INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 05 MELBOURNE, AUGUST 15 –18, 2005

THE DESIGN OF SELF EVOLVING BEHAVIOUR: SELF-EVOLUTION I

W. Hollier [†], C. Burvill [§] and J. Davoren [§]

Abstract

Advanced mechanical technology capable of self-organization, self-assembly, self-replication (self evolving behaviour = SEB) facilitating self repair, environment adaption and recycling requires new approaches to the analysis and design of systems capable of such complex behaviour. Cellular evolution, mechanical self-replication and number system evolution are studied and compared and two new techniques – extensible generative grammars with closure, and endo-exo symmetry transformation analysis – introduced that can be applied to the analysis of SEB and complexity theory. These techniques could also form the basis of advanced engineering design and analysis environments.

Keywords: Self-Organization, Self-Assembly, Self-Evolving Behaviour, Environmental Design

1. Introduction

Advanced mechanical technology with life-like capabilities such as self-repair, automatic production, adaption to environment and recycling of materials offers environmental, social and economic advantages. Partial solutions have been developed for the design of such life-like technologies, from which we have learnt that the initial production of such technology can proceed from existing machines. The first problem to be solved is not how to build but how to design self-evolving mechanistic systems. We postulate this requires new design methods to extend design to higher order systems with multi-level nested self-reference.

Numerous studies in various fields (complexity, automata theory, physics, chemistry, biology, cell-biology) have addressed single characteristics of life. These characteristics encompass self-reference (self-description, self-recognition, immunity), self-construction (self-organization, self-assembly, self-repair), self-directed change (eg development / ontogeny, growth) and, learning and optimisation (eg. adaption, evolution), which similarly require closure via feedback.

This paper seeks to identify a basis for the integrated study of all such behaviour and will use the term ‘self-evolving behaviour’ (SEB) to refer to the integrated life processes. This foundation, a common basis for modelling these processes, must also support the design process in particular for mechanistic systems.

To design self-evolving systems calls for novel, innovative modes of analysis and design that are beyond existing methods. It is postulated that the solution of this problem is through the use of general systems characteristics, requiring modularity and subtle forms of symmetry (for example, complement, similarity and enantiomorphism) [1, 2].

[†] EnGen Institute and [§] The University of Melbourne

2. Method

There is a venerable tradition of applying numerical traits to design (eg. the golden ratio, fibonnacci numbers and fractals) [1]. These same numerical traits have been identified in nature, thus introducing natural characteristics into design and establishing an analogy between nature and design. This design method is most often found in two-dimensional decoration [2], but is also found in three-dimensional forms (eg Platonic and Archimedean solids) and occasionally in functional design (eg Archimedean spiral used to pump water).

Our method will establish a similar analogy between natural, mathematical and mechanistic systems having self-evolving behaviour. This ‘natural model’ is derived from a synopsis of the Endocytobiological Theory of Cellular Evolution [3]. This is then related to a mechanistic model using the classes of self-replicating systems derived by NASA [4, 5]. This blend of biological and mechanistic systems is further illuminated through the introduction of a numerical model: the stepwise development of number systems.

3. Biological self-evolution principles

Endocyto-biological theory states that some co-evolved variants of cells absorb others and form symbiotic relationships building new levels of modularity. The emergence of complexity via divergent and convergent phases (fig 1) and the development of new levels of modularity can be traced from atomic structure [3].

Each new level of enclosing modularity is produced by a cycle of divergence (variants of the current top module are generated) then there is a transition (co-evolved variants collaborate to produce new functionality) to convergence (new module with new functionality replaces variants not part of the new order). Divergence produces the materials of a supportive environment. Transition commences with the discovery of new functionality assembled from the supportive environment and continues as the new module integrates the generative functionality of the variants of which it is composed and learns to become independent of the supportive environment. Convergence occurs as the new module independent of its nursery environment progresses to dominate the ecology, and after becoming the new top module it will both alter and adapt to its environment and in so doing diversify and hence recommence divergence.

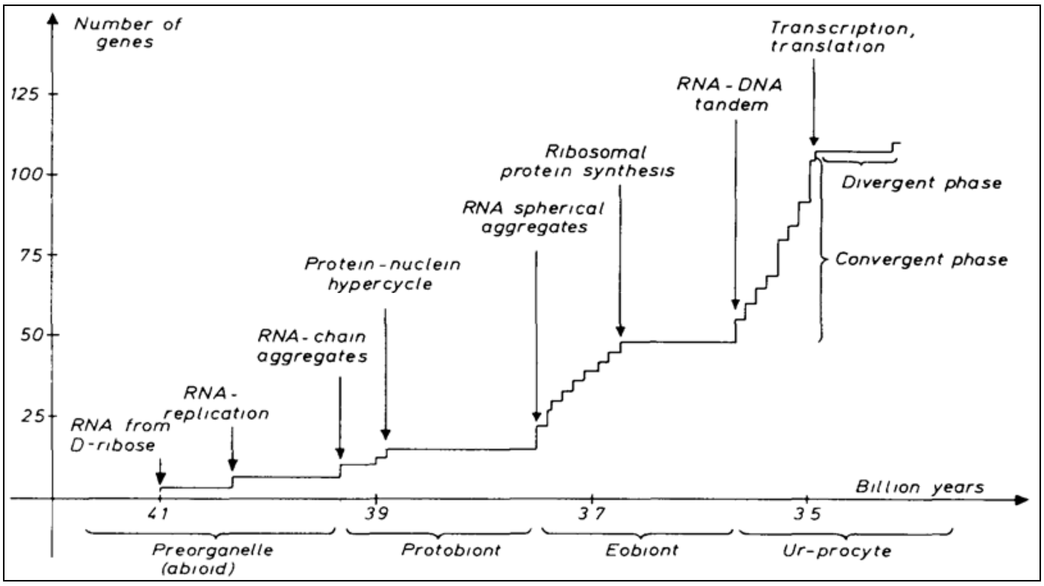

Figure 1. – Gene increase in precytes [3]

