originating from the same genome. Figure 6.4 shows the distribution of words between two growth processes, indicated by Exp1 and Exp2 respectively. The x axis represents the index of all words in the genome (48 different words). The y axis corresponds to the total number of times a particular word has been expressed during the growth process. Figure 6.4 gives a partial answer to the evolution of growth robustness. The average difference of the same words expressed between Exp1 and Exp2 is 12.5, There are also now words which have been expressed in Exp1 and not in Exp2 and vice versa. This indicates that although both phenotypes have been growing under a stochastic environment, they both have expressed the same subset of words an almost identical number of times. Also we notice that words 1, 2, 5, 8, 16, 17, 32, 35, 38 have been expressed a similar number of times, word 4 has been expressed the largest number of times and word 9 has been expressed the least number of times. Figure 6.5 shows the preservation of words through evolution, where every column correspond to a different word. We can see that at some point during the evolutionary process (generation 70), the genome of the best individual was comprised of new words only. From this generation forward, none of the old words have ever been used again. This fact suggested that the indirect encoding using the non-”WDM” is capable of either using old words (toolkit) on top of new ones, or reinventing new words from scratch. It also demonstrates the ability of the method to escape local minima very efficiently if nothing of what is already available can be used for finding a new solution, a new toolkit will eventually be generated.

### 6.3.2 Words Distribution include ”WDM”

The same analysis has been done for the ”WDM” case. Figures 6.6a and b show the words expression of the fourlegs phenotype of Figure 6.1b where Figure 6.6b is simply a change in scale of Figure 6.6a. We can see clearly from this figure that word number 12 has been expressed a thousands times more then the other words. The words which have been expressed during the growth process are shown in Table 6.2. We can see that unlike the non-”WDM” case, where executing the growth process twice results in the same expression of words, in the non-”WDM” case, during the second execution of the growth process, a small number of new words have been expressed which

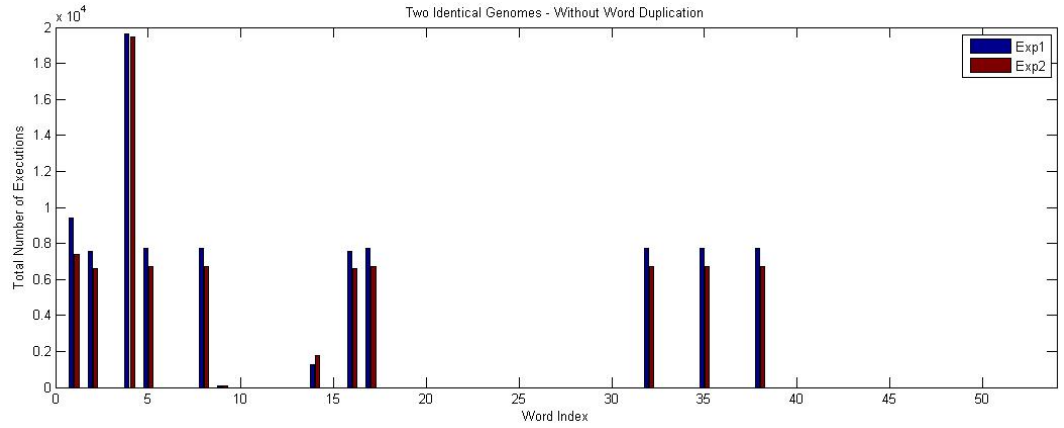


Figure 6.4: Non-“WDM”, the distribution of words expression.

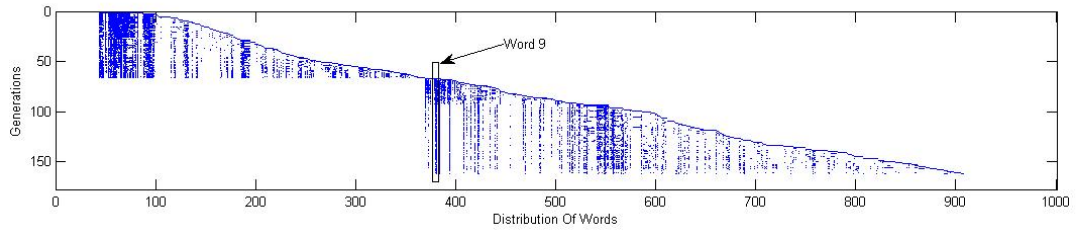


Figure 6.5: The evolution of new words during evolution.

Table 6.1: Non-“WDM” Active words

Word	Index
<i>R1Z1V160g158hS3gA0p24Dp03p15P22p13P14p02P2p22</i>	1
<i>R1Z1C9hV135h106hC8g</i>	2
<i>R1Z1V112a155tP2p00P5p25P36p00P36p22A0p14A2p11P9p00C1t</i>	4
<i>R1Z1P14p04V129h134hP47p25F1</i>	8
<i>R1Z8V73g199gDp04V47h54h</i>	9
<i>R1Z1P42p25V159h53tC3hP17p25P38p24P15p24P38p22P23p01F1P4p13</i>	16
<i>R1Z3V66h78hV112a36hA0p02P21p10</i>	17
<i>R1Z1V199h137tV113a68hP7p02</i>	32
<i>R1Z1V69h72hV190a187aP13p23S2hP50p02C9aP28p00P26p12P34p01P37p12C8h</i>	35
<i>R1Z1V140h178hV19h155hP3p10P41p22P39p23S4aDp21p15p23P47p05P50p22P23p02P47p21</i>	38

have not been expressed previously. This supports our hypothesis, which claims that the non-”WDM” case was able to evolve genomes which were more robust in their growth process, compared with the ”WDM” case. Figure 6.7a shows the number of times each word in the genome is repeated. The y scale is a  $\log_{10}$  scale. Figure 6.7b shows both the number of times a word appears in the genome and the number of times it has been used during the growth process. Based on Figures 6.7a and b we first learn that the number of repeated words in the genome is nonuniform. In fact, word number 12 has appeared 110,944!! times in the genome, which corresponds to 96 percent of the total length of the genome. This means that any mutation will most likely mutate a single copy of this word. This word domination phenomena will eventually limit the ability to search for better solutions, and will cause the evolution to be stuck at a local minimum. From Figure 6.7b we learn that there is a high correlation between the number of times a word is repeated in the genome and the number of times it has been executed. The last statement is not obvious given the fact that we only present words that execute an operation, so one can have many copies of words in the genome which will never be expressed. This type of correlation can be explained by the evolutionary process and in particular by natural selection. Words that get to be expressed during the growth process are affecting the topology of the phenotype in some way. If these effects increase the performance of the phenotype, then these words become a crucial element in the performance of the phenotype. High-performance phenotypes will have a high probability of being selected, and thus crucial words will have a high probability of being preserved in the next generation. From this point, the cross-over operation will create new genomes with multiple copies of the crucial words which will eventually accumulate.

## 6.4 Exploring the Effect of Words on the Phenotype Topology

The growth process itself, which executes the genome and turns it into a mature phenotype, is very complex. In the previous section we have shown the number of times each word has been expressed. The direct effect of words on the phenotype is almost impossible to predict. In order to get some understanding of how words affect the phenotype, we have performed an experiment where we removed a single active word (words which were actually expressed during the growth process) and executed the growth process dozens of times and plotted the fitness value. Again, the reason for a repeated execution is to look at the distribution of the phenotypes among a big population of phenotypes that has been grown with the same genome. Figure 6.8 shows the effect of word elimination over the fitness value of the phenotype for the non-”WDM” case. Since in the

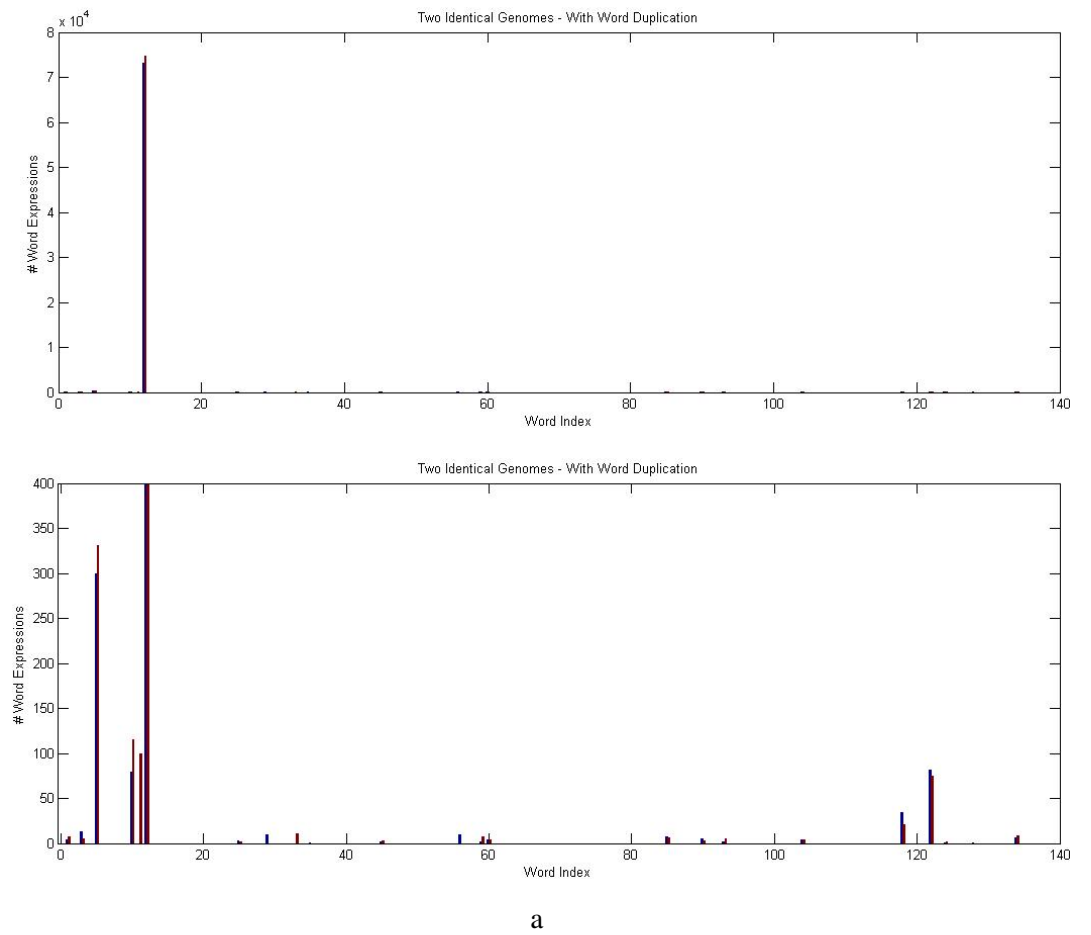


Figure 6.6: "WDM", the distribution of words expression.

Table 6.2: Non-“WDM” Active words

<b>Word</b>	<b>Index</b>
<i>R1Z1A2iV69h182hV79a63aKS3hP7p11S10hDp00ip20</i>	1
<i>R1Z3P4p10A1p05P5p13P13p15P37p15</i>	3
<i>R11Z1P41p04A0p12P15p00P34p01P27p20P14p13P22p24</i>	5
<i>R110944Z1V39h185gC8hP25p15S4gP48p10P43p01C5hP48p23P27p12P49p21C2h</i>	12
<i>R1Z1V39h185gC8hP25p15S4gP48p10P39p22C5hP48p23P27p12P49p21C2h</i>	24
<i>R1Z1A4p05V93h153h</i>	25
<i>R1Z1V39h185gC8hP25p15S4gP48p10P43p01C5hP48p23P27p12P18p01C2h</i>	32
<i>R2Z1V39h185gC8hP25p15S4gS1hP48p10P43p01C5hP48p23P27p12P49p21C2h</i>	33
<i>R1Z1V39h185gC8hP25p15S4gP48p10P43p01P25p03C5hP48p23P27p12P49p21C2h</i>	41
<i>R1Z1P20p24V199t44a</i>	45
<i>R1Z1A2iV69h182hV79a63aKS3hP7p11S10hDp00ip20</i>	1
<i>R3Z1V39h185gC8hP25p15S4gP48p10P43p01P48p23P27p12P49p21C2h</i>	56
<i>R1Z1P37p15V136h109aA1p24KP19p13A1p23Dip13p04</i>	59
<i>R1Z1P17p12V186t165tP39p13</i>	60
<i>R1Z1V39h185gC8hP25p15P48p10P43p01C5hP48p23P27p12P49p21C2h</i>	70
<i>R1Z1V39h185gC8hP25p15S4gP48p10P43p01C5hP48p23P27p12P38p03P49p21C2h</i>	84
<i>R1Z1C1hV130h129hKP46p25A0p15KC8tC10aP4p00P39p02S8h</i>	85
<i>R1Z1F0V178a61h</i>	90
<i>R1Z1V39h185gC8hP25p15S4gP48p10P43p01C5hP48p23P27p12P26p12C2h</i>	91
<i>R1Z1V39h185gC8hP25p15S4gP48p10P43p01C5hP48p23P27p12P49p21P45p11C2h</i>	94
<i>R1Z1V39h185gC8hP25p15S4gP48p10P6p15P43p01C5hP48p23P27p12P49p21C2h</i>	98
<i>RR1Z1A0p01V115h60gP40p02P50p25P12p04P8p23Dp25p14P45p04P33p10A4p20 Dp04p20</i>	104
<i>R1Z1V39h185gC8hP25p15S4gP48p10C5hP48p23P27p12P49p21C2h</i>	110
<i>R30Z1V128a56hA0p03P40p20P43p04P16p05Dp01P44p01P12p10A1p02P2p05</i>	118
<i>R18Z1F0V94g84h</i>	122
<i>R1Z1V9t133tF0V94g84h</i>	124
<i>R1Z1V124a188aV2a144hP24p15</i>	128
<i>R1Z1V120g189hS7hP47p04</i>	134

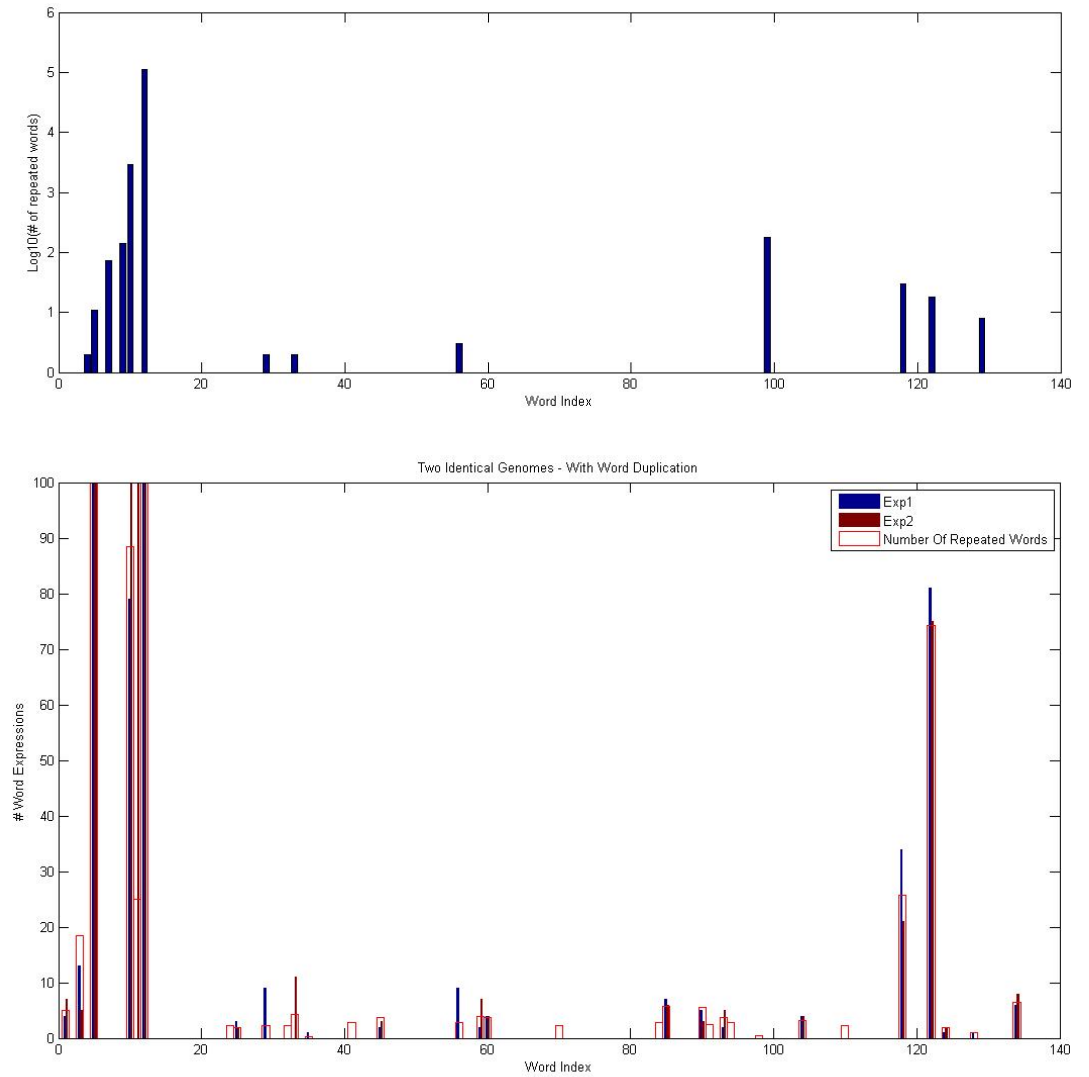


Figure 6.7: "WDM", repeated times in the genome vs number of executions. (a). "WDM", The number of times words appear in the genome on a  $\log_{10}$  scale. (b.) Repeated time of the words Vs number of executions.

