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

SYMMETRY IN GEOMETRIC FEATURES AND SELF-ASSEMBLY: SELF-EVOLUTION-II

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

Abstract

This investigation has correlated the architecture of self-assembling automata and geometric features with localised and specific symmetry. Parametric prototypes incorporating these geometric features can be placed in a library and accessed from a feature based computer aided design (CAD) systems for self-assembling structures.

When such a CAD system is based on a hierarchy of models each defined with formal grammars and linked via formal semantics the model hierarchy can be integrated and closure defined. Constraints between models (across abstraction levels) allow self-assembling systems to be defined and provide a tool to further investigate self-evolving behaviour.

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

1. Introduction

Prior study of Self-Evolving Behaviour (SEB) [1] has identified symmetry as underlying such behaviour and proposed two techniques, 'multi-stage extensible generative grammars (EGGs) with closure' and 'endo-symmetry', to represent and design mechanical developmental systems with SEB.

This paper reports the findings of a study of symmetry properties of modular systems in order to identify a set of symmetric geometric features [2] to be incorporated into a Simulation Based Design (SBD) software tool [3], the purpose of which is to design and simulate self-assembling structures [4] and further investigate self-evolving mechanisms.

2. Method

The investigation is restricted to man-made, or at least potentially man-made, mechanistic systems. In