Schwemmler [3] introduces 2 principles – modularity and phase (convergent, divergent) that describe the mechanism of evolution. We can add to these the following observations –

Structure and generative grammar:

- Modularity is here embedding producing systems from nested subsystems, with nested containment and environment levels.
- The alternative compositions of component sub-systems form a generative grammar

Internal evolution:

- Containment (spatial closure), even when intermittent and partial, creates feedback. Negative feedback produces control and positive feedback produces structure in these non-equilibrium dissipative systems.
- At the base of this behaviour is a common milieu in which input, output, and control signals are coupled; a system describable via evolution equations

Co-evolution:

- Across each containment membrane - materials, energy and information selectively pass in each direction, creating transformation cycles, and cycles of cycles
- Cycles of transformations, in which subcomponents of all intra and inter compartment /cell /environment transformations balance out (complement), evolve over time

External evolution:

- These generative systems produce variants adapted to various uses (divergence) until functional complements form a new functional closure (convergence) and new module
- This new module with superior performance replaces variants not part of its structure and then starts to produce variants of its own structure (divergence again one level up)

We postulate that the internal, external and co-evolution are coupled and that the extension of development (complexity) outward can be viewed as a single process. Further, we postulate that this process can be integrated and understood as cycles of symmetry transformations. Three forms of symmetry are particularly evident:

- complement (geometric and property antisymmetry)
- similarity (increases as accuracy of catalysis increases)
- functional closure (cyclic processes) produces/extends generative systems

A further form of symmetry, enantiomorphism (handedness) has been associated with biology and life forms since Pasteur observed that crystals of biological material always rotate light in one direction while geological and man made crystals always occur in both left and right hand versions [6]. This asymmetry of biological materials has been confirmed again and again, leading to the widely held view that asymmetry is a fundamental characteristic of biology.

Before discussing mechanical and mathematical developmental theories and then returning to discuss symmetry in more detail, we outline structure and assembly in cells and organisms, in particular Bateson’s Rule and the mechanism of enantiomorphism in organisms.

4. Structure and assembly in nature

There is no reason to assume that evolution is completely reiterated during gestation (biological self-assembly), so that the biological maxim ‘Ontogeny reiterates Phylogeny’ need not be strictly correct. While evolution includes exploratory, discovery and analysis processes during each divergence and transition phase, biological development can proceed by directly expressing the information previously encoded during the transition and convergence phases.

4.1 Structure and assembly in cells

Spatially localised forces (eg electric charges and local fields) on particles with freedom of movement (in solution) form localized bonds enabling self-assembly. However the structures are haphazard unless the particles are uniform and have space-filling symmetry (or are attracted to a symmetric surface or structure) as only then will regular structures with predictable properties, and symmetries, be formed [7].

The quaternary structure of protein is assembled from tertiary (folded) forms of secondary (coiled) forms of primary (sequenced) protein. Complexes of quaternary structure are often composed of one or two components formed into sheets, tubes, toroids and other geometric configurations.

The next level of structure above protein complexes is the organelles which are the functional components from which cells are formed. Quaternary protein, complexes and organelles are analogous to materials, parts and sub-systems in engineering. Sometimes the design and construction of organelles bears a striking similarity to engineering devices.

The similarity between the functionality of cell organelles and the functionality of organs in organisms is noteworthy, and both appear to be related to the conserved physical quantities. As first proved by Emmy Noether [8] and since expanded by Chavchanidze [9], conserved quantities are associated with symmetries.

4.2 Organic Development - Bateson's Rule

When colonies of cells form organisms they also develop structure –

- internal structure associated with functional organs,
- external structure associated with mobility,
- boundary structure associated with the input and output of -
  - information via sensors (touch, vision, hearing, smell, taste) & effectors (vocal cords)
  - energy via heat exchange (ie hair, sweat) and
  - matter via the gut (nose for lungs, mouth for stomach etc).

We are not concerned with the purposeful/functional aspects of the architecture of organisms, only the self-assembly / development (ontogeny) and will address this by considering the mechanisms of formation of external structure (limbs).

How is it that cells in multiplying do not continue to cluster as a sphere (the classic spherical cow)? This question can be answered informally by considering an unfertilised frog's egg with animal and vegetal poles but no difference in its equatorial radii. Such an egg is able to develop into an embryo with bilateral symmetry because the point of entry of the spermatozoon marks one meridian as different from all others [10].

Bateson [11] and later studies, for example [10, 12], explain the mechanisms of formation of asymmetric external structure.

William Bateson believed evolution could not proceed solely on the basis of selection as described by Darwin but was constrained by physical laws and internal order. He coined the word 'genetics' to describe his studies of the internal constraints on organisms and studied genetic errors in order to discover this underlying order. He classified genetic errors and discovered "Bateson's Rule" which "*asserts in its simplest form that when an asymmetrical lateral appendage (eg. a right hand) is reduplicated, the resulting reduplicated limb will be bilaterally symmetrical, consisting of two parts each a mirror image of the other and so placed that a plane of symmetry could be imagined between them*" [12].