"WDM" case, we saw that different words have been expressed in different growth process, we will not be able to make a strong conclusion for the effect of eliminating words. Every bar on figure 6.8 corresponds to a different word that has been removed from the genome. Every genome has grown 40 phenotypes and their  $\log_{10}$  distribution is shown on the x axis. The y axis corresponds to the number of phenotypes. The blue color corresponds to the case where no words have been removed from the genome. We can see that most of the phenotypes that have been grown from this genome have low fitness values which means that they performed well under the environment and their growth process was highly robust to the environment. By removing words 4, 5 and 8 we see that some of the genomes are still developed in phenotypes with high performance, although the distribution of those phenotype are in fact reduce. In Figures 6.9c,d and e we can view the phenotypes which correspond to the elimination of words 4, 5 and 8. Viewing these phenotypes show that their topology is similar to the 'tripod' phenotype. On the other hand, removing words 8 and 9 has completely damaged the growth process such that the phenotypes were incapable of growing, as shown in Figures 6.9b and f. Removing word 1 however, has reduced the performance level of the phenotype, but at the same time generated a new topology, shown in Figure 6.9a. Based on this analysis we see that some words are more crucial then others, so removing some words will barely affect the phenotype while removing other words can completely ruin the growth process. Also, removing a word can alter the growth process such that a new configuration will be developed. Comparing Figure 6.4 to Figure 6.8 we see that there are is correlation between the number of times a particular word was been expressed and the necessity of this word during the growth process. As an example, we observe word number 9 which has been expressed the least number of times during the growth model compared to all of the other words, as shown in Figure 6.4. Nevertheless, the elimination of this word had the most destructive effect on the genome, indicated by Figure 6.8. In order to understand why some words are more dominant than others we should first look at the growth process of the 'tripod' phenotype. Figure 6.10 shows the six stages during the growth process of the 'tripod'. We first see that it starts as a single cell subjected to a high mechanical stress, indicated by the red color. At 50 sec it grows half way to the load morphogen, and in 85sec it reaches it. Then it starts producing additional, symmetric legs which go into the ground (150 sec and 100 sec). From there on it remains stable, which indicates that it reached a maturity state. Observing these stages, we see that in the first step, a straight rod has grown to the load morphogen. Looking closely into word number 9, we find the following set of rules *R1Z8V73g199gDp04V47h54h*. The first rule is a control mechanism which enables the execution of the other rules if the morphogen



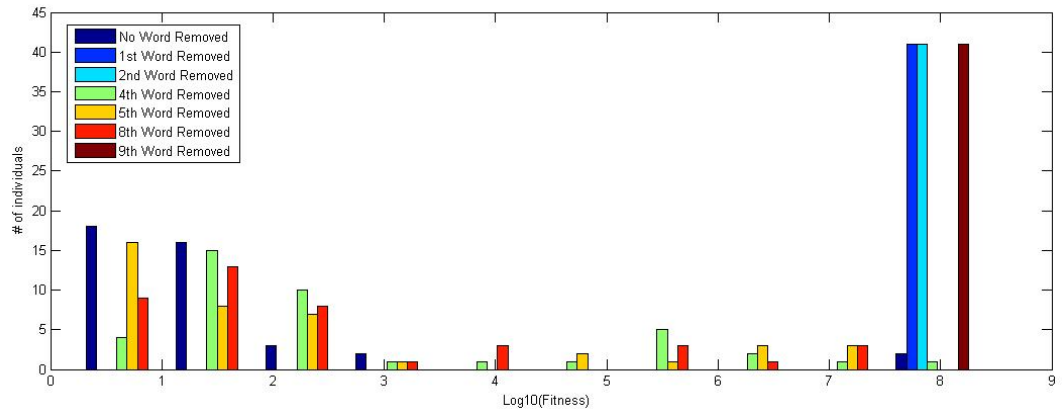


Figure 6.8: The effect of word elimination on the fitness distribution.

intensity is within 73 to 200 percent. If that condition is satisfied this word will produce cells in the north direction in the presence of protein 0. So if we remove this word, no new cells will grow to the load morphogen and we will be stuck with an "ungrown" structure as shown in Figure 6.9f. Protein  $p_0$  can be either produced by word 8 or word 9, which explains why removing either one of those can still result in a successful phenotype.

## 6.5 The Evolution of Simple Modules

In this section we would like to explore the evolution of the 'tripod' phenotype. We already saw that word 9 was very significant to the success of the phenotype since it creates a long base structure that starts from the bottom and grows toward the load morphogen. Looking at Figure 6.5 we see that word 9 was first discovered in generation 71. From generation 71 word 9 has been preserved through the entire course of evolution. In Figure 6.11 we show the topology of the most primitive phenotype which had contained this word. This phenotype is not able to sustain the load, indicated by the red cells which are over stressed. However, this phenotype was indeed capable of growing high enough to reach the load morphogen and had a symmetric wide base which helps in distributing the high bending stress across a large number of cells. The obvious common module for the tripod phenotype and the primitive one is that both of them contain a long thin rodlike structure starting from the ground and ending at the load morphogen. This observation suggested that evolution had first found primitive forms such as simple rod structures and then combined those into a more complex structure such as the tripod. This kind of behavior had been observed in fossils where some toolkit genes that were first found in ancient vertebrates, which had a very primitive form, had been

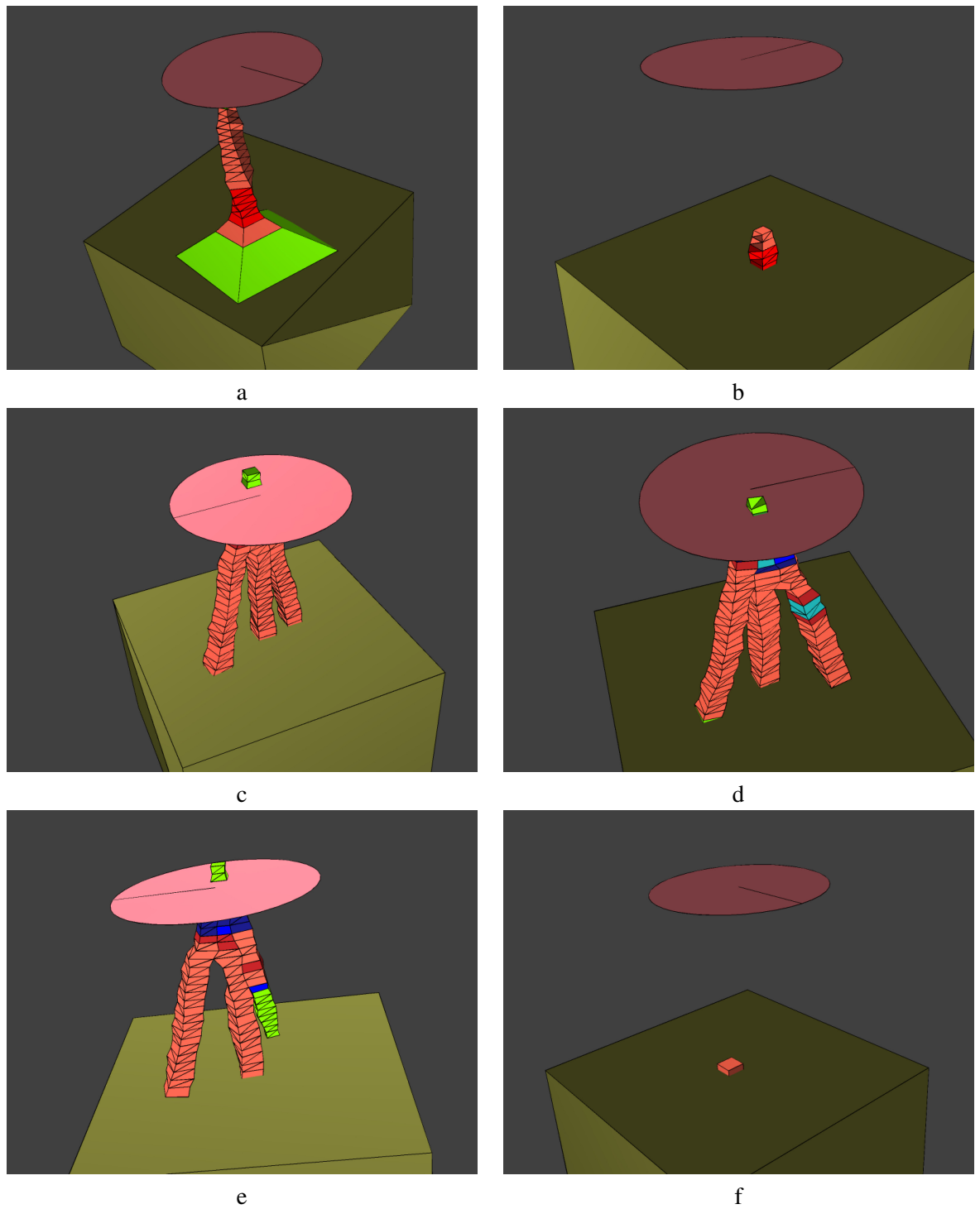


Figure 6.9: Phenotype resulting from word elimination. (a). 1st word removed. (b). 2nd word removed. (c). 4th word removed. (d). 5th word removed. (e). 8th word removed. (f). 9th word removed.

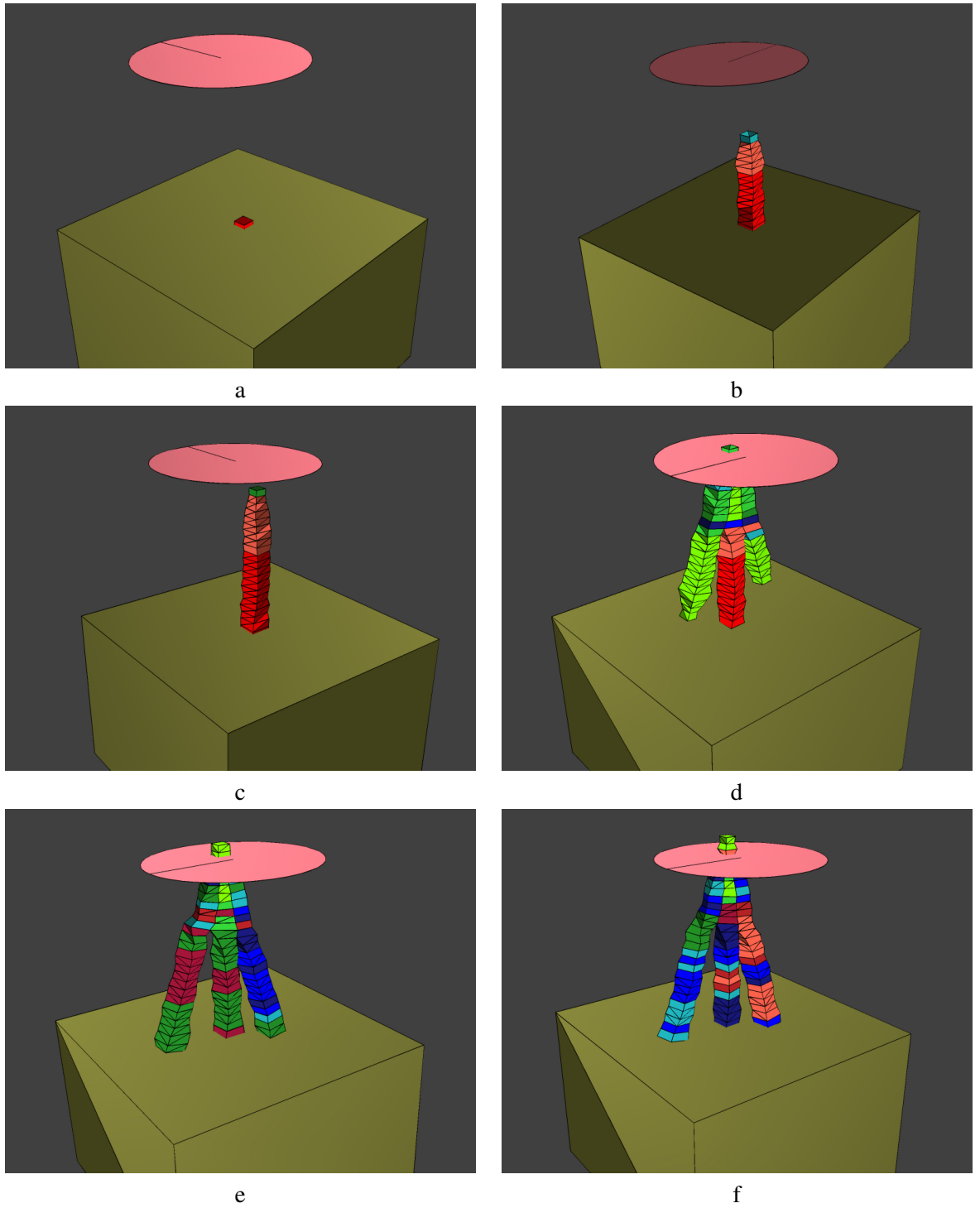


Figure 6.10: Six stages in the growth and development of the 'tripod' phenotype .

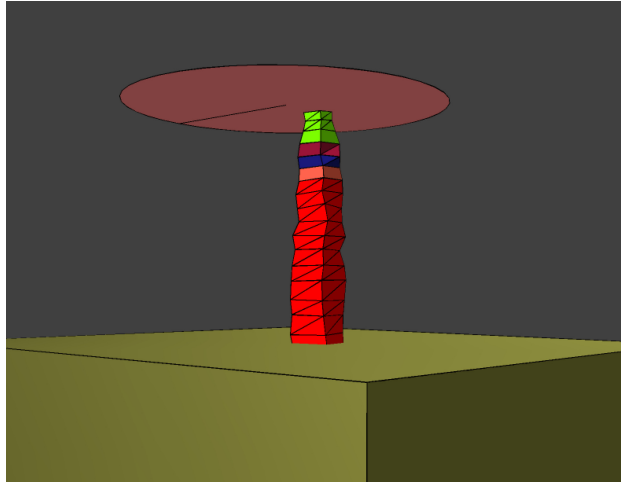


Figure 6.11: The first primitive phenotype, contains word 9.

found again in more modern vertebrates [1]. This kind of repeated use of the same toolkit gene in other creatures or forms is the consequence of the development process which is guided by a simple set of rules. It also shows the strength of our evolutionary scheme in finding new solutions utilizing old words.

### 6.5.1 The Effect of a Nonsymmetric Environment

The environment in our model is the essential component which drives both the evolution the growth process. Modification of the environment results in a different growth process and topology of the phenotypes. We have shown two types of loads; concentrated load and wind load. In all the simulations so far the load was symmetric and thus the environment was consider to be symmetrical too. In this section we will examine the effects of nonsymmetric environment over the topology of an evolved phenotype. Figure 6.12 shows several phenotypes, resulted from a various types of environment and loads. Figure 6.12a shows a phenotype that has been evolved under a nonsymmetric environment where the load was a concentrated load. It has three modules, the first module in the front of the structure is tilted toward the load, while the other two module are on the back of it. All three modules are connected between themselves, so that the middle part of the structure is hollow. This structure is much less symmetric, comparing to the 'tripod' structure. Figure 6.12b shows a phenotype which has been evolved under a constant environment, thus one directional load does not vary in time. The topology of this phenotype is obvious, a thin wall pointed to the direction of the load. This structure is very superior in supporting constant load but will fail if the load changes.

Figure 6.12c and figure 6.12d show two phenotypes which have been evolved separately under a symmetric wind load which has been applied in all directions (Figure 6.12c) and on a more narrow region (Figure 6.12d). We can see that both these phenotypes have a similar symmetric structure. The last phenotype which is shown in Figure 6.12e has evolved under a nonsymmetric wind load environment. This phenotype has a shell like structure which increases its stiffness in supporting a bending load.