Subsequent studies have elucidated the mechanism – each loss of symmetry from the infinite symmetry of the sphere is selected by some external information (like the fertilization point). Established cells, sensing their surroundings, act according to the strength and sequence of the chemicals at their location in gradient fields (fig 2). This process is reiterated via the formation of local (nested) fields with their own sequence of signals. Cells then develop according to information in both the preceding (host) parent field and the local field and one response may be to create yet another new sub-field.

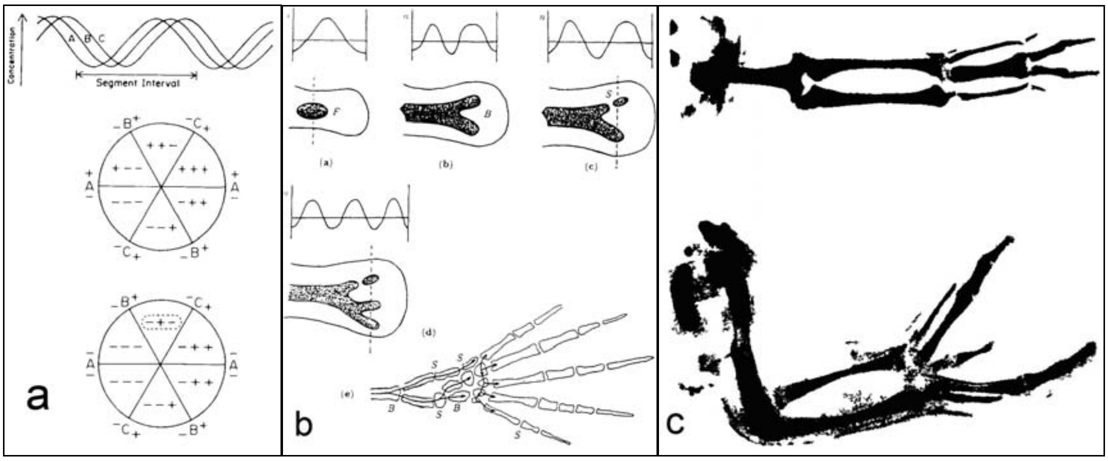

Figure 2. Sequences of cyclic chemical signals in gradient fields in developing limb buds [10]

This system architecture of nested fields or domains, with symmetry determined by parent and child domains, and the symmetry in a field depending on its ‘lineage’, explains how left and right appendages (enantiomorphs) develop. The regularity of developmental genetic errors observed by Bateson has also been partially explained (ie. Fig 2C: when the symmetry information of the parent field is missing, due to the late development of the supplementary limb, the limb reverts to bilateral symmetry). However, the details of these ‘lineages’ of symmetries in nested developmental gradient fields are yet to be completely and formally explained.

5. Mechanical SEB

5.1 Mechanical self-organization

Ball bearings in a box will self-organize under the influence of gravity into regular, symmetric arrangements due to the symmetry of the ball bearings. This is self-organization, however, the ball bearings are not joined and we do not consider this to be an example of self-assembly. The force at work (gravity) is a distributed field and the box provides containment.

5.2 Mechanical self-assembly

Magnetic components will self-assemble [7] if brought close enough to each other (ie. if first self-organized). Random shapes will produce a tangled assembly, however if the components have the same 2D and/or 3D symmetries and are space-filling, regular tiled surfaces and/or structures will be produced. Crystallography catalogues the considerable variety of structures that can be made from simple symmetric objects in regular arrangements. These structures have very high permutation symmetry between components and the bonding between components is uniform.

Components with interfaces (bonding restricted by constraints) will assemble into structures with less permutation symmetry between components but greater structural variety and hence encode more information. An interesting example is the ‘mechanical molecules’ introduced by Lionel and Roger Penrose [13]. These plywood and spring linkage units with various interfaces require external random motion and physical guidance (ie self-organisation) to ‘self-assemble’. The mechanical molecules not only assembled into particular sequences but were designed so that an initial pattern would be repeated by extending the initial chain. When a full duplicate of the original pattern was assembled the duplicate was disconnected thus replicating the original arrangement.

5.3 Logical self-replication

In 1948, John von Neumann initiated the study of “self-producing automata” (later known as “self replicating automata”) [14]. Von Neumann intended to study the logical requirements for a system to have the lifelike characteristic behaviour of being able to produce a duplicate of itself. From the outset, von Neumann only considered self-assembly and self-reproduction under programmed instruction.

Von Neumann was impressed by Turing’s proof of Universal Computation (in essence, all programs can be reduced to a composition of basic operations and, hence, any computer can be programmed to simulate any other computer), and reasoned that a system capable of ‘Universal Construction’ was, since it could build anything, capable of Self-Replication. However, the logical ‘construction’ of von Neumann is not Universal Construction in the sense of a system able to assemble any arrangement possible from a given set of components.

He soon abandoned considering assembly of physical components as too complex in favour of a simple mathematical model – an array of simple computers (Turing machines) operating in synchrony. Each computer examined its last output and that of its immediate neighbours to determine its next output. The output of each cell can be one of a small number of states and is displayed as one cell in an array. The states can represent circuit elements (so that a simple computer can be constructed by ‘drawing its circuit’), or represent program instructions that are interpreted by this simple ‘constructed’ computer, itself embedded (simulated) in the Cellular Automata, that is frequently simulated on a conventional single processor computer.

Some ‘special’ instructions of the ‘constructed’ computer direct cells at the ends of these growing circuits to change to new states, thus extending the circuit (analogous to signals to stem cells telling them to become nerve cells). The circuit grows to create a duplicate of itself and its program instructions. Thus von Neumann demonstrated state configurations in cellular automata capable of replicating. He did not however progress to the stage of describing the architectural design of mechanisms and machinery able to self-replicate.

5.4 Mechanical self-replication

In 1980 a NASA study investigated mechanical self-replication based on the ideas of von Neumann [4, 5]. The associated reports propose “robots assembling robots” exactly the same as themselves as a mechanical implementation of self-replication through self-assembly. This was classified as ‘self-replication in the environment of pre-made components’ (fig 3).

The study recognized that this is a highly specialized environment and defined two further classes of self-replication; ‘self-replication from materials’, which requires the self-replicating system (SRS) first duplicate all its own parts from feedstock then self-assemble, and ‘self-replication from raw materials’ requiring the additional capabilities of extracting and refining minerals (fig 4). The three classes of SRS each require energy and control systems in addition to assembly systems.

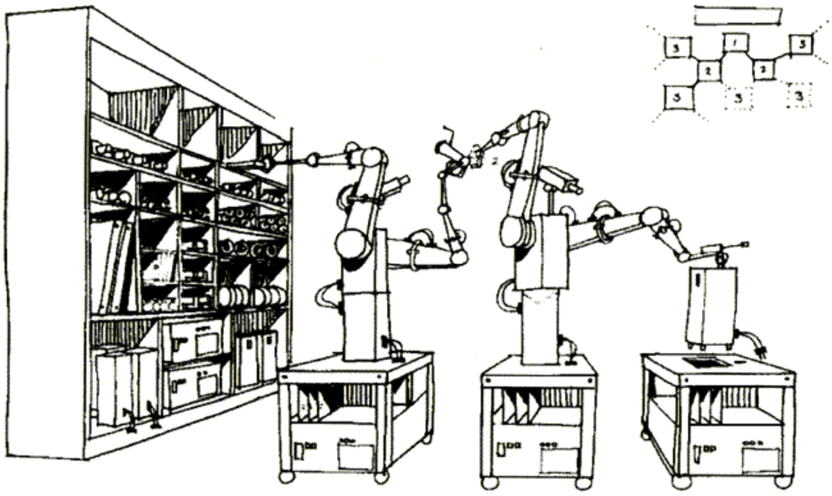

Figure 3. NASA model of mechanical self-replication as self-assembly in an ‘environment of parts’ [4, 5]

The following strategy to develop the technology was defined by NASA (our summary):

1. Design and construct a robot which, when supplied only with
   - main assemblies, can assemble itself, then when supplied only with
   - subassemblies, can assemble itself, then when supplied only with
   - sub-subassemblies, can assemble itself, etc.

   This process is iterated until it produces a SRS which when supplied only with
   - elementary parts, can duplicate itself (SRS1).
2. Design construction robots and a factory building, then from an initial set of parts –
   - use the self-assembling robot, SRS1, to assemble the construction robots,
   - use the construction robots to construct the factory
3. Design ‘SRS1 parts’ production machinery including parts production robots, then from an initial set of parts –
   - use the SRS1 robot to assemble the production machines & production robots.

   This produces a SRS which when supplied only with
   - feedstock, can duplicate itself (SRS2).
4. Design ‘general purpose parts’ factory(s), production machinery & production robots, then from initial feedstock and initial parts –
   - use the construction robots to extend factory or construct new factory(s)
   - use SRS2 to manufacture and assemble general purpose production facilities.
5. Design mining plant and materials processing equipment, factory & robots, then –
   - use the construction robots to extend factory or construct new factory(s)
   - use the general purpose parts production facilities to produce and assemble mining plant, materials processing equipment and materials handling robots.

   This produces a SRS which when supplied only with
   - raw materials, can duplicate itself (SRS3).

The following paragraphs outline the additional information in the NASA reports:

SRS1 is a computer connected to one or more manipulators. Under control of the computer, the manipulator(s) will assemble another computer (with duplicate software), and another set of manipulator(s) from well-defined assemblies. In further iterations, the assemblies would be assembled from sub-assemblies, then from sub-subassemblies (eg integrated circuits, resistors, motors, bearings, shafts, and gears). This iterative development stage will proceed for quite some time as the techniques for assembling each level of assembly from its subassemblies are developed and implemented sequentially.

The development of construction robots and factories in stage 2 was mentioned but not further described except by graphics of stationary and mobile ‘Universal Constructors’ (fig 5).

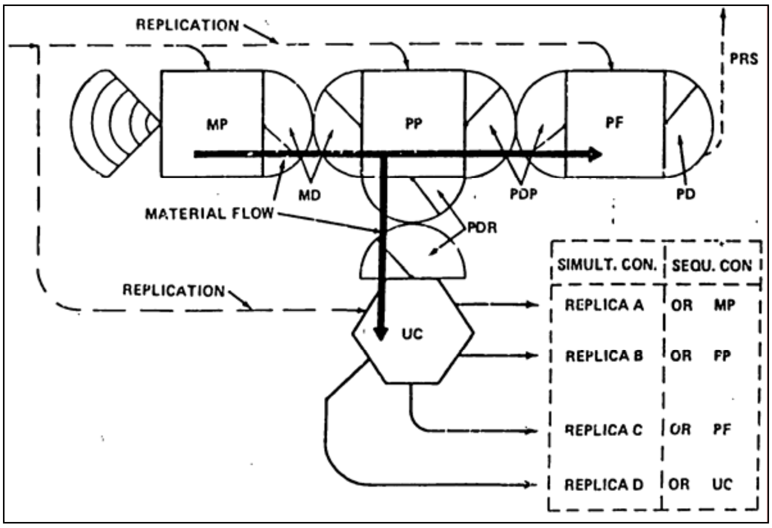

Figure 4. NASA Self-replicating factory schema [4, 5]

Similarly stage 3, development of an autonomous self-replicating manufacturing facility, was not described other than to state “*For example the manipulators could assemble a printed circuit board manufacturing machine or a gear manufacturing machine. The problem of closure becomes a major issue at this point*”. Closure being defined as, “*the entire factory and all of its machines are broken down into their component parts. If the factory cannot fabricate every one of these items, then parts closure does not exist and the system is not fully self-replicating*”. Again an iterative approach is proposed to solve this design problem.