Observing all of these phenotypes, we can conclude that the environment has a crucial effect on the topology of the phenotype. We can argue that if the environment is symmetric, most likely the evolved phenotypes will be highly symmetric too. Also if the type of the environment changes (concentrate vs wind load), the structural pattern will change as well. We have seen evidence showing that evolving phenotypes under a nonsymmetric environment, result in less symmetric phenotypes than were evolved under a symmetric environment. Nevertheless, even in the case of a nonsymmetric environment, we have witness modularity and a small degree of symmetry which indicates that there might be additional factors beside the environment, that creates symmetric and modular phenotypes.

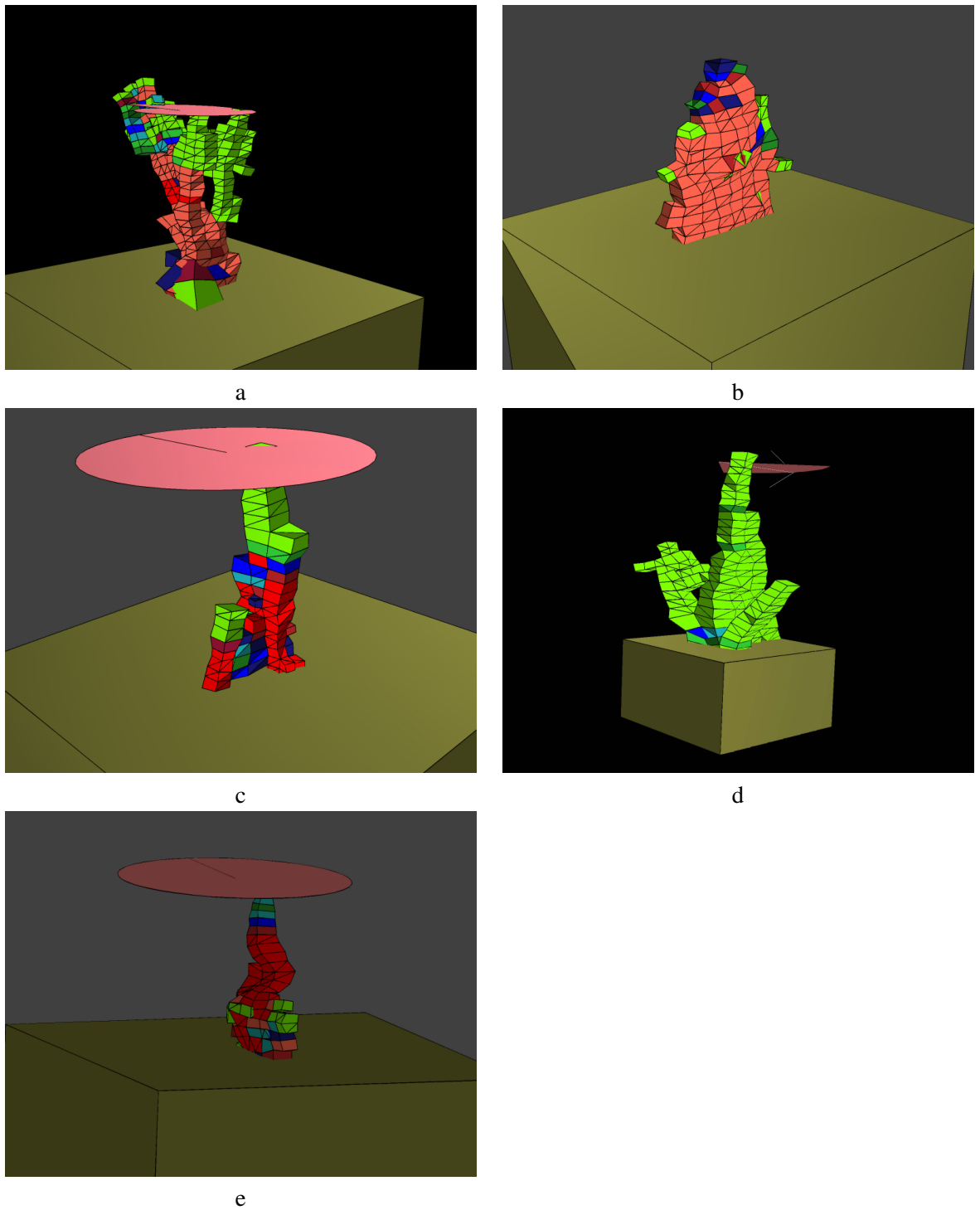


Figure 6.12: The effect of the environment over the topology of the phenotype. a. Concentrated load, non-symmetric environment b. Concentrated load, constant environment. c,d. Two phenotypes - wind load, symmetric environment. e. Wind load - nonsymmetric environment.

## Chapter 7

# SelfRecovery Mechanism

One of the greatest abilities which exists in nature and is lacking in almost all artificial machines is the ability of phenotypes to recover from external damage. All phenotypes in nature have some ability to recover themselves from a body deterioration, cause by some environmental effects. This kind of behavior is directly related to the survival and thus to natural selection. Lizards for example, may regrow their tail in case it was removed by a predator. Here we have tried to reproduce this kind of behavior in our threedimensional structures. The idea was to creates an artificial damage to the phenotype during the growth process and to observe whether it will have an ability to recover from it. The strategy we have taken is to randomly removed a cluster of cells from the phenotype at a random location. Once the cells have been removed, the growth process continues as before until the phenotype reached maturity. At the maturity state, the phenotype is then evaluated. The evolutionary challenge, in this case, is to find solutions (genomes) which will overcome this random cell removal by recovering themselves. Figure 7.1 shows the growth process of a phenotype which has been evolved with the new evolutionary scheme. Figure 7.1b shows the phenotype before the cells have been removed. Figure 7.1c and d show the phenotype after a cluster of cells have been removed from it, indicated by a big hole in the middle. Figure 7.1e shows the recovery process. We can see that the hollow place is starting to be filled with new cells, until it is almost completely recovered, as shown in Figure 7.1f.

This result demonstrates the power of the computational embryogeny scheme in solving problems that are mimicked from nature, such as self recovery. In the direct encoding approach where the growth process is omitted, these kinds of problems are simply not accessible.

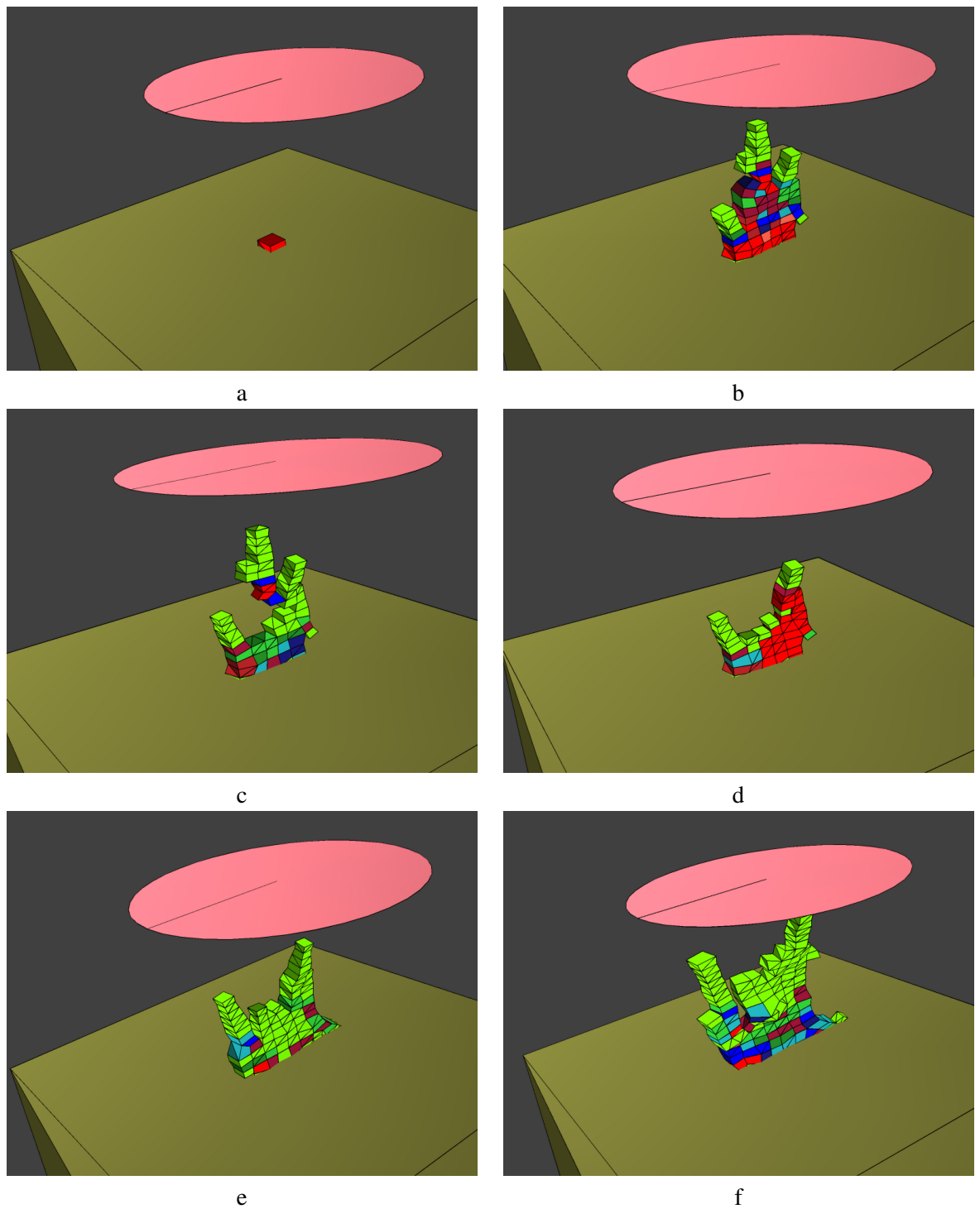


Figure 7.1: Six stages during the growth of a self recovered phenotype. a. Single cell b. Before destruction occur. c. Destruction of of the phenotype by removing a cluster of cells on the middle. d. Further destruction. e. First recovery f. Final stage, complete recovery.



## Chapter 8

# Appendix

The following sections provide a detailed explanation regarding the numerical methods and schemes used here.

### 8.1 Configuration

Three types of configurations are defined; local, reference, and deformed. These configurations are useful for the computing process of certain parameters such as mechanical stresses and morphogen gradient. The deformed configuration corresponds to the phenotype after it has been subjected to the load, and is thus deformed from its original state. This deformation generates strains which generate mechanical stresses. The reference configuration corresponds to the configuration of the phenotype before it has been subjected to the load. The local configuration is a topological configuration where every cell in the reference configuration is mapped to cube, centered at the origin, shown in Figure 8.1.

### 8.2 Finite Element Scheme

Every cell in the structure is a 3D hexahedral finite element. Every cell in the reference configuration has eight nodes. Every node has 3 coordinates as shown in Figure 8.1. If the cell is topologically convex, then there exists a one-to-one isoparametric mapping which maps every point inside the cell in the reference configuration into a point inside a cube centered on the origin with unit volume. A point inside the cell in its reference configuration is marked with  $X, Y, Z$ . The coordinates of the cell nodes in the reference configuration are defined as  $X_i, Y_i, Z_i \quad i = 0, \dots, 7$ . A point inside the cell in its local configuration is marked with  $r, s, t$ . The mapping between a point

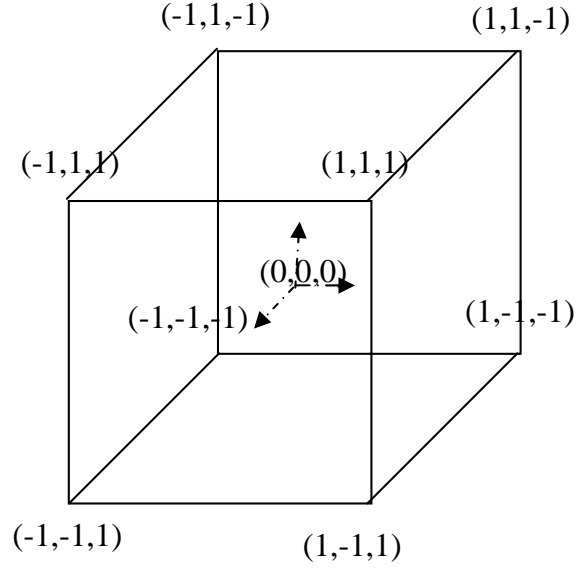


Figure 8.1: The local-global mapping.

in the local configuration of the cell  $(r, s, t)$  to a point in the reference configuration  $(X, Y, Z)$  is defined by equation 8.2. The shape function,  $N_i$ , is defined in equation 8.2.2.

$$\begin{aligned}
 X &= \sum_{i=0}^7 N_i(r, s, t) X_i \\
 Y &= \sum_{i=0}^7 N_i(r, s, t) Y_i \\
 Z &= \sum_{i=0}^7 N_i(r, s, t) Z_i
 \end{aligned} \tag{8.2.1}$$

where

$$\begin{aligned}
 N_0 &= \frac{1}{8}(1+r)(1-s)(1+t) \\
 N_1 &= \frac{1}{8}(1+r)(1+s)(1+t) \\
 N_2 &= \frac{1}{8}(1-r)(1+s)(1+t) \\
 N_3 &= \frac{1}{8}(1-r)(1-s)(1+t) \\
 N_4 &= \frac{1}{8}(1+r)(1-s)(1-t) \\
 N_5 &= \frac{1}{8}(1+r)(1+s)(1-t) \\
 N_6 &= \frac{1}{8}(1-r)(1+s)(1-t) \\
 N_7 &= \frac{1}{8}(1-r)(1-s)(1-t)
 \end{aligned} \tag{8.2.2}$$

A phenotype is simply a collection of cells in the reference configuration. The finite element approximation relates the displacement field  $\mathbf{u}$  at every point inside the cell to the displacement field

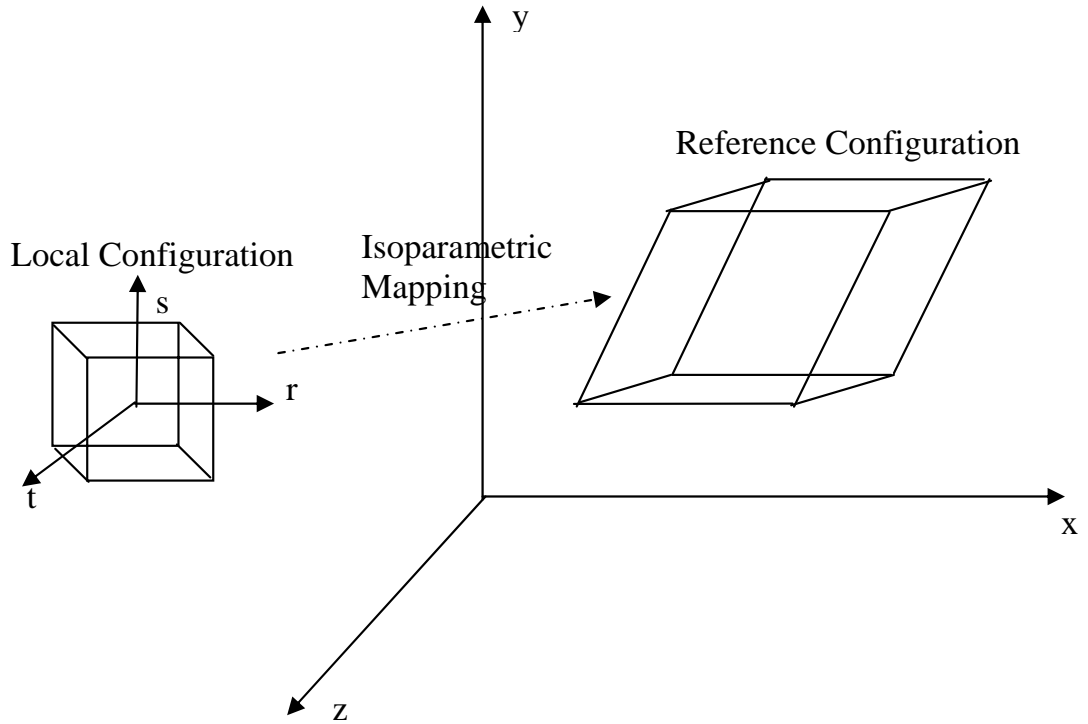


Figure 8.2: The basic structural hexahedral “brick” 3-D finite element.

of its nodes  $\mathbf{u}_i$  through the shape function in equation 8.2.2 according to equation 8.2.3.

$$\mathbf{u} \approx \sum_{i=0}^7 N_i(r, s, t) \mathbf{u}_i \quad (8.2.3)$$

From linear elasticity theory the strain tensor  $\varepsilon$  at every point is defined by equation 8.2.4 where  $\mathbf{L}$  is a linear operator defined in equation 8.2.5.