Stages 4 - 5 go beyond self-replicating manufacturing to a system able to “*live off the land*”. Again no designs or design methods other than “*an iterative approach*” are described. Note that energy systems also have to be provided or designed and constructed for each SRS.

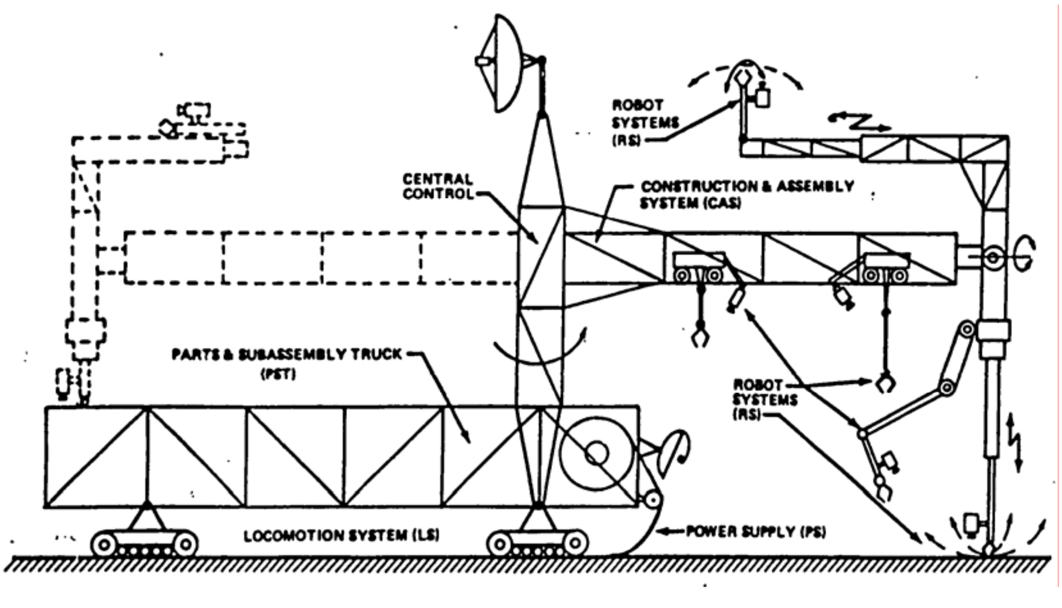

Figure 5. NASA Concept of Mobile Universal Construction [4, 5]

5.5 Comparison with natural development

SRS3 is a development of SRS2 that retains and embeds SRS2. SRS2 is a development of SRS1 which retains and embeds SRS1. This architecture can be viewed as a developmental system when looking forward from SRS1 to SRS3, and/or as a nested embedding system when viewed from SRS3 towards SRS1. Similarly the postcytes (compound cells) are a development of and retain and embed the procytes (simple cells) as organelles, and the procytes are a development of and retain and embed the precyte (RNA/DNA and membrane).

We observe that stage 1 convergences to SRS1, stage 2 divergences, stage 3 convergences to SRS2, stage 4 divergences, and stage 5 convergences to SRS3. This observation, of evolution through alternating divergence and convergence resulting in a series of new ‘modules’ embedding earlier modules, provides evidence of the correctness of Schwemmler’s analysis.

Natural evolution in the longer term proceeds from assembly of elemental components to the assembly of complex organisms (particles, atoms, molecules, cells), however with each new level of modularity the evolutionary process temporarily proceeds in the opposite direction, from the assembly of available complex components to the synthesis of their subcomponents. NASA’s method proceeds in a similar manner from a single self-assembling robot to a mechanical system able to ‘live off the land’ while temporarily in its development cycle proceeding in the opposite direction (eg robot self-assembly from assemblies, sub-assemblies, …, basic parts).

The supportive environment (design office, Computer Aided Design (CAD), machine tools) and development agency (engineers) are analogous to the supportive environment of early earth (the result of cosmic and chemical evolution) where the agency was the laws of physics, in particular the behaviour of non-equilibrium dissipative systems.

SRS1 resulted from iterated development during which the initial self-replicating robot was developed by the external agency from final assembly of major components to the stage of self-assemble of all of its components from elementary parts. Similarly the precytes started by random assembly of lipospheres and clumps of random bio-molecules and developed by physical laws to the eobiont which could assemble all its components and had machinery for programmable control and feedback learning thus no longer solely dependent on physical law.

SRS2 is analogous to the development of a cell membrane. However in cellular evolution this capability is developed concurrently with self-assembly and programmability and is intimately integrated with precyte development. The cell membrane acts as both a template for structures and as a catalysis site facilitating early stage process closure. This suggest NASA’s concept plan could be improved in this regard and that we should view stages 1 and 2 as two aspects of one development cycle [15].

5.6 Completing NASA’s outline

The development sequence proposed by NASA (section 5.4) does not integrate construction of structures with the production of parts and self-assembly. The proposed development sequence, while an outstanding achievement initiating a new field of study, is incomplete:

Firstly, while SRS1 self-assembles and assembles SRS2 which then assembles SRS3, there is no guarantee that any of these stages can be achieved; only a subdivision of the design for self-assembly into assembly levels with a search for a solution on each level by iterative redesign.

Secondly, the strong linkage of design requirements from SRS1 to SRS2 to SRS3 is not captured and processed in a systematic manner. Initial choices will either constrain later development or multi-stage redesign iteration will be required.

Thirdly, the design process compounds in difficulty as, for instance, each factory must be designed to suit its function and environment while its assembly is constrained by the design of the construction robots which are in turn constrained by the capability of the initial SRS1.

We now consider some additional requirements for a more complete mechanical development architecture:

An integrated approach to design relating constraints across multiple levels in an embedded architecture, including parts closure and process closure, is required, for which we propose to develop an integrated, intelligent design environment. This design environment should also integrate product and process selection and design, and integrate the design process from conceptual design through functional design, layout design, detail design, design for manufacture, manufacturing, deployment, operation, decommissioning and recycling, that is across the product life-cycle. Associated with each life cycle phase are sets of constraints which impact on other life-cycle phases.

Virtual prototypes generated via integrated constraint solving from a knowledgebase should be automatically, methodically and comprehensively tested and analysed – freeing designers from mundane tasks to attend to high-level integrated simulation based design - followed by direct manufacturing and progressive hardware-in-the loop simulation-stimulation testing.

While such comprehensive simulation based approaches to engineering design currently exist they are not integrated. Both the US Department of Defence ‘Simulation Based Acquisition’ initiative [16] and NASA’s ‘Integrated Synthesis Environment’ initiative [17] each aspired to a lesser degree of integrated design than that required for self-replicating systems, however neither initiative had a sound formal basis and both are inactive. In 2002 Hollier [18] described the software architectural requirements and a development strategy for such an integrated design environment.

These design environments integrate constraints throughout the life-cycle encompassing design, manufacture, operation, testing, training, deployment, operation, maintenance, repair and disposal. By extending this design approach, relating constraints across multiple levels in an embedded architecture, including parts closure and process closure, an environment able to support self-replicating systems design is defined. Both life cycle and systemic constraints can then become heritable ‘know how’ stored in the design environments knowledgebase.

5.7 Mechanical self-development

There is a clear distinction between organic development and current mechanical production. Nature is constrained in what it can produce due to the architecture of its production method, while the design of machinery is not so severely constrained by current production methods. Consequently, for example, aircraft and birds operate in different performance envelopes.

However, the mechanical development process is not so different from nature if we trace the history of machine tool development, which has the potential to be distilled and automated. SRS not only requires a design methodology for self-replicating systems (as proposed in section 5.6) but also a technically precise methodology of mechanical self-development. That is, there are two bodies of knowledge required, not only the design of the final organism or product but knowledge of how to construct it. This second body of knowledge is deeper than ‘design for manufacture’, assembly plans, or even knowledge of how the production machinery operates. We must know how to manufacture the production machinery, the machinery on which the production machinery was manufactured, … etcetera, and know how to tame this infinite regression.

This infinite regression of backwards analysis is tamed by an initial duplication. In nature self-replication follows fertilization (new extraction of knowledge from the genome) and commences with duplication of RNA/DNA. However, thereafter a foetus is self-developing, expressing stored construction knowledge acquired by evolution, progressing from a simple to a complex organism; while its mother provides an initial supportive environment supplying processed materials as feedstock, and energy. Similarly in NASA's strategy [4, 5] the self-replication sequence begins with duplication of the core system followed by a sequence of systems each building a more elaborate successor. This later sequence is self-replication and not mere duplication. We now need a formalism to encode this self-development sequence.

Hollier [19] combined the essential history of machine tool development with a new formal methodology to describe mechanical self-development in an invited submission to NASA. Extensible generative grammars, with closure, provide a formalism for describing multi-stage development with each stage an outcome of the preceding stages. Their application to the design of a system capable of mechanical self-development was defined by Hollier as:

The sequence – tool maker] machine tool > factory machine > product machine [useful function – we refer to as a generative sequence and the system with this behaviour as a generative system. A generative sequence is a path through a meta-schema hierarchy, at each step a choice or selection from a range of possibilities is made – eg the machine tool is directed to make one of the many possible production machines.

Self-replication is possible when a generative system is automated and has reflexive closure. The requirements of the first step are within the capabilities of the last step – the product machine is the initial machine tool. (Closure indicated by "product machine > [").

The self-replicating behaviour of a system depends on the selection of generative sequence. Sequences that do not close are terminal (eg flexible manufacturing). Sequences that close by reproducing the original machine tool result in population growth by invariant breeding. Sequences which close by changing the original schema result in mutation – if the schema is reduced regression occurs, if the schema is enlarged a progressive increase in capability (ie evolution) results.

Systems are only self-replicating in given environments. The environmental inputs and outputs are shown above as "tool maker]" and "[useful function". The phrase tool maker should be understood as implying both an information component and the necessary materials and energy.

Before finalising our definition we remark that all such systems are dual as observed by John von Neumann and exemplified by DNA & cell, computer software & hardware, and syntax & semantics of text & reader.

The 'hardware' aspect of the dual system is a combination of self-assembling modular linkage robotics [15,19] and self-replicating machine-shop tools. An initial production sequence on existing conventional computer-aided manufacturing (CAM) machines creates the initial self-replicating system (SRS) hardware. Then the first self-replicating sequence occurs when instructions for self-replication, produced from a self-replicating design (developed in the design environment) are interpreted by the initial SRS hardware.