$$\varepsilon = \mathbf{L}\mathbf{u} \quad (8.2.4)$$

$$\mathbf{L} = \begin{bmatrix} \partial/\partial X & 0 & 0 \\ 0 & \partial/\partial Y & 0 \\ 0 & 0 & \partial/\partial Z \\ \partial/\partial Y & \partial/\partial Z & 0 \\ 0 & \partial/\partial Z & \partial/\partial Y \\ \partial/\partial Z & 0 & \partial/\partial X \end{bmatrix} \quad (8.2.5)$$

Using both equation 8.2.4 and the linear operator  $\mathbf{L}$  the *strain-nodal displacement matrix*  $\mathbf{B}$  is

defined as shown in equation 8.2.6.

$$\mathbf{B} = \mathbf{LN} = \begin{bmatrix} \sum_{i=0}^7 \frac{\partial N_i(r,s,t)}{\partial x} & 0 & 0 \\ 0 & \sum_{i=0}^7 \frac{\partial N_i(r,s,t)}{\partial y} & 0 \\ 0 & 0 & \sum_{i=0}^7 \frac{\partial N_i(r,s,t)}{\partial z} \\ \sum_{i=0}^7 \frac{\partial N_i(r,s,t)}{\partial y} & \sum_{i=0}^7 \frac{\partial N_i(r,s,t)}{\partial z} & 0 \\ 0 & \sum_{i=0}^7 \frac{\partial N_i(r,s,t)}{\partial z} & \sum_{i=0}^7 \frac{\partial N_i(r,s,t)}{\partial y} \\ \sum_{i=0}^7 \frac{\partial N_i(r,s,t)}{\partial z} & 0 & \sum_{i=0}^7 \frac{\partial N_i(r,s,t)}{\partial x} \end{bmatrix} \quad (8.2.6)$$

The derivatives of the shape function 8.2.6, with respect to the local coordinates, are mapped to their derivatives with respect to the reference configuration using the Jacobian, shown in equation 8.2.7.

$$\mathbf{J} = \begin{bmatrix} \sum_{j=0}^7 \frac{\partial N_j}{\partial r} x_j & \sum_{j=0}^7 \frac{\partial N_j}{\partial r} y_j & \sum_{j=0}^7 \frac{\partial N_j}{\partial r} z_j \\ \sum_{j=0}^7 \frac{\partial N_j}{\partial s} x_j & \sum_{j=0}^7 \frac{\partial N_j}{\partial s} y_j & \sum_{j=0}^7 \frac{\partial N_j}{\partial s} z_j \\ \sum_{j=0}^7 \frac{\partial N_j}{\partial t} x_j & \sum_{j=0}^7 \frac{\partial N_j}{\partial t} y_j & \sum_{j=0}^7 \frac{\partial N_j}{\partial t} z_j \end{bmatrix} \quad (8.2.7)$$

Using the definition in equation 8.2.7, the derivatives of the shape function can be expressed with respect to the reference configuration as shown in equation 8.2.8.

$$\begin{bmatrix} \frac{\partial N_i}{\partial X} \\ \frac{\partial N_i}{\partial Y} \\ \frac{\partial N_i}{\partial Z} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \frac{\partial N_i}{\partial r} \\ \frac{\partial N_i}{\partial s} \\ \frac{\partial N_i}{\partial t} \end{bmatrix} \quad (8.2.8)$$

In order to derive the relation between the external forces and the strains, the functional  $\Pi$  is defined, as shown in equation 8.2.9. The functional is defined as the difference between the internal strain energy and work on the boundary.

$$\Pi = \int_{V^e} \sigma \varepsilon dV - \int_{V^e} \mathbf{ub} dV - \int_{S^e} \mathbf{us} dS - \sum_{i=0}^7 \mathbf{uf}_i \quad (8.2.9)$$

where  $\mathbf{b}$  represents the body force (*e.g.*, gravity),  $\sigma$  is the mechanical stress tensor,  $\mathbf{s}$  are the surface tractions, and  $\mathbf{f}_i$  are the forces applied to the nodes.

The principle of virtual work applied to equation 8.2.9 results in equation 8.2.10.

$$\begin{aligned} \delta\Pi = \int_{V^e} \sigma \delta(\varepsilon) dV - \int_{V^e} \delta(\mathbf{u}) \mathbf{b} dV - \\ \int_{S^e} \delta(\mathbf{u}) \mathbf{b} dS - \sum \delta(\mathbf{u}) \mathbf{f}_p = 0 \end{aligned} \quad (8.2.10)$$

where

$$\begin{aligned} \int_{V^e} \sigma \delta(\varepsilon) dV = \int_{V^e} \delta(\mathbf{u}) \mathbf{b} dV + \\ \int_{S^e} \delta(\mathbf{u}) \mathbf{s} dS + \sum \delta(\mathbf{u}) \mathbf{f}_p \end{aligned} \quad (8.2.11)$$

Equation 8.2.10 represents the relationship between the external forces, body forces and surface traction to the stress and the strains at every point in the cell. Two assumptions are made; the strain-displacement relation is linear, as shown in equation 8.2.4, and the constitutive equation which relates stress to strain is linear as well. The constitutive relation is presented in equation 8.2.12, where  $\mathbf{D}$  is the material property matrix defined in equation 8.2.13. The material properties  $E$  and  $\nu$  are the elastic modulus and Poisson's ratio respectively.

$$\sigma = \mathbf{D}\varepsilon \quad (8.2.12)$$

$$\mathbf{D} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & & & \\ -\nu & 1 & -\nu & & & \\ -\nu & -\nu & 1 & & & \\ & & & 1 + \nu & & \\ & & & & 1 + \nu & \\ & & & & & 1 + \nu \end{bmatrix} \quad (8.2.13)$$

Using equations 8.2.4, 8.2.12, and 8.2.6 in equation 8.2.10 a set of of eight linear equations are obtained, represented in matrix form as shown in equation 8.2.14.

$$\mathbf{K}\mathbf{u}_i^e = \mathbf{f}_i^e \quad (8.2.14)$$

where  $\mathbf{K}$  is the stiffness matrix, defined in equation 8.2.15. Equation 8.2.15 is defined as the integration of  $\mathbf{B}^T \mathbf{D} \mathbf{B}$  over the volume of the cell. The stiffness matrix should be positive definite and symmetric.

$$\mathbf{K} = \int_{V^e} \mathbf{B}^T \mathbf{D} \mathbf{B} dV \quad (8.2.15)$$

### 8.2.1 Integration scheme

The stiffness matrix in equation 8.2.15 requires integration across the volume of the cell. In order to perform this integration efficiently, the Gauss integration method is used. The expression  $\mathbf{B}^T \mathbf{D} \mathbf{B} dV$  in equation 8.2.15 is a function of the local coordinates  $r, s, t$  only. The first step is to write the integrand in equation 8.2.15 over a cube in the local coordinate system. Using the definition of the Jacobian in equation 8.2.7 the stiffness matrix can be written, as shown in equation 8.2.16.

$$K^e = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \mathbf{B}^T \mathbf{D} \mathbf{B} |\det \mathbf{J}| dr ds dt \quad (8.2.16)$$

The integral in equation 8.2.16 can be computed numerically using the Gauss integration method, as shown in equation 8.2.17. The quadrature points  $r_i, s_i, t_i$  are defined in equation 8.2.18.

$$K^e = \sum_{r_i, s_i, t_i}^8 |\det \mathbf{J}| \left( \mathbf{B}^T \mathbf{D} \mathbf{B} \right) \Big|_{r_i, s_i, t_i} \quad (8.2.17)$$

$$r_i, s_i, t_i = \{\pm 0.57735, \pm 0.57735, \pm 0.57735\} \quad (8.2.18)$$

### 8.2.2 Assembly scheme

The system of equations defined in equation 8.2.14 corresponds to a single cell. In order to evaluate the displacements of the entire structure, all of the local stiffness matrices must be collected into a single global stiffness matrix  $\bar{\mathbf{K}}$ . The corresponding external forces, body forces and surface tractions from all nodes must also be collected into a global vector  $\bar{\mathbf{f}}$ . At the end of the assembly routine a system of linear equations is produced, shown in equation 8.2.19.

$$\bar{\mathbf{K}} \bar{\mathbf{u}} = \bar{\mathbf{f}} \quad (8.2.19)$$

The first step is to assign sequential numbers to all the nodes. Every node gets three numbers, since the displacement field and is a 3D vector. A typical system of local equations in the form of equation 8.2.12 is presented in equation 8.2.20.

$$\begin{bmatrix} K_{00} & K_{07} \\ \cdot & \cdot \\ K_{70} & K_{77} \end{bmatrix} \begin{bmatrix} u_{p_0} \\ \cdot \\ u_{p_7} \end{bmatrix} = \begin{bmatrix} f_{p_0} \\ \cdot \\ f_{p_7} \end{bmatrix} \quad (8.2.20)$$

The global indices of the nodes are defined as  $p_0$  to  $p_7$ . Starting with an empty global stiffness matrix, every row in the local stiffness matrix will be inserted into the global stiffness matrix in the following way; the index of the node which corresponds to this row will be the number of the row in the global stiffness matrix. Every term is then sequentially placed in column indices according to  $p_0$  to  $p_7$ .

The assembly process for the external force will simply assign the force on a node to its location in the global matrix. This scheme will eventually generate a global set of equations for which the displacement field at each node can be computed. Once the displacement field is computed, the mechanical stress at each point inside the cell is determined using equations 8.2.4 and 8.2.12.

## Chapter 9

# Cell Adhesion Scheme

The following sections provide a detailed explanation regarding the scheme used here for adhesion between neighboring cells. This method may also be utilized in standard finite element mesh generation, where the mesh starts at several locations, from the exterior boundary and propagates to the center.

### 9.1 Cell Adhesion Scheme in Finite Element Mesh

The adhesion problem between two finite elements is common, especially in 3D meshes of nonconvex shapes. The mesh generation of nonconvex shapes usually starts at multiple locations along the boundary and proceeds towards the center. At some point during this process each of the individual meshes must coincide and adhere. This problem, from a geometric viewpoint, is complex because the adjacent element orientations are generally not able to be predicted, and it is difficult to determine the best configuration for adhesion *a priori*. As a result, this kind of problem is generally solved with heuristic rules. These rules handle different situations in different ways and are primarily determined by the skill and experience of the programmer. The disadvantages of using heuristics include lack of generality, and difficulties with unforeseen situations. A new approach to address the problem of adhesion is shown here. This solution does not apply any heuristic rules, but rather uses a natural approach based on minimization of strain energy that can automatically adhere two cells in an optimum configuration. This approach, with minor adjustments, is suitable for most types of 3D elements.



## 9.2 Overview of Existing Mesh Techniques

Despite considerable work, the problem of generalizing meshes into arbitrary non-convex three dimensional bodies remains an open area of research. Excluding mesh-free methods, 3-D meshes are generally composed of either tetrahedral or hexahedral elements. In the tetrahedral case, three popular methods have been developed for automated mesh generation for three dimensional bodies. The three methods are: *octree*, *Delaunay triangulation* and *advancing front*.

The *octree* method was developed by 1980 by Mark Shephard [86] and [91]@. With this method, cubes containing the geometric model are recursively subdivided until the desired resolution is reached. The *octree* method has a significant disadvantage in that a predefined surface mesh is not matched, but rather surface facets are formed wherever the internal *octree* structure intersects the boundary.

In contrast, the *Delaunay triangulation* method matches the body surface more accurately [75, 83]. This method is based on the *Delaunay criterion*, which tests a triangulated set of vertices such that any circumsphere that predefines the mesh does not contain any of these vertices. This criterion is used in generating tetrahedral meshes for 3D shapes [79, 90]. Although the *Delaunay triangulation* method can generate a mesh for any given convex body defined as a convex hull, it cannot generate meshes for non-convex domains [87].

The *advancing front* method deals with the issues of non-convexity successfully, and today is a popular method for generating meshes for both convex and non-convex bodies. In this method the tetrahedra are generated progressively inwardly from the triangulated surface [84, 85]. This is analogous to the front of a wave transmitted in the body. In the case of tetrahedral elements, the algorithm first generates a triangle base on each facet of the front, and then optimizes the location of a fourth node. The fourth node might be new or an existing node of an adjacent element. The algorithm keeps running a collision detection sub-algorithm to ensure that no elements are overlapping. The *advancing front* method has also been used with meshes of hexahedral elements [96]. The algorithm produces two tetrahedral elements that combine into a single quadrilateral.

The common solution for addressing adhesion of finite elements is to create a set of heuristic rules. These rules test and categorize states elements according to their relative position and orientation. Each state corresponds to an adhesion process that merges elements in a predefined manner.

The use of heuristic rules creates many problems. First, it requires a long set of conditional

statements that introduce complexity into the code. Additionally, the rules must address and capture all situations regarding the relative position and orientation of the elements, and decide which kind of adhesion process to execute. Since the number of possible orientations is high, it is difficult to ensure that all possible situations are addressed. This is the motivation for developing a new method that automates the adhesion process between two finite elements regardless of their orientations

### 9.2.1 Our Approach

Our approach focuses on the *nonoverlapping* criterion as part of the *advancing front* method, and is motivated by two coincident finite elements that need to reform and attach in an optimum way. The case of two overlapping polyhedra may also appear in other applications such as computer graphics and CAD software.

Our approach mimics an observed adhesion phenomena in nature, such as the adhesion of biological cells and bubbles. By using the basic principles used in nature, a fully automatic adhesion routine has been created that can address any situation of two nearby or overlapping elements that are to be adhered.

### 9.2.2 Overview

In Section 9.3 the idea of formulating each finite element as an artificial cell containing some particular properties is introduced. The principle of minimum potential energy, as observed in bubble adhesion processes, is also established and explained. Section 9.4 describes the governing equations for applying this method to an eight node hexahedral linear brick finite element. Section 9.5 presents the derivation of the conjugate gradient method used here to minimize strain energy in the configuration of the adhered elements. An analytical expression for the strain energy gradient has been derived, which significantly increases the speed of convergence. Section 9.6 shows the results of the adhesion of two elements using this method. Several examples illustrate the capabilities of this approach in automating the adhesion process without using heuristic rules. Conclusions are presented in Section 9.7.

## 9.3 The Adhesion Approach

Consider two finite elements (Figure 9.1a) that under certain conditions need to go through an adhesion process. The condition for adhesion may be based upon the distance between the elements

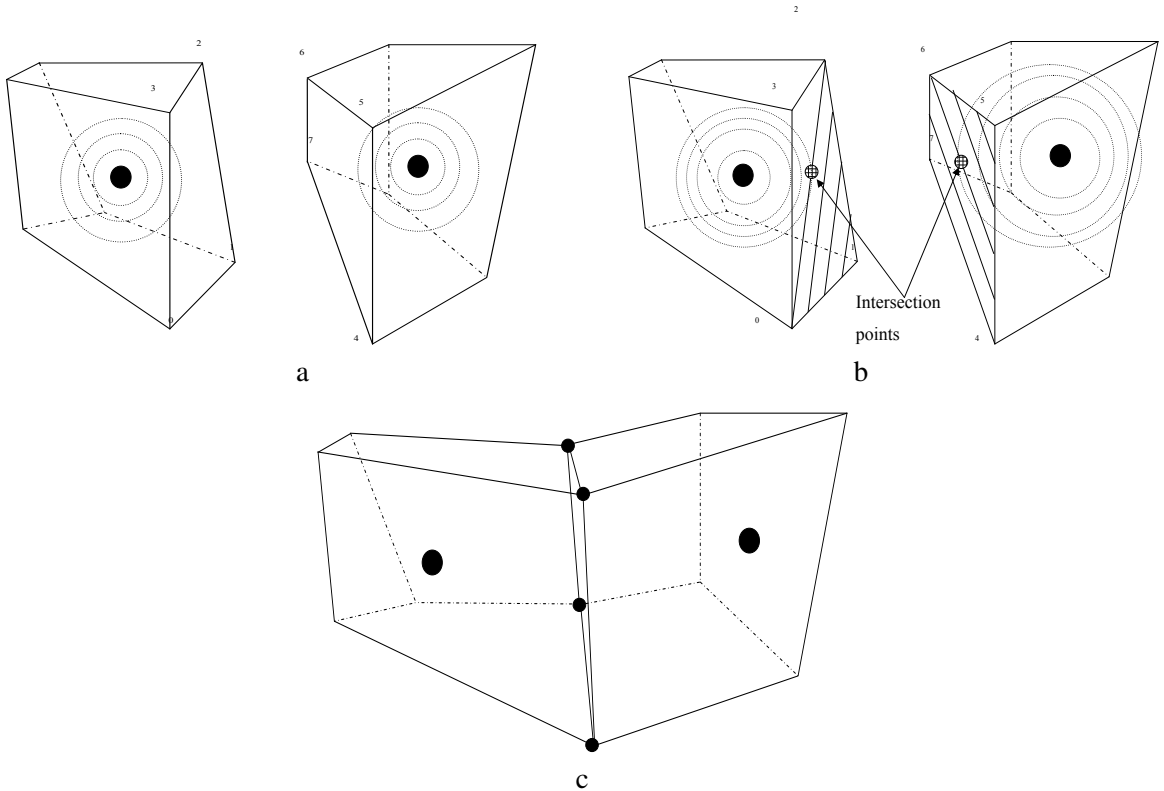


Figure 9.1: a) Two cells prior to adhesion with radiating signals from the center of each cell. b) The faces of the two cells to be adhered. c) The cells after adhesion.

or an intersection between them [76]. The adhesion process takes one face from each element and merges their corresponding nodes.

The method introduced here comprises three steps: first determine the corresponding faces to be adhered; second determine the pairs of nodes to be merged, and third optimize the positions of the merged nodes.

Cells in nature, prior to adhesion, communicate by way of a chemical signaling mechanism [74]. This process is mimicked here by considering each finite element to be an analog of a cell. Each cell generates a signal at its center and this signal propagates through the faces of the cell and into the adjacent cells, schematically illustrated in Figure 9.1a. Each signal is characterized by an exponential function 9.3.1 that decays with distance from the center of the cell:

$$\vec{S} = \exp(-\alpha \vec{x}_c) \quad (9.3.1)$$

where  $\alpha$  is a constant,  $\vec{x}_c$  is the vector of the coordinates of the center of the cell, and  $\vec{S}$  is a gradient vector.

The face of each cell which contains the maximum intensity of the adjacent signal from the adjacent cell is chosen for adhesion, as illustrated schematically in Figure 9.1b. The inverse isoparametric mapping [94] is applied, which maps each cell to its local configuration. By inverse mapping the signal as well, the detection of the point with the maximum signal gradient on the adjacent face becomes straightforward.

Once the faces to be adhered have been identified, the best four pairs of nodes to be merged must be selected. Each pair of nodes will be merged into a single node initially positioned at the *mean* of the coordinates of both original nodes. At the end of the process, the two cells adhere to each other without any overlap, as illustrated in Figure 9.1c.

For two hexahedral elements, there are a total of 24 different configurations of merged node pairs from which to choose. The selection of the optimum configuration mimics the process of bubble adhesion [82]. When bubbles adhere, they adopt a shape that reduces their total adhesion and surface energy. This principle maintains the bubbles spherical shape, resulting in minimum distortion.

Using this general principle for all 24 possible combinations of nodes to merge, the pairing of nodes that results in the minimum distortion to both cells, and thus introduces the minimum change in strain energy (when each pair of nodes is merged into a single node positioned at the *mean* of the original coordinates of the nodes) is selected. Any contribution to the change in strain energy will originate only from the distortion of the cells. This idea is applied rigorously here by considering the constitutive properties of the material of each cell to be neo-Hookean for the purpose of adhering the cells.

Prior to the adhesion process, both cells are in a *reference configuration*. While the adhesion process is being applied, a finite displacement constrains each pair of nodes. The strain energy density relationship for neo-Hookean materials is given by  $w$  in 9.3.2:

$$w = C_1 (I_1 - 3) \quad (9.3.2)$$

where  $C_1$  is a constant, and:

$$\begin{aligned} I_1 &= \vec{C} \cdot \vec{I} \\ &= \left( \vec{F}'^T \vec{F}' \right) \cdot \vec{I} \end{aligned} \quad (9.3.3)$$

$$\vec{F}' = \det(\vec{F})^{-1/3} \vec{F} \quad (9.3.4)$$

$$\begin{aligned} \vec{F} &= \frac{\partial \vec{x}}{\partial \vec{X}} = \frac{\partial(\vec{X} + \vec{u})}{\partial \vec{X}} d \\ &= \vec{I} + \frac{\partial \vec{u}}{\partial \vec{X}} \end{aligned} \quad (9.3.5)$$

where  $\vec{u}$  are the displacements,  $\vec{x}$  are the new coordinates, and  $\vec{X}$  are the initial coordinates of the cell;  $\vec{C}$  is the Cauchy stress tensor,  $\vec{F}'$  is the distortional part of the deformation gradient, and  $\vec{F}$  is defined in 9.3.5 such that  $\det(\vec{F}') = 1$  [78]. This is the pure measure of distortion.

The deformation gradient  $\vec{F}$  is composed of two parts; the identity tensor  $\vec{I}$  and the derivatives of the displacements with respect to the reference configuration. The displacements of the nodes are known, and the derivatives can be computed using the isoparametric mapping [89, 96]. Once the strain energy density  $w$  is determined, the total strain energy can be computed by integrating  $w$  over the volume of the cell as shown in 9.3.6:

$$U = \int_{\Omega} w dV \quad (9.3.6)$$

Out of the 24 possible combinations, the one that produces the minimum strain energy  $U$  is chosen. At the end of this process, the two cells are merged with no overlap, and the strain energy required for the operation is the minimum of all 24 cases.

This state can be improved, in terms of energy minimization. The initial assumption of merging a pair of nodes by taking the average of their corresponding coordinates will not generally result in a global energy minimization for the two adhered cells. In order to improve the initial assumption, the conjugate gradient method is applied to the displacements to find the global minimum of the strain energy, as shown in 9.3.7, where  $\vec{u}_k^1$  are the displacements of the nodes corresponding to the selected face of the first cell, and  $\vec{u}_k^2$  is defined in the same way for the selected face of the second cell. The iteration of  $\vec{u}_{k_j}^m$ ,  $j \geq 0$  in the conjugate gradient method satisfies the recursion shown in 9.3.8 [80, 81].

$$\min(U(\vec{u}_k^1 + \vec{u}_k^2) : \vec{u} \in R^3) \quad k = 1, \dots, 4 \quad (9.3.7)$$

$$\vec{u}_{k_{j+1}}^m = \vec{u}_{k_j}^m + \alpha_j \vec{d}_j \quad (9.3.8)$$

The directions  $\vec{d}_j$  are generated by the rule in 9.3.9:

$$\vec{d}_{j+1} = -\vec{g}_{j+1} + \beta_j \vec{d}_j \quad (9.3.9)$$

where:

$$\begin{aligned} \vec{d}_0 &= -\vec{g}_0 \\ \vec{g}_j &= \nabla U(\vec{u}_{k_j}^m) \end{aligned} \quad (9.3.10)$$

The step size  $\alpha_j$  is determined by a global line search developed by Fletcher-Reeves [77], shown in 9.3.11:

$$\alpha_j = -\frac{\vec{g}_j^T \cdot \vec{d}_j}{\vec{d}_j^T \cdot \vec{g}_j} \quad (9.3.11)$$

An additional constraint was developed by Wolf [95], shown in 9.3.12, with the inequality  $0 < \delta < \sigma < 1$ , and where  $\beta$  is determined according to Fletcher-Reeves algorithm 9.3.13:

$$\begin{aligned} U(\vec{u}_{k_j}^m) - U(\vec{u}_{k_j}^m + \alpha_j \vec{d}_j) &\geq -\delta \alpha_j (\vec{g}_j^T \cdot \vec{d}_j) \\ |U'(\vec{u}_{k_j}^m)| &\leq -\sigma (\vec{g}_j^T \cdot \vec{d}_j) \end{aligned} \quad (9.3.12)$$

$$\beta = \frac{(\vec{g}_{j+1}^T \cdot \vec{g}_{j+1}^T)}{(\vec{g}_j^T \cdot \vec{g}_j^T)} \quad (9.3.13)$$

The primary difficulty in the conjugate gradient method is determining  $\vec{g}_j$  in a closed analytical form. In this paper a closed, compact, expression for  $\vec{g}_j$  is presented that can easily be modified for different types of finite elements. The next sections describe the steps for applying this method using a hexahedral linear “brick” finite element.

## 9.4 Hexahedral Linear Brick Element

To begin, the two faces to be merged, and their corresponding pairs of nodes, need to be identified, utilizing the procedure outlined above.

Assuming that every cell is convex, a one-to-one isoparametric mapping can be defined 9.4.14, where  $(r, s, t)$  represents a point in the local configuration of the cell, and  $(X, Y, Z)$  represents the point in the reference configuration of the cell.

$$\begin{aligned}
X &= \sum_{i=1}^8 N_i(r, s, t) X_i \\
Y &= \sum_{i=1}^8 N_i(r, s, t) Y_i \\
Z &= \sum_{i=1}^8 N_i(r, s, t) Z_i
\end{aligned} \tag{9.4.14}$$

The inverse isoparametric mapping of points in the reference configuration to the local configuration is determined using the Newton-Raphson method. For a given  $\vec{X} = \{X, Y, Z\}$ , the corresponding  $\vec{r} = \{r, s, t\}$  is found, using 9.4.15:

$$\begin{aligned}
f(\vec{X} - \vec{r}) &= 0 \\
f(\vec{X} - \vec{r}) &\approx f(\vec{X} - \vec{r}_0) + \frac{\partial f(\vec{X} - \vec{r})}{\partial \vec{r}} \delta \vec{r} = 0
\end{aligned} \tag{9.4.15}$$

where,  $\delta \vec{r}$  is a small increment from the initial guess  $\vec{r}_0$ , which is determined by 9.4.16:

$$\delta \vec{r} = - \frac{\partial f(\vec{X} - \vec{r})}{\partial \vec{r}}^{-1} f(\vec{X} - \vec{r}_0) = \vec{M} f(\vec{X} - \vec{r}_0) \tag{9.4.16}$$

Since a solution exists, and is unique, this method will always converge. This iterative process terminates when  $\|\delta \vec{r}\| < \varepsilon$ . The stiffness matrix  $\vec{M}$  is defined in 9.4.17:

$$\vec{M} = \begin{bmatrix} \frac{\partial \left( \sum_{i=0}^7 N_i(r, s, t) X_i \right)}{\partial r} & \frac{\partial \left( \sum_{i=0}^7 N_i(r, s, t) X_i \right)}{\partial s} & \frac{\partial \left( \sum_{i=0}^7 N_i(r, s, t) X_i \right)}{\partial t} \\ \frac{\partial \left( \sum_{i=0}^7 N_i(r, s, t) Y_i \right)}{\partial r} & \frac{\partial \left( \sum_{i=0}^7 N_i(r, s, t) Y_i \right)}{\partial s} & \frac{\partial \left( \sum_{i=0}^7 N_i(r, s, t) Y_i \right)}{\partial t} \\ \frac{\partial \left( \sum_{i=0}^7 N_i(r, s, t) Z_i \right)}{\partial r} & \frac{\partial \left( \sum_{i=0}^7 N_i(r, s, t) Z_i \right)}{\partial s} & \frac{\partial \left( \sum_{i=0}^7 N_i(r, s, t) Z_i \right)}{\partial t} \end{bmatrix} \tag{9.4.17}$$

The maximum signal gradient on the face of each cell, generated by the adjacent cell, is simply the intersection point of a line which connects the center of both cells, illustrated in Figure 9.1b.

Define  $\vec{A} = \{a_1, a_2, a_3\}$  to be a vector representing the line which connects the center of both cells.  $\vec{O}_i$  represents the coordinates of the  $i^{th}$  cell center point.  $\vec{b}_i = \vec{O}_i + \varepsilon \vec{A}$  represents a point that is near the center of the cell along the vector  $\vec{a}$ . Since  $\vec{B}_i$  is an internal point, it can be mapped to the local configuration of the cell using 9.4.15, 9.4.16 and 9.4.17. Defining  $\vec{b}_i$  to be the inverse mapping of  $\vec{B}_i$ , the point  $(0, 0, 0)$  is the inverse mapping of  $\vec{O}_i$ . The inverse mapping of the vector

$\vec{A}$  is simply  $\vec{a}_{\vec{z}} = \vec{B}_i - (0, 0, 0) = \vec{B}_i$ . The face which intersects  $\vec{a}_{\vec{z}}$  corresponds to the sign of the maximum term in  $\vec{a}_{\vec{z}}$ , and also corresponds to the face which intersects  $\vec{A}$  in the reference configuration, *i.e.*, if  $\vec{a}_{\vec{z}} = (0.14, 0.98, 0.14)$  then the face above the origin in the  $y$ -direction intersects the vector  $\vec{a}_{\vec{z}}$ . Utilizing this method, the two faces to be adhered are determined.

Next, the most suitable configuration for adhesion of the 24 possible combinations of pairs of nodes must be found. For each case, the displacement of every pair of nodes is computed, as shown in 9.4.18:

$$\vec{u}_k^m = \frac{\vec{X}_k^1 + \vec{X}_k^2}{2} - \vec{X}_k^m \quad m = 1, 2 \quad k = 1, \dots, 4 \quad (9.4.18)$$

where  $\vec{u}_k^m$  are the displacements at node  $k$ , and  $\vec{X}_k^1$  are the initial coordinates of the node.

The steps for computing the strain energy  $U$ , for every possible combination of node pairs, are explained next.

The deformation gradient tensor  $\vec{F}$ , in terms of displacements  $\vec{u}$  [88], is given by 9.4.19:

$$\vec{F} = \frac{\partial \vec{x}}{\partial \vec{X}} = \frac{\partial (\vec{X} + \vec{u})}{\partial \vec{X}} = \vec{I} + \frac{\partial \vec{u}}{\partial \vec{X}} = \begin{bmatrix} 1 + \frac{\partial u}{\partial X} & \frac{\partial u}{\partial Y} & \frac{\partial u}{\partial Z} \\ \frac{\partial v}{\partial X} & 1 + \frac{\partial v}{\partial Y} & \frac{\partial v}{\partial Z} \\ \frac{\partial w}{\partial X} & \frac{\partial w}{\partial Y} & 1 + \frac{\partial w}{\partial Z} \end{bmatrix} \quad (9.4.19)$$

where all the of the derivatives in 9.4.19 are with respect to the reference configuration [96], which is the configuration before the adhesion process.

Once the displacements at the nodes have been determined, the displacements inside the elements can be approximated using the finite element relation given in 9.4.20:

$$\begin{aligned} u &\approx N_i(r, s, t) u_i \\ v &\approx N_i(r, s, t) v_i \quad i = 1, \dots, 8 \\ w &\approx N_i(r, s, t) w_i \end{aligned} \quad (9.4.20)$$

where  $N_i(r, s, t)$  are the shape functions with respect to the local coordinates [89] given by the



relations shown in 9.4.21:

$$\begin{aligned}
N_1 &= \frac{1}{8} (1+r)(1-s)(1+t) & N_2 &= \frac{1}{8} (1+r)(1+s)(1+t) \\
N_3 &= \frac{1}{8} (1-r)(1+s)(1+t) & N_4 &= \frac{1}{8} (1-r)(1-s)(1+t) \\
N_5 &= \frac{1}{8} (1+r)(1-s)(1-t) & N_6 &= \frac{1}{8} (1+r)(1+s)(1-t) \\
N_7 &= \frac{1}{8} (1-r)(1+s)(1-t) & N_8 &= \frac{1}{8} (1-r)(1-s)(1-t)
\end{aligned} \tag{9.4.21}$$

By combining 9.4.19, 9.4.20, and 9.4.21, the displacement derivatives with respect to the reference coordinate system are obtained 9.4.22.

$$\begin{aligned}
\frac{\partial u}{\partial X} &= \frac{\partial N_i}{\partial X} u_i & \frac{\partial u}{\partial Y} &= \frac{\partial N_i}{\partial Y} u_i & \frac{\partial u}{\partial Z} &= \frac{\partial N_i}{\partial Z} u_i \\
\frac{\partial v}{\partial X} &= \frac{\partial N_i}{\partial X} v_i & \frac{\partial v}{\partial Y} &= \frac{\partial N_i}{\partial Y} v_i & \frac{\partial v}{\partial Z} &= \frac{\partial N_i}{\partial Z} v_i \\
\frac{\partial w}{\partial X} &= \frac{\partial N_i}{\partial X} w_i & \frac{\partial w}{\partial Y} &= \frac{\partial N_i}{\partial Y} w_i & \frac{\partial w}{\partial Z} &= \frac{\partial N_i}{\partial Z} w_i
\end{aligned} \tag{9.4.22}$$

The mapping from the local coordinate system  $(r, s, t)$  to the reference system is defined by the Jacobian 9.4.23, where  $(X_j, Y_j, Z_j)$  are the coordinates of the node in the reference configuration.