The design environment based on an 'extensible generative grammars with closure' hierarchy, is able to simulate and evolve the hardware and software design via virtual prototyping and to generate developmental (multi-stage production and construction) sequences.

The distinction between mechanical evolution and mechanical development cycles is even greater than in biology, where as noted above ‘Ontogeny reiterates Phylogeny’ is not correct in detail.

5.8 Mechanical self-evolution

The four basic life form behaviours are replication, development, evolution and self-repair[5].

Self–replication is a result of closure of a construction sequence resulting from mechanical self-development. Such construction sequences may be multi-stage (multiple closure) and are initially produced by a generative sequence design environment.

Self-development of a mechanical self-replicating system involves not only multi-stage assembly but must include programmable control. It must include facilities of a nervous system, unlike elementary biological organisms. An organism with a nervous system can have programmable behaviour beyond self-replication and self-development in the sense of growth. Operating systems of this kind are well known and often agent based in the field of robotics.

Self-evolution requires the ability to change behaviour and form and the capacity to learn. An evolving knowledgebase (genome) can be represented as a generative grammar encapsulating a body of knowledge about possible assemblies; configurations including all the generative sequences that can be produced by a developmental strategy. Such a developmental generative grammar knowledgebase can be extended with learning algorithms. An organism that includes self-development generative grammars, and learning algorithms, not just generative sequences, is capable of self-directed evolution.

Self-repair, if interpreted loosely as in self-replication being self-duplication, can be effected by another identical robot. This requires a stock of parts and that reversible joining be used during assembly so that assembly can be reversed (disassembly) until the faulty component or sub-assembly can be replaced and the system reassembled and tested.

However, as discussed in section 5.7, self-replication is more correctly a multi-stage production & construction sequence. An SRS that embeds its generating sequence may be able to self-disassemble and regenerate. This would require going into states of regression that would impact on the ability of the organism to remain operational (survive). Strategies for partial and local regeneration (healing) are therefore of value to an organism, but will not be discussed in this paper.

6. Development of number systems

Pure mathematicians investigating the foundations of mathematics define number systems via a constructive process as shown in table 1. This evolutionary process is similar to natural and mechanical evolutionary processes, requiring the creation of an initial system provided by an external agent, then divergent and convergent phases creating new operators (organisms) and enriched number fields (environments).

The perspective applied in the evolution of successive number systems varies from biological and mechanical evolution in that the focus of attention and emphasis is on the development of the number fields rather than the development of the operators within the fields. The analogy may be extended from elementary operators to functions and classes of polynomials being considered as organisms.

Phase	*Environment*	*Organism*	*Variants*	*New Module*	
	Field	**Operator**	**Product**	**Extension**	**Sym**
Creation	Void (nul)	Distinction	Entity	Unit (one)	1
Diverge	Unit (one)	Increment	Positive Nos.	Natural numbers	N
Converge	Natural numbers	Decrement	Negative Nos. - Zero	Integer numbers	Z 0
Diverge	Integer numbers	Addition	Sums	Linear numbers	
Converge	Linear numbers	Subtraction	Differences - Primes	Inverse numbers	N^{-1} P
Diverge	Inverse numbers	Multiplication	Products	Bilinear numbers	
Converge	Bilinear numbers	Division	Quotients - Rationals	Real numbers	R Q
Diverge	Real (irrationals)	Exponentiation	Exponents	Powers Series	
Converge	Powers	Inverse exp.	Inv. Exp. - Irrationals	Convergent Series	√

Table 1 - Development of number systems and operators

A further observation of relevance to our investigation is the cycles in the calculation (action of the operators) and representation of numbers in the more developed number fields such as the real numbers. Real numbers have a decimal representation in which fractions produced by division have either finite representations (terminating calculations) or infinite representations (cyclic repeating calculations) that consist of a repeating series of digits. Real numbers also include the irrationals which have infinite representations in which the digits never repeat (a non terminating calculation that is infinitely variable and never closes to become cyclic).

These observations can be related to the cyclic generative systems discussed in section 5.7. Furthermore, we note that automata are formally modelled by power series which suggests an analogy between convergent series and self-reference in automata.

While counting, addition, subtraction, multiplication and division are different operators they are composed of common elements. Divergence (increment, add, multiply, …) is a result of successive, compounding, regular grouping and each inverse operation (decrement, subtract, divide, …) leads to convergence as a field extension (integers, inverses, real numbers, …) introduces a new functionality (zero, primes, rationals, …).

The common elements are:

‘repetition’ of an operator (eg increment) to generate new products (eg positive numbers);
‘inversion’ of the operator (eg decrement) to generate new products (eg negative numbers;
‘extension’ of the number field with a new functionality (eg, zero); and
‘embedding’ of the evolutionary sequence (eg powers (products (sums (unit)))).

7. Symmetry and meta-operators

In section 3.6, we postulated connection between internal, external and co-evolving processes and structures and across levels of embedded modularity using connected cycles of symmetry transformations as biological self-evolution principles. We have now identified repeating operations across cycles of evolution: repetitive generation, inverse generation, functional extension and embedding.

These repeating classes of operator are meta-operators associated with symmetries: repetition of an operator is a process isomorphism creating products with internal symmetries, inversion is mirror symmetry, extension is a process closure (self-mapping) and embedding is repeated involution.

Exotic forms of symmetry will be required to prove or disprove our conjecture and that would take us beyond the scope of this paper so we limit ourselves to the following observations: that complements, inversions and nesting operators are all involutions, and that involutions are essential to study these endo-exo chains of symmetry transformations in detail.

8. Complexity

Endo-exo symmetry transformation analysis of developmental and evolutionary processes and emergent properties could provide a common basis for analysing, comparing and unifying the different approaches to complexity.