$$\vec{J} = \begin{bmatrix} \sum_{j=1}^8 \frac{\partial N_j}{\partial r} X_j & \sum_{j=1}^8 \frac{\partial N_j}{\partial r} Y_j & \sum_{j=1}^8 \frac{\partial N_j}{\partial r} Z_j \\ \sum_{j=1}^8 \frac{\partial N_j}{\partial s} X_j & \sum_{j=1}^8 \frac{\partial N_j}{\partial s} Y_j & \sum_{j=1}^8 \frac{\partial N_j}{\partial s} Z_j \\ \sum_{j=1}^8 \frac{\partial N_j}{\partial t} X_j & \sum_{j=1}^8 \frac{\partial N_j}{\partial t} Y_j & \sum_{j=1}^8 \frac{\partial N_j}{\partial t} Z_j \end{bmatrix} \tag{9.4.23}$$

Using 9.4.23, the derivatives of the shape function with respect to the reference coordinate system are defined in 9.4.24.

$$\begin{bmatrix} \frac{\partial N_i}{\partial X} \\ \frac{\partial N_i}{\partial Y} \\ \frac{\partial N_i}{\partial Z} \end{bmatrix} = \vec{J}^{-1} \begin{bmatrix} \frac{\partial N_i}{\partial r} \\ \frac{\partial N_i}{\partial s} \\ \frac{\partial N_i}{\partial t} \end{bmatrix} \tag{9.4.24}$$

Substituting 9.4.23 and 9.4.24 into 9.4.19,  $\vec{F}$  can be determined at each point in the cell using 9.4.25:

$$\vec{F} = \vec{I} + \begin{bmatrix} \frac{N_i(r,s,t)u_i}{\partial X} & \frac{N_i(r,s,t)u_i}{\partial Y} & \frac{N_i(r,s,t)u_i}{\partial Z} \\ \frac{N_i(r,s,t)v_i}{\partial X} & \frac{N_i(r,s,t)v_i}{\partial Y} & \frac{N_i(r,s,t)v_i}{\partial X} \\ \frac{N_i(r,s,t)w_i}{\partial X} & \frac{N_i(r,s,t)w_i}{\partial Y} & \frac{N_i(r,s,t)w_i}{\partial Z} \end{bmatrix} \quad i = 1, \dots, 8 \quad (9.4.25)$$

The strain energy density can be found using 9.3.2 and 9.3.4. Finally the total strain energy 9.3.5 can be found using the Gauss-Legendre integration method given by 9.4.26:

$$\begin{aligned} U = \int_{\Omega} w dV &= C_1 \int_{\Omega} (I_1 - 3) \det(\vec{J}) dr ds dt \\ &\approx C_1 \sum_{i=1}^8 (I_1 - 3)_{r=r_i, s=s_i, t=t_i} w_i \det(\vec{J}_i) \end{aligned} \quad (9.4.26)$$

where  $r_i, s_i, t_i$  represent a set of 8 different evaluation points; each coordinate can each take on the value  $\pm 0.577350269189626$ . Each point  $r_i, s_i, t_i$  is multiplied by its corresponding weight  $w_i = 1$  respectively, and with  $C_1$  set arbitrarily.<sup>1</sup>

## 9.5 The Conjugate Gradient Method

The goal of this method is to minimize the strain energy of the merged cells, by adjusting the position of the merged nodes from their initial configuration. As described above, the initial configuration is the one that introduces the minimum change in strain energy when each pair of nodes is merged into a single node, where each merged node is initially positioned at the *mean* of the original coordinates of the nodes. The conjugate gradient method requires the computation of the gradient of the strain energy  $w$ , with respect to the displacements of the nodes  $\vec{u}_k$  (the nodes which correspond to the merged faces), and is given in 9.5.27:

$$\frac{\partial w}{\partial \vec{u}_k} = C_1 \frac{\partial (I_1)}{\partial \vec{u}_k} \quad (9.5.27)$$

Using the finite element relation 9.5.28, the first invariant with the modification of the deformation gradient tensor 9.3.4, can be determined using 9.5.29:

---

<sup>1</sup>A suggested value is the same order of magnitude as the Young's modulus of the material.

$$\begin{aligned}\vec{u} &= \sum_{i=1}^8 N_i \vec{u}_i \\ \vec{\dot{u}} &= \sum_{i=1}^8 N_i \vec{\dot{u}}_i\end{aligned}\tag{9.5.28}$$

$$\begin{aligned}I_1 = \vec{C} \cdot \vec{I} &= \left( \vec{F}^T \vec{F}' \right) \cdot \vec{I} \\ &= \left( J^{-1/3} \vec{F}^T J^{-1/3} \vec{F} \right) \cdot \vec{I} \\ &= J^{-2/3} \left( \vec{F}^T \vec{F} \right) \cdot \vec{I} \\ &= J^{-2/3} \vec{C} \cdot \vec{I}\end{aligned}\tag{9.5.29}$$

where  $J = \det(\vec{F})$ .

The derivatives of  $I_1$  with respect to  $\vec{u}_k$  are then computed, as shown in 9.5.30:

$$\frac{\partial I_1}{\partial \vec{u}_i} = -\frac{2}{3} J^{-5/3} \frac{\partial J}{\partial \vec{u}_i} \left( \vec{C} \cdot \vec{I} \right) + J^{-2/3} \frac{\partial \left( \vec{C} \cdot \vec{I} \right)}{\partial \vec{u}_i}\tag{9.5.30}$$

The two terms in 9.5.30 are handled differently. The Cauchy stress tensor  $\vec{C}$ , is expressed in terms of the finite element mapping given by 9.5.31:

$$\begin{aligned}\vec{C} = \vec{F}^T \vec{F} &= \left[ \vec{I} + \left[ \sum_{j=1}^8 \frac{\partial N_j}{\partial \vec{X}} \vec{u}_j \right]^T \right] \left[ \vec{I} + \sum_{j=1}^8 \frac{\partial N_j}{\partial \vec{X}} \vec{u}_j \right] \\ &= \vec{I} + 2 \sum_{j=1}^8 \frac{\partial N_j}{\partial \vec{X}} \vec{u}_j + \left[ \sum_{j=1}^8 \frac{\partial N_j}{\partial \vec{X}} \vec{u}_j \right]^T \left[ \sum_{j=1}^8 \frac{\partial N_j}{\partial \vec{X}} \vec{u}_j \right] \\ &= \vec{I} + \left( 2 \sum_{j=1}^8 \frac{\partial N_j u_j^m}{\partial X_k} \right) \vec{e}_m \otimes \vec{e}_k \\ &\quad + \left( \sum_{j=1}^8 \frac{\partial N_j u_j^m}{\partial X_k} \right) \vec{e}_k \otimes \vec{e}_m \left( \sum_{j=1}^8 \frac{\partial N_j u_j^l}{\partial X_n} \right) \vec{e}_l \otimes \vec{e}_n \\ &= \vec{I} + \left( 2 \sum_{j=1}^8 \frac{\partial N_j u_j^m}{\partial X_k} \right) \vec{e}_m \otimes \vec{e}_k \\ &\quad + \left( \sum_{j=1}^8 \frac{\partial N_j u_j^m}{\partial X_k} \right) \left( \sum_{i=1}^8 \frac{\partial N_j u_j^m}{\partial X_n} \right) \vec{e}_k \otimes \vec{e}_n\end{aligned}\tag{9.5.31}$$

9.5.31 is substituted into 9.5.27 giving 9.5.32:

$$\begin{aligned}\vec{C} \cdot \vec{I} &= 3 + 2 \left( \sum_{j=1}^8 \frac{\partial N_j u_j^m}{\partial X_m} \right) + \left( \sum_{j=1}^8 \frac{\partial N_j u_j^m}{\partial X_m} \right) \left( \sum_{i=1}^8 \frac{\partial N_j u_j^m}{\partial X_m} \right) \\ &= 3 + 2 \left( \sum_{i=1}^8 \frac{\partial N_i u_i^m}{\partial X_m} \right) + \left( \sum_{i=1}^8 \frac{\partial N_i u_i^m}{\partial X_m} \right)^2\end{aligned}\tag{9.5.32}$$

The derivative of 9.5.31 is taken with respect to  $\vec{u}_k$  to produce 9.5.33:

$$\begin{aligned}
\frac{\partial(\vec{C} \cdot \vec{I})}{\partial \vec{u}_k} &= \frac{\partial(\vec{C} \cdot \vec{I})}{\partial u_k^m} \vec{e}_m \\
&= 2 \left( \frac{\partial N_k}{\partial X_m} \right) \vec{e}_m + 2 \left( \sum_{j=1}^8 \frac{\partial N_j u_j^m}{\partial X_m} \right) \left( \frac{\partial N_k}{\partial X_m} \right) \vec{e}_m
\end{aligned} \tag{9.5.33}$$

Note that 9.5.33 is in a vectorial form.

The next term to derive is  $\frac{\partial J}{\partial \vec{u}_k}$ . 9.5.34 is produced by using the chain rule and the time derivatives.

$$\begin{aligned}
\vec{v} = \dot{\vec{x}} &= \left( \dot{\vec{X}} + \dot{\vec{u}} \right) = \dot{\vec{u}} \\
\dot{J} &= \frac{\partial J}{\partial \vec{u}_k} \cdot \dot{\vec{u}}_k = J \operatorname{div}(\vec{v}) = J \operatorname{div}(\dot{\vec{u}}) = J \frac{\partial \dot{\vec{u}}}{\partial X_m} \cdot \vec{e}_m
\end{aligned} \tag{9.5.34}$$

Substituting 9.5.28 into 9.5.34, 9.5.35 is obtained.

$$\begin{aligned}
\dot{J} = \frac{\partial J}{\partial \vec{u}_k} \cdot \dot{\vec{u}}_k &= J \frac{\partial \dot{\vec{u}}}{\partial X_m} \cdot \vec{e}_m = J \frac{\partial \left( \sum_{j=1}^8 N_j \dot{u}_j \right)}{\partial X_m} \cdot \vec{e}_m \\
&= \left[ J \frac{\partial \left( \sum_{j=1}^8 N_j \right) \vec{e}_m}{\partial X_m} \right] \cdot \dot{\vec{u}}_k
\end{aligned} \tag{9.5.35}$$

The expression  $\frac{\partial J}{\partial \vec{u}_k}$  can be derived from 9.5.35, shown in 9.5.36.

$$\frac{\partial J}{\partial \vec{u}_k} = \left[ J \frac{\partial \left( \sum_{j=1}^8 N_j \right)}{\partial X_m} \right] \vec{e}_m \tag{9.5.36}$$

Combining 9.5.36 and 9.5.33 into 9.5.30, 9.5.37 is obtained.

$$\begin{aligned}
\frac{\partial I_1}{\partial \vec{u}_k} &= \left\{ -\frac{1}{9} J^{-5/3} \left[ J \left( \sum_{j=1}^8 \frac{\partial N_j}{\partial X_m} \right) \right] (\vec{C} \cdot \vec{I}) \right. \\
&\quad \left. + 2 J^{-2/3} \left[ \left( \frac{\partial N_k}{\partial X_m} \right) + \left( \sum_{j=1}^8 \frac{\partial N_j u_j}{\partial X_m} \right) \left( \frac{\partial N_k}{\partial X_m} \right) \right] \right\} \vec{e}_m(X) \\
&= \left\{ -\frac{1}{9} \left[ \left( \sum_{j=1}^8 \frac{\partial N_j}{\partial X_k} \right) \right] (\vec{C} \cdot \vec{I}) \right. \\
&\quad \left. + 2 \left[ \left( \frac{\partial N_k}{\partial X_m} \right) + \left( \sum_{j=1}^8 \frac{\partial N_j u_j}{\partial X_m} \right) \left( \frac{\partial N_k}{\partial X_m} \right) \right] \right\} J^{-2/3} \vec{e}_m
\end{aligned} \tag{9.5.37}$$

9.5.37 is substituted into the complete form of the strain energy  $U$ , and taking its derivatives with respect to  $\vec{u}_k$ , and using the Gauss-Legendre integration method, 9.5.38 is obtained.

$$\begin{aligned}
\frac{\partial U}{\partial \vec{u}_k} &= C_1 \sum_{i=1}^8 \left( \frac{\partial I_1}{\partial \vec{u}_k} d \right)_{r=r_i, s=s_i, t=t_i} \det(\vec{J}_i) \\
&= C_1 \sum_{i=1}^8 d \left\{ -\frac{1}{9} \left[ \left( \sum_{j=1}^8 \frac{\partial N_j}{\partial X_m} \right) \right] (\vec{C} \cdot \vec{I}) \right. \\
&\quad \left. + 2 \left[ \left( \frac{\partial N_k}{\partial X_m} \right) + \left( \sum_{j=1}^8 \frac{\partial N_j u_j}{\partial X_m} \right) \left( \frac{\partial N_k}{\partial X_m} \right) \right] \right\} J_{r=r_i, s=s_i, t=t_i}^{-2/3} \det(\vec{J}_i) \vec{e}_m
\end{aligned} \tag{9.5.38}$$

All of the components of 9.5.38 have been developed for a single cell with respect to the displacements applied to the nodes. In order to work directly with the coordinates of the new merged nodes, the relation between them is defined in 9.5.39, where  $\vec{x}_m$  corresponds to the new coordinates of node  $m$ .

$$\begin{aligned}
\vec{u}_k^1 &= \vec{x}_m - \vec{X}_k^1 \\
\vec{u}_k^2 &= \vec{x}_m - \vec{X}_k^1
\end{aligned} \quad m = 1, \dots, 4 \tag{9.5.39}$$

The total energy gradient with respect to  $\vec{x}_m$  is given by 9.5.40:

$$\begin{aligned}
\frac{\partial U^{total}}{\partial \vec{x}_m} &= \frac{\partial U}{\partial \vec{u}_k^1} \frac{\partial \vec{u}_k^1}{\partial \vec{x}_m} + \frac{\partial U}{\partial \vec{u}_k^2} \frac{\partial \vec{u}_k^2}{\partial \vec{x}_m} \\
&= \frac{\partial U}{\partial \vec{u}_k^1} + \frac{\partial U}{\partial \vec{u}_k^2}
\end{aligned} \tag{9.5.40}$$

Figure 9.2 contains pseudocode outlining each step in the algorithm.

## 9.6 Results

The adhesion method introduced here is capable of successfully adhering cells regardless of their arrangement and is completely general, unlike heuristic rules.

To illustrate the method, three examples were selected, shown in Figures 9.3, 9.4, and 9.5. The portion of each figure identified with the letter "a" illustrates cells selected for adhesion after testing for proximity or intersection. The portion of each figure identified with the letter "c" contains a chart showing the distribution of strain energy among the 24 possible initial adhesion configurations. The  $y$ -axis indicates the strain energy corresponding to the configuration number on the  $x$ -axis. The configuration with the minimum energy has been selected as the initial state of adhesion. This state sets the initial location of the new merged nodes, which are located midway between their original

```

1  Select Faces to Merge;
2      Inverse map the signal and the cell into the reference configuration;
3      Find the faces with the maximum signal gradient generated by
      the adjacent cell;
4  Select Configuration;
5      Compute the initial displacements of the merged nodes for ev-
      ery pairing configuration;
6      Compute the change in strain energy for every configuration;
7      Select the configuration that results in the minimum change in
      strain energy to the merged cells;
8  Conjugate Gradient Search;
9      For all 4 merged nodes:
10         Compute the strain energy gradient with respect to the co-
            ordinates of a single node;
11         Displace the node in the direction of the conjugate gradient;
12         Compute the total strain energy;
13         If the total strain energy has reached a plateau:
14             then stop;
15             else return to line 9;

```

Figure 9.2: Pseudocode outline of the algorithm.

positions.

The portion of each figure identified with the letter "d" illustrates the energy optimization process through the conjugate gradient method. The  $y$ -axis corresponds to the strain energy while the  $x$ -axis corresponds to the number of iterations it takes to reach global energy minimization. Notice that the number of iterations is relatively small. This is because only the lowest energy configuration of merged nodes is selected as the starting point for the optimization. The high rate of convergence is also a direct result of the closed-form analytical expression derived for the energy gradient.

The portion of each figure identified with the letter b) presents the final configuration of the cells after the adhesion process. The distortion of the cells in order to achieve adhesion is minimal, and results in elements that are suitable for further computation. It can also be seen that the method is robust in that it successfully merges nodes and adheres cells in a variety of initial arrangements.