Logical complexity theory developed in the 1930's from Turing's proof of the undecidability of the halting problem (ie. whether a program will close to a cycle and loop or terminate) and Godel's incompleteness theorems (ie. the inability to prove all true theorems in a formal logic). While von Neumann defined complexity and emergence of new properties as follows:

"*There is a minimum number of parts below which complication is degenerative, in the sense that if one automaton makes another the second is less complex than the first, but above which it is possible for an automaton to construct other automata of equal or higher complexity. Where this number lies depends how you define the parts*."

Mathematical complexity with divergent, non-linear and evolutionary equations and 'order from chaos' developed after the work of Turing and von Neumann, as did computational complexity with genetic programming, neural net theory and co-evolving agents, and physical complexity with self-organizing and self-assembling phenomena in phase transitions.

The original logical approach to complexity high-lighted the limitations of the simple non-referential systems that resulted from banishing paradox by avoiding it with class theory and type theory. Computational complexity provided means for exploring emergence but little in the way of explanation. Finally mathematical complexity provided explanation and theoretical understanding via 'order from chaos', symmetry-breaking and second order thermodynamics.

The phenomenon of life show us that our understanding is as yet still elementary and new techniques are required to analyse the emergent sequences that characterize Self-Evolving Behaviour. The analysis of SEB by studying changes in symmetry occasioned by the cyclic application of these meta-operators is proposed as a method of both integrating and extending our understanding of complexity.

9. Conclusion

Self-organization, self-assembly, self-reproduction and development/ontogeny - a spectrum of Self-Evolving Behaviour (SEB) of increasing sophistication - have been related through the study of biological and mechanistic systems. SEB was related to number system development by reduction to generation, inversion, extension and embedding meta-operators.

Two new techniques – extensible generative grammars (EGGs) with closure, and endo-exo symmetry transformation analysis (endo-symmetry) have been described. These techniques can be applied to the analysis of SEB and complexity theory. These techniques could also form the basis of advanced engineering design and analysis environments [15].

An early demonstration of this new methods value would be the development of mechanical automata (ie mobile robotics) by applying nested symmetry coding to develop limb designs.

Maturity and success in applying these Self-Evolving Behaviour based design techniques will be apparent when SEB based design automation can derive self-replicating systems.

References

1 Martin, G. (1982) *Transformation Geometry: An Introduction to Symmetry*, Springer Verlag

2 Kappraff, J. (2001) *Connections: the geometric bridge between art and science*, 2nd ed., World Scientific

3 Schwemmler, W. (1984) *Reconstruction of Cell Evolution: A Periodic System*, CRC Press Florida

4 NASA TM 78304 (1980) *Self-Replicating Systems- A Systems Engineering Approach*, July 1980

5 Frietas, R., Gilbreath, W (1982), *Advanced Automation for Space Missions*, Proc. NASA/ASEE Summer Study 1980, NASA Conference Publication 2255

6 Goldanski, V. (2000) *The role of spontaneous breaking of mirror symmetry in chemical and pre-biotic evolution*, in Symmetry 2000 …..

7 Whiteside, G., Grzybowski, B. (2002) *Self-Assembly at All Scales*, Science, vol 295, pp. 2418-2421, Mar 2002

8 Byers, N (1998) *E. Noether's Discovery of the Deep Connection Between Symmetries and Conservation Laws*, on internet: arXiv:physics/9807044 v2 23 Sep 1998

9 Chavchanidze, G. (2002) *Remarks on Conservation Laws associated with non-noether symmetries*, available on internet: arXiv:math-ph/0207021 v1, 18 Jul 2002

10 Webster, G., Goodwin, B., (1996) *Form and Transformation: Generative and Relational Principles in Biology*, Cambridge University Press

11 Bateson, W. (1894) *Materials for the Study of Variation*, London, McMillan and Co.

12 Bateson, G. (1971) *A Re-examination of Bateson's Rule*, Journal of Genetics 1971, reprinted in Bateson, G (1971) *Steps to an Ecology of Mind*, Ballantyne Books

13 Penrose, L. (1959) *Self-Reproducing Machines*, Scientific American, June 1959

14 von Neumann, J. (1996) *Theory of Self-Replicating Automata* (Hixon Symposium, September 1948), University of Illinois

15 Hollier, W., Burvill, C., Davoren, J. (2005) *Symmetry in Geometric Features and Self-Assembly: Self-evolution II*, 15th International Conference on Engineering Design, these proceedings

16 Bernstein, R. (1998) *A Roadmap for Simulation Based Acquisition*, Report of the Joint Simulation Based Acquisition Taskforce, US DoD Acquisition Council

17 Braun, R. (2001) *NASA's Intelligent Synthesis Environment – Revolutionizing the Agency's Engineering and Science Practice*, Integrated Enterprise Vol 2(1):2-6, Advanced Enterprise Solutions

18 Hollier, W., Beckett, P. (2002) *Simulation-Based Acquisition: Architecture and Implementation Part 1 & Part 2*, SimTect 2002 Conference, SIAA, Melbourne 2002

19 Hollier, W. (1990) *Advances in Evolutionary Self-Replicating Systems Engineering*, Invited Report to NASA

CORRESPONDING AUTHORS:

William E. Hollier
Director, EnGen Institute
PO Box 189, Black Rock, VIC 3193
Australia
Email: will@engen.org

Colin Burvill, Senior Lecturer,
Dept. of Mechanical and Manufacturing Engineering
The University of Melbourne, Parkville, VIC 3010
Australia
Email: colb@unimelb.edu.au