## 9.7 Application

The method described above has been used as an integral part of an evolutionary computational technique for design synthesis of three dimensional structures [92, 93]. The evolved structures are composed of cells and each cell is a hexahedral finite element. During the growth procedure, cells

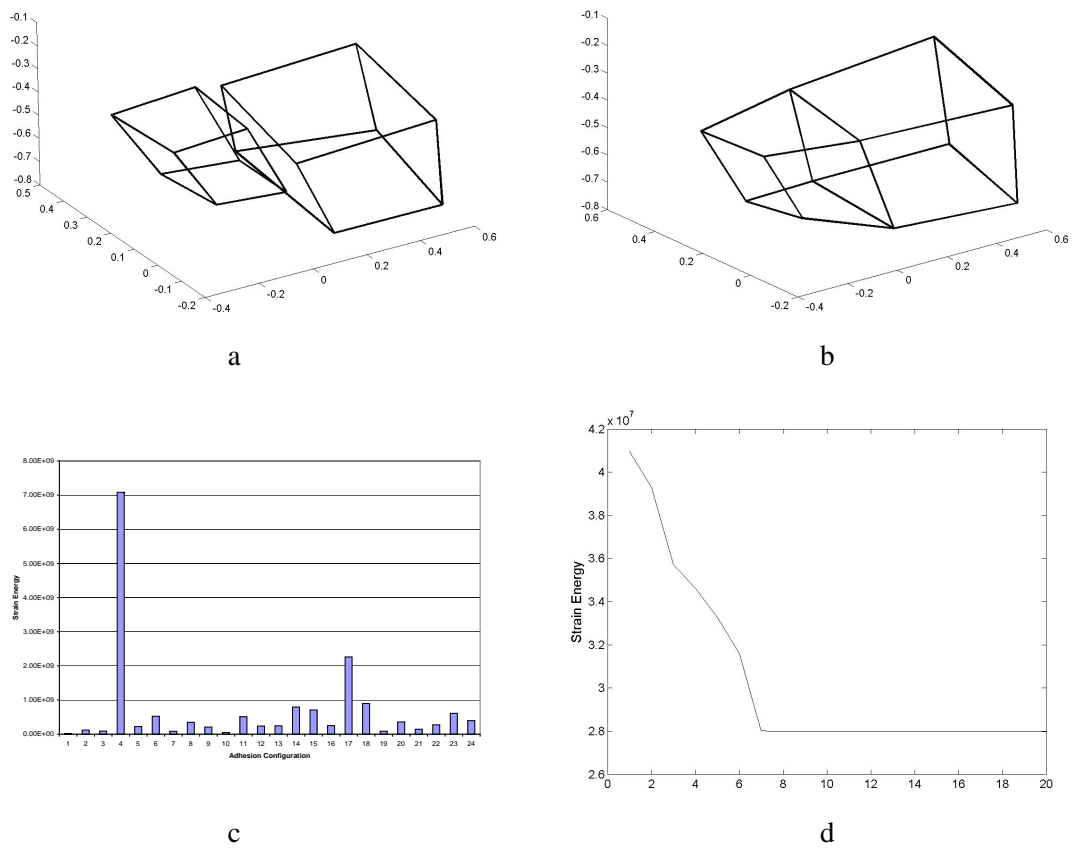


Figure 9.3: Example 1. (a) Two non-overlapping but nearby cells prior to adhesion. (b) The two cells subsequent to adhesion. (c) The energy states of all 24 possible adhered configurations. The minimum one is selected. (d) The strain energy minimization using the conjugate gradient method.

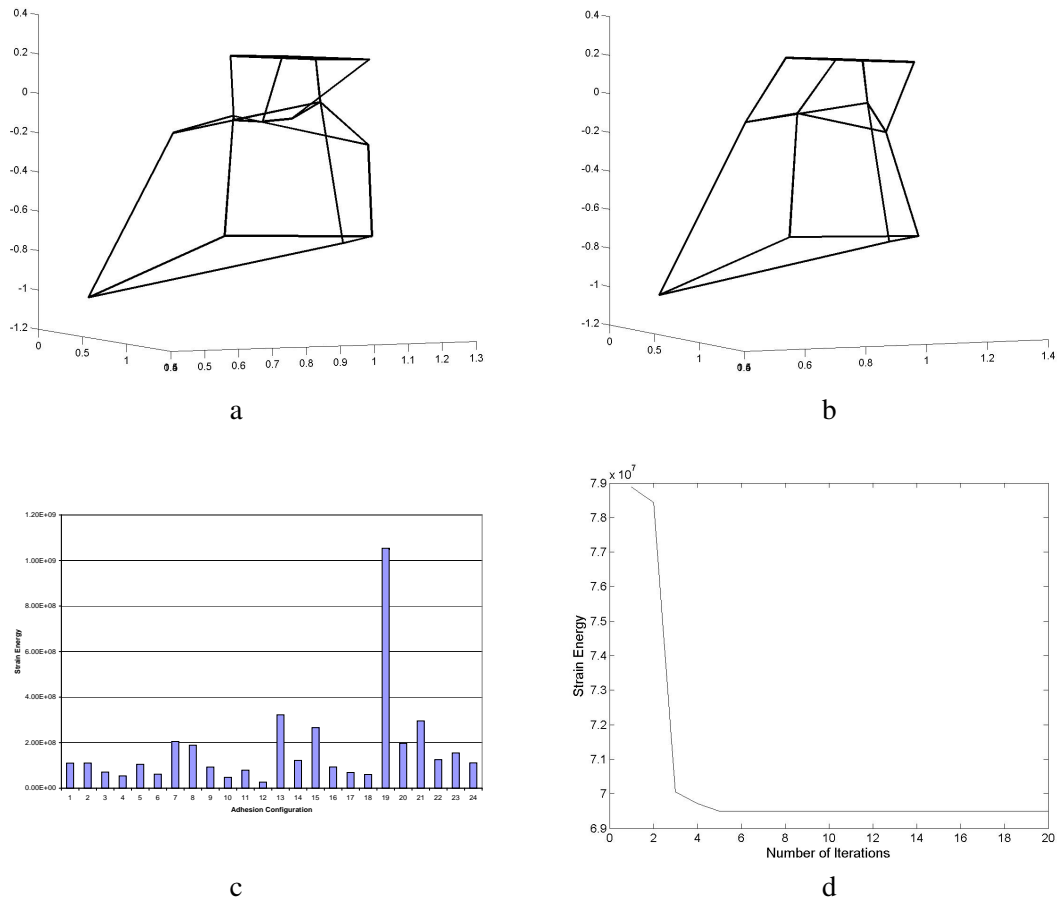


Figure 9.4: Example 2. (a) Two cells prior to adhesion. (b) The two cells subsequent to adhesion. (c) The energy states of all 24 possible adhered configurations. The minimum one is selected. (d) The strain energy minimization using the conjugate gradient method.



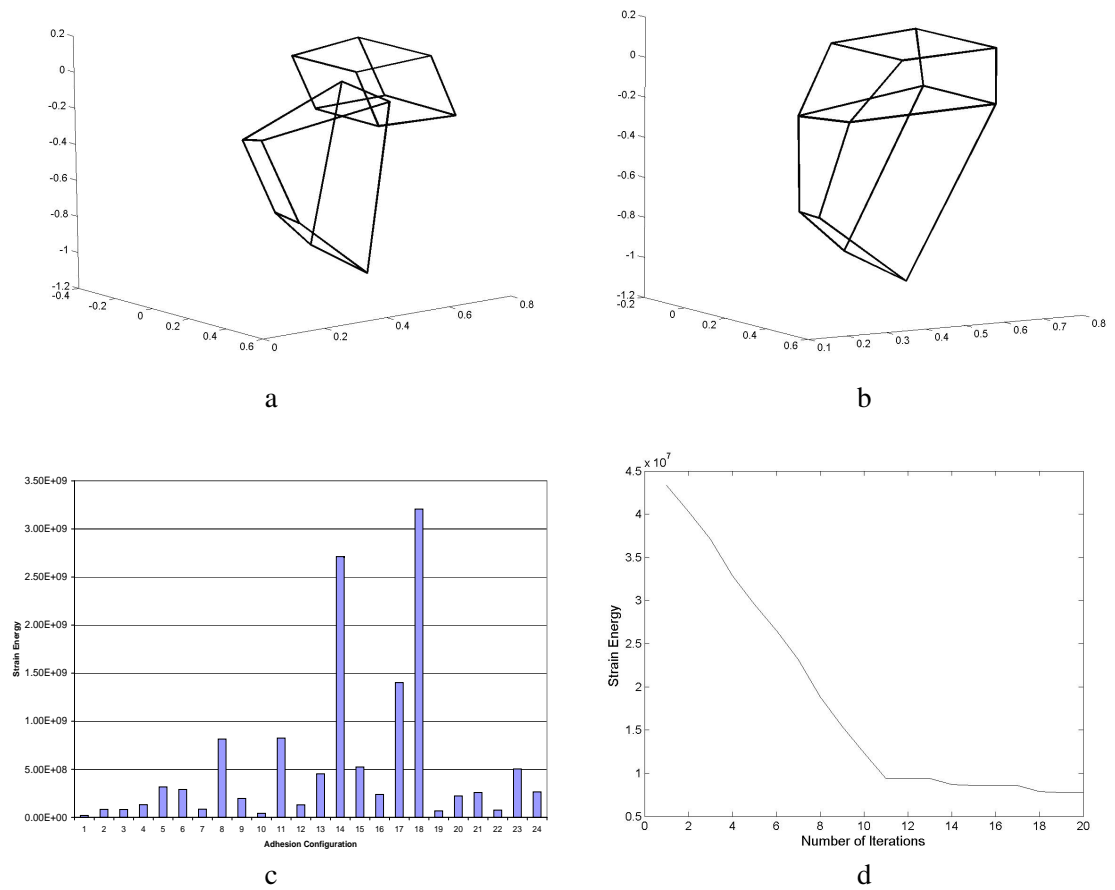


Figure 9.5: Example 3. (a) Two intersecting cells prior to adhesion. (b) The two cells subsequent to adhesion. (c) The energy states of all 24 possible adhered configurations. The minimum one is selected. (d) The strain energy minimization using the conjugate gradient method.

adhere after being tested for geometric interference with nearby cells. The growth process contains stochastic elements that deform the cells and make their orientation arbitrary and unpredictable. Due to the randomness of the orientations of the cells, an efficient robust adhesion tool was needed.

## 9.8 Conclusions

The research reported here demonstrates that by simulating the interdependent processes of evolution and development, robust high performance design configurations can be synthesized *in silico* purely in response to engineering criteria. A simple set of *if-conditional then-action* rule elements can be evolved (in concert with the development of each evolved rule set into a phenotype, and the evaluation of the performance of each phenotype) into sets of rules (genotypes) that will develop into phenotypes (by executing the rules) that meet stated performance criteria. The results shown here demonstrate that the interplay of evolution and development can produce useful results, when guided only by evaluations of how well phenotypes meet the performance criteria.

Interestingly, some of the configurations synthesized by this method exhibit characteristics of modularity (distinct structural elements, both internally, inhomogeneous structures and externally). These characteristics emerged spontaneously, without any established preference or requirement. The evolutionary process synthesizes solutions (genomes) which overcome these random effects and can repeatedly produce high performance phenotypes. This indicates that the environment and its variability play a major role in the evolution and development of robust high-performance phenotypes.

We have also found a way to overcome an important problem in generative development; growth control. Our method mimics three control processes from biology (diseases, thermodynamic balancing and control genes). The utilization of these processes lead to evolution of phenotypes that not only performed well under the environment but also lead to growth and development via a controlled process. Our results indicate a high correlation between the performance of the phenotype regarding the environment and its ability to grow in a stable manner.

We have also shown two different kinds of robustness; growth robustness and phenotype robustness, which have been evolved simultaneously using a model of embryogenesis of 3-D structures. Phenotype robustness is a direct outcome of variation in the environment during the evaluation scheme, while growth robustness relates to both the variability in the environment and the gene execution process. We have shown that the configuration of phenotypes which have been subjected

to an unvarying environment during their growth and their evaluation process were regular and relatively less complex, and phenotypes that have been evolved under a stochastically varying environment were more complex and modular. In both experiments the gene execution process was the same.

We have shown that the environment in our model is the most crucial element which effect both the growth the final topology of the phenotypes.

From a practical mesh generation method, we have shown a novel approach for an adhesion process between any two 3-D finite elements. Using first principles, including minimum potential energy and cell signaling, a model was developed that adheres any two finite elements, regardless of their orientation to one another or their deformation. This model may be used in any application that requires the adhesion of cells or finite elements. One of the major applications that may utilize this approach is the *advancing front* method for generating 2-D or 3-D meshes of convex and non-convex bodies. The major advantage of our approach is the absence of any heuristic rules commonly used in these problems. The approach introduced here imitates nature and works smoothly and robustly, and produces well-conditioned merged meshes.

From a mathematical stand point we have shown that a gene duplication mechanism may slow the convergent rate of the evolutionary process comparing to suppressing this effect. Nevertheless, phenotypes which have been evolved on both methods, had similar patterns such as symmetry and modularity.

The last and most important aspect of this work is our ability to explore and understand a plausible relationship between the topology of the evolved phenotypes and their genome, which were driven by the environment. We have found similar relationships between these three elements, which have been observed in nature. One of these relationships is the role of a gene in nature or a word in our model. We have found words which were essential to the growth process, words which were meaningless (DNA trash) and words which serve as a backup mechanism to other words. We have also find that our evolutionary process, may reused words which have been found in the early stages of evolution in order to create more complex phenotypes during late stages of evolution. This kind of mechanism has been observed in nature and described in the literature as "toolkit gene".

## **9.9 Future Work**

Future work will include studying the structure of the evolved genomes and the configurations of the phenotypes they produce. The work will focus on the relationships between evolution and development and the rules and the modularity they produce with the goal of better understanding the relationship between the evolved genome and the form of the phenotype.

# Bibliography

- [1] CARROLL, S. B. *Endless Forms Most Beautiful: The new science of evo devo and the making of the animal kingdom*. W. W. Norton & Co., 2005.
- [2] PALMER, A. R. Symmetry breaking and the evolution of development. *Science* 306, 5697 (Oct. 2004), 828–833.
- [3] STERELNY, K., AND KITCHER, P. The return of the gene. *Journal of Philosophy* 85, 7 (July 1988), 339–361.
- [4] STOLTZFUS, A. Mutationism and the dual causation of evolutionary change. *Evolution & Development* 8, 3 (2006), 304–317.
- [5] CARROLL, S. B., GRENIER, J. K., AND WEATHERBEE, S. D. *From DNA to Diversity: Molecular Genetics and the Evolution of Animal Design*, 2<sup>nd</sup> ed. Blackwell Publishing, Malden, MA, 2004.
- [6] BENTLEY, P. J., AND KUMAR, S. Three ways to grow designs: A comparison of embryogenies for an evolutionary design problem. In *Proceedings of the Genetic and Evolutionary Computation Conference* (Orlando, FL, 1999), W. Banzhaf, J. Daida, A. E. Eiben, M. H. Garzon, V. Honavar, M. Jakiela, and R. E. Smith, Eds., Morgan Kaufmann, pp. 35–43.
- [7] Skaar, J., Wang, L., and Erdogan, T., 2001. “Synthesis of thick optical thin-film filters with a layer-peeling inverse-scattering algorithm”. *Optical Society of America*, **40**(3).
- [8] Yang, J.-M., and Kao, C.-Y., 2001. “Efficient evolutionary algorithm for the thin-film synthesis of inhomogeneous optical coatings”. *Applied Optics*, **40**(19).
- [9] Ryu, W., Fasching, R. J., Vyakarnam, M., Greco, R. S., and Prinz, F. B., 2006. “Microfabrication technology of biodegradable polymers for interconnecting microstructures”. *IEEE/ASME Journal of Microelectromechanical Systems*, **15**(6), Dec., pp. 1457–1465.

- [10] BENDSØE, M. P., AND SIGMUND, O. *Topology Optimization: Theory, Methods and Applications*, 2<sup>nd</sup> ed. Springer-Verlag, Berlin, Heidelberg, 2003.
- [11] LIPSON, H., AND POLLACK, J. B. Automatic design and manufacture of robotic lifeforms. *Nature* 406 (Aug. 2000), 974–978.
- [12] POLLACK, J. B., LIPSON, H., HORNBY, G., AND FUNES, P. Three generations of automatically designed robots. *Artificial Life* 7, 3 (2001), 215–223.
- [13] ZYKOV, V., MYTILINAIOS, E., ADAMS, B., AND LIPSON, H. Self-reproducing machines. *Nature* 435 (May 2005), 163–164.
- [14] HOLLAND, J. H. *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, Ann Arbor, MI, 1975.
- [15] EGGENBERGER, P. Evolving morphologies of simulated 3D organisms based on differential gene expression. In *Fourth European Conference on Artificial Life*, P. Husbands and I. Harvey, Eds. MIT Press, 1997, pp. 205–213.
- [16] MACREADY, W. G., SIAPAS, A. G., AND KAUFFMAN, S. A. Criticality and parallelism in combinatorial optimization. *Science* 271, 5245 (Jan. 1996), 56–59.
- [17] BOWERS, C. P. Evolving robust solutions with a computational embryogeny. In *Proceedings of the UK Workshop on Computational Intelligence: UKCI-2003* (Bristol, UK, 2003), M. Rossiter and T. Martin, Eds., University of Bristol.
- [18] BOWERS, C. P. Simulating evolution with a computational model of embryogeny: Obtaining robustness from evolved individuals. In *Advances in Artificial Life, Proceeding of the 8th European Conference on Artificial Life: ECAL 2005* (2005), M. S. Capcarrere, A. A. Freitas, P. J. Bentley, C. G. Johnson, and J. Timmons, Eds., Springer-Verlag, pp. 149–158.
- [19] KUMAR, S., AND BENTLEY, P. J. Computational embryology: Past, present and future. In *Advances in Evolutionary Computing, Theory and Applications*, A. Ghosh and S. Tsutsui, Eds. Springer-Verlag, New York, 2003, pp. 461–478.
- [20] BOWERS, C. P. Formation of modules in a computational model of embryogeny. In *Proceedings of the 2005 Congress on Evolutionary Computation (CEC'05)* (Piscataway, NJ, Sept. 2005), vol. 1, IEEE Press, pp. 537–542.

- [21] LIPSON, H., POLLACK, J. B., AND SUH, N. P. On the origin of modular variation. *Evolution* 56, 8 (Aug. 2002), 1549–1556.
- [22] KASHTAN, N., AND ALON, U. Spontaneous evolution of modularity and network motifs. *Proceedings of the National Academy of Sciences* 102, 39 (Sept. 2005), 13773–13778.
- [23] MAES, P. Modeling adaptive autonomous agents. *Artificial Life*, 1 (1&2), 9 (1994).
- [24] VAARIO, J., AND SHIMOHARA, K. Synthesis of developmental and evolutionary modeling of adaptive autonomous agents. *Lecture Notes in Computer Science, ICANN97* (1997).
- [25] VAARIO, J., HORI, K., AND OHSUGA, S. Toward evolutionary design of autonomous systems. *The International Journal in Computer Simulation* (1994).
- [26] VAARIO, J. The role of environmental adaptation in evolutionary computation.
- [27] KITANO, H. Designing neural networks using genetic algorithms with graph generation system. *Complex Systems* 4, 4 (1990), 461–476.
- [28] KITANO, H. A simple model of neurogenesis and cell differentiation based on evolutionary large-scale chaos. *Artificial Life* 2, 1 (1994), 79–99.
- [29] KOWALIW, T., GROGONO, P., AND KHARMA, N. Environment as a spatial constraint on the growth of structural form. In *GECCO '07: Proceedings of the 9th annual conference on Genetic and evolutionary computation* (New York, NY, USA, 2007), ACM, pp. 1037–1044.
- [30] KOWALIW, T., GROGONO, P., AND KHARMA, N. The evolution of structural design through artificial embryogeny. *Artificial Life, 2007. ALIFE '07. IEEE Symposium on* (April 2007), 425–432.
- [31] STEINER, T., JIN, Y., AND SENDHOFF, B. A cellular model for the evolutionary development of lightweight material with an inner structure. In *Proceedings of the 10<sup>th</sup> annual conference on Genetic and Evolutionary Computation (GECCO '08)* (New York, NY, USA, 2008), M. Keijzer, Ed., Association for Computing Machinery, ACM, pp. 851–858.
- [32] FUNES, P., AND POLLACK, J. Computer evolution of buildable objects. In *Fourth European Conference on Artificial Life*, P. Husbands and I. Harvey, Eds. MIT Press, 1997, pp. 358–367.

- [33] PRINZ, F. B., AND WEISS, L. E. Novel applications and implementations of shape deposition manufacturing. *Naval Research Reviews, Office of Naval Research L*, 3 (1998), 19–26.
- [34] SENDHOFF, B. Evolutionary optimised ontogenetic neural networks with incremental problem complexity during development. In *Proceedings of the 2000 Congress on Evolutionary Computation* (July 2000), vol. 2, IEEE, pp. 1443–1450.
- [35] ALBERTS, B., JOHNSON, A., LEWIS, J., RAFF, M., ROBERTS, K., AND WALTER, P. *Molecular Biology of the Cell*, 4<sup>th</sup> ed. Garland Publishing, New York, London, 2002.
- [36] SHERR, C. J. Principles of tumor suppression. *Cell* 116 (Jan. 2004), 235–246.
- [37] ZIENKIEWICZ, O. C., AND TAYLOR, R. L. *The Finite Element Method*, 6<sup>th</sup> ed. Butterworth-Heinemann, Elsevier, Oxford, 2005.
- [38] PUECH, P., TAUBENBERGER, A., ULRICH, F., KRIEG, M., MULLER, D. J., AND HEISENBERG, C. Measuring cell adhesion forces of primary gastrulating cells from zebrafish using atomic force microscopy. *Journal of Cell Science* 118 (2005), 4199–4206.
- [39] TAN, W. S., AND CHEN, Y. L. Quantitative investigations of cell-bubble interactions using a foam fractionation technique. *Cytotechnology* 15, 1-3 (1994), 321–328.
- [40] COEN, E., ROLLAND-LAGAN, A., MATTHEWS, M., BANGHAM, J. A., AND PRUSINKIEWICZ, P. The genetics of geometry. *Proceedings of the National Academy of Sciences* 101, 14 (Apr. 2004), 4728–4735.
- [41] ZHANG, Y., BAJAJ, C., AND SOHN, B.-S. 3D finite element meshing from imaging data. *Computer Methods in Applied Mechanics and Engineering* 194 (Nov. 2005), 5083–5106.
- [42] PEDERSEN, C. B. W., BUHL, T., AND SIGMUND, O. Topology synthesis of large-displacement compliant mechanisms. *International Journal for Numerical Methods in Engineering* 50 (2001), 2683–2705.
- [43] HUFNAGEL, L., TELEMAN, A. A., ROUAULT, H., COHEN, S. M., AND SHRAIMAN, B. L. On the mechanism of wing size determination in fly development. *Proceedings of the National Academy of Sciences* 104, 10 (Mar. 2007), 3835–3840.



- [44] WERTZ, X., SCHOËVAËRT, D., MAITOURNAM, H., CHASSIGNET, P., AND SCHWARTZ, L. The effect of hormones on bone growth is mediated through mechanical stress. *Comptes Rendus Biologies* 329, 2 (Feb. 2006), 79–85.
- [45] SHRAIMAN, B. I. Mechanical feedback as a possible regulator of tissue growth. *Proceedings of the National Academy of Sciences* 102, 9 (Mar. 2005), 3318–3323.
- [46] DAY, C. Mechanical force may determine the final size of tissues. *Physics Today* 60, 4 (Apr. 2007), 20–21.
- [47] WEST, G. B., BROWN, J. H., AND ENQUIST, B. J. Growth models based on first principles or phenomenology? *Functional Ecology* 18, 2 (Apr. 2004), 188–196.
- [48] KLEIBER, M. Body size and metabolism. *Hilgardia* 6 (1932), 315–353.
- [49] KLEIBER, M. Body size and metabolic rate. *Physiological Reviews* 27, 4 (1947), 511–541.
- [50] KNUPP, P. M. A method for hexahedral mesh shape optimization. *International Journal for Numerical Methods in Engineering* 58, 2 (Sept. 2003), 319–332.
- [51] FREITAG, L. A., AND KNUPP, P. M. Tetrahedral mesh improvement via optimization of the element condition number. *International Journal for Numerical Methods in Engineering* 53, 6 (2002), 1377–1391.
- [52] KNUPP, P. M. Hexahedral and tetrahedral mesh untangling. *Engineering with Computers* 17, 3 (Oct. 2001), 261–268.
- [53] KNUPP, P. M. Achieving finite element mesh quality via optimization of the jacobian matrix norm and associated quantities. Part II - A framework for volume mesh optimization and the condition number of the Jacobian matrix. *International Journal for Numerical Methods in Engineering* 48, 8 (July 2000), 1165–1185.
- [54] KNUPP, P. M. Achieving finite element mesh quality via optimization of the jacobian matrix norm and associated quantities. Part I - A framework for volume mesh optimization. *International Journal for Numerical Methods in Engineering* 48, 3 (May 2000), 401–420.
- [55] FONSECA, C. M., AND FLEMING, P. J. Multiobjective evolutionary algorithms: A comparable case study and the strength Pareto approach. *Evolutionary Computation* 3, 1 (1995), 257–271.

- [56] SRINIVAS, N., AND DEB, K. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation* 2, 3 (1994), 221–248.
- [57] ZITZLER, E., AND THIELE, L. An overview of evolutionary algorithms in multiobjective optimization. *IEEE transaction on evolutionary computation* 3, 1 (1999), 257–271.
- [58] SCOTT, M. J., AND ANTONSSON, E. K. Aggregation Functions for Engineering Design Trade-offs. *Fuzzy Sets and Systems* 99, 3 (1998), 253–264.
- [59] MA, L., AND ANTONSSON, E. K. Robust Mask-Layout and Process Synthesis. *Journal of Microelectromechanical Systems* 12, 4 (Aug. 2003).
- [60] JU, M.-Y., LIU, J.-S., SHIANG, S.-P., CHIEN, Y.-R., HWANG, K.-S., AND LEE, W.-C. A novel collision detection method based on enclosed ellipsoid. *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation* 3 (2001), 2897–2902.
- [61] ZIENKIEWICZ, O. C., AND TAYLOR, R. L. *Finite Element Method: Volume 2, Solid Mechanics*. Elsevier, 2000.
- [62] FLORY, P. J. Thermodynamic relations for high elastic materials. *Transactions of the Faraday Society* 57 (1961), 829–838.
- [63] HAGER, W. W., AND ZHANG, H. A conjugate gradient method with guaranteed descent and an efficient line search. *SIAM Journal on Optimization* 16, 1 (2005), 170–192.
- [64] HAGER, W. W., AND ZHANG, H. CG Descent, A conjugate gradient method with guaranteed descent. *ACM Transactions on Mathematical Software (TOMS)* 32, 1 (Mar. 2006), 113–137.
- [65] FLETCHER, R., AND REEVES, C. M. Function minimization by conjugate gradients. *The Computer Journal* 7, 2 (1964), 149–154.
- [66] ZHANG, J., AND XU, C. Properties and numerical performance of quasi-Newton methods with modified quasi-Newton equations. *Journal of Computational and Applied Mathematics* 137, 2 (Dec. 2001), 269–278.
- [67] KUMAR, S., AND BENTLEY, P. J. Implicit evolvability: An investigation into the evolvability of an embryo. In *Proceedings of the Genetic and Evolutionary Computation Conference* (Las Vegas, NV, 2000), Morgan Kaufmann.

- [68] KUMAR, S., AND BENTLEY, P. J. *On Growth, Form and Computers*. Academic Press, New York, 2003.
- [69] BOWERS, C. P., AND BULLINARIA, J. A. Embryological modelling of the evolution of neural architecture. In *Modeling Language, Cognition and Action* (Singapore, 2005), A. Cangelosi, G. Bugmann, and R. Borisjuk, Eds., World Scientific, pp. 375–384.
- [70] , J. ARJAN AND G. M. VISSER AND JOACHIM HERMISSON AND GENTER P. WAGNER AND ANCEL MEYERS AND HOMAYOUN BAGHERI-CHAICHIAN AND JEFFREY L. BLANCHARD AND LIN CHAO AND JAMES M. CHEVERUD AND SANTIAGO F. ELENA AND WALTER FONTANA AND GREG GIBSON AND F. HANSEN DAVID KRAKAUER AND GEORGE VON DASSOW AND ANDREAS WAGNER AND MICHAEL C. WHITLOCK EVOLUTION OF ROBUSTNESS PERSPECTIVE: EVOLUTION AND DETECTION OF GENETIC ROBUSTNESS. In *Evolution, international journal of organic evolution* (September 2003), pp. 1959-1972.
- [71] SUSUMU OHNO *Evolution by gene duplication*. Springer-Verlag, 1970.
- [72] GERHARD SCHLOSSER, PETER P. WAGNER *Modularity in Development and Evolution*. Springer-Verlag, 2000.

- [73] BOWERS, C. P. *Simulating Evolution with a Computational Model of Embryogeny*. PhD thesis, University of Birmingham, 2006.
- [74] ALBERTS, B., JOHNSON, A., LEWIS, J., RAFF, M., ROBERTS, K., AND WALTER, P. *Molecular Biology of the Cell*. Garland Science, Taylor & Francis, 2002.
- [75] BAKER, T. J. Automatic mesh generation for complex three-dimensional regions using a constrained Delaunay triangulation. *Engineering with Computers* 5, 3-4 (June 1989), 161–175.
- [76] EHMANN, S. A., AND LIN, M. C. Accurate and fast proximity queries between polyhedra using convex surface decomposition. In *EG 2001 Proceedings*, A. Chalmers and T.-M. Rhyne, Eds., vol. 20(3). Blackwell Publishing, 2001, pp. 500–510.
- [77] FLETCHER, R., AND REEVES, C. M. Function minimization by conjugate gradients. *The Computer Journal* 7, 2 (1964), 149–154.
- [78] FLORY, P. J. Thermodynamic relations for high elastic materials. *Transactions of the Faraday Society* 57 (1961), 829–838.
- [79] GEORGE, P.-L., HECHT, F., AND SALTEL, E. Automatic mesh generator with specified boundary. *Computer Methods in Applied Mechanics and Engineering* 92, 3 (Nov. 1991), 269–288.
- [80] HAGER, W. W., AND ZHANG, H. A conjugate gradient method with guaranteed descent and an efficient line search. *SIAM Journal on Optimization* 16, 1 (2005), 170–192.
- [81] HAGER, W. W., AND ZHANG, H. CG Descent, A conjugate gradient method with guaranteed descent. *ACM Transactions on Mathematical Software (TOMS)* 32, 1 (Mar. 2006), 113–137.
- [82] JANMEY, P. A., AND DISCHER, D. E. Developmental biology: Holding it together in the eye. *Nature* 431 (Oct. 2004), 635–636.
- [83] LAWSON, C. L. Software for  $C^1$  surface interpolation. In *Mathematical Software III*, J. R. Rice, Ed. Academic Press, New York, 1977, pp. 161–194.
- [84] LO, S. H. Volume discretization into tetrahedra - II. 3D triangulation by advancing front approach. *Computers and Structures* 39, 5 (1991), 501–511.
- [85] LÖHNER, R. Progress in grid generation via the advancing front technique. *Engineering with Computers* 12, 3-4 (Sept. 1996), 186–210.
- [86] SHEPHARD, M. S., AND GEORGES, M. K. Automatic three-dimensional mesh generation by the finite octree technique. *International Journal for Numerical Methods in Engineering* 32, 4 (1991), 709–749. Special Issue: Adaptive Meshing. Issue Edited by Olgierd C. Zienkiewicz, Richard H. Gallagher, Ronald W. Lewis.

- [87] SHIN, D., AND TIAHJADI, T. Triangular mesh generation of octrees of non-convex 3D objects. In *Proceedings of the 18<sup>th</sup> International Conference on Pattern Recognition* (2006), vol. 3, IEEE Computer Society, pp, 950–953.
- [88] SPENCER, A. J. M. *Continuum Mechanics*. Dover, New York, 2004.
- [89] STASA, F. L. *Applied Finite Element Analysis for Engineers*. Oxford University Press, 1979.
- [90] WEATHERILL, N. P., AND HASSAN, O. Efficient three-dimensional Delaunay triangulation with automatic point creation and imposed boundary constraints. *International Journal for Numerical Methods in Engineering* 37, 12 (1994), 2005–2039.
- [91] YERRY, M. A., AND SHEPHARD, M. S. Automatic three-dimensional mesh generation by the modified octree technique. *International Journal for Numerical Methods in Engineering* 20 (1984), 1965–1990.
- [92] YOGEV, O., AND ANTONSSON, E. K. A Novel Synthesis Design Approach for Continuous Inhomogeneous Structures. In *19<sup>th</sup> International Conference on Design Theory and Methodology (DTM)* (Sept. 2007), ASME. Paper Number: DETC2007/DTM-35662.
- [93] YOGEV, O., AND ANTONSSON, E. K. Growth and Development of Continuous Structures. In *GECCO 2007, Genetic and Evolutionary Computation Conference* (London, UK, 2007), pp. 1064–1065.
- [94] YUAN, K. Y., HUANG, Y.-S., YANG, H.-T., AND PIAN, T. H. H. The inverse mapping and distortion measures for 8-node hexahedral isoparametric elements. *Computational Mechanics* 14, 2 (May 1994), 189–199.
- [95] ZHANG, J., AND XU, C. Properties and numerical performance of quasi-Newton methods with modified quasi-Newton equations. *Journal of Computational and Applied Mathematics* 137, 2 (Dec. 2001), 269–278.
- [96] ZIENKIEWICZ, O. C., AND TAYLOR, R. L. *Finite Element Method: Volume 2, Solid Mechanics*. Elsevier, 2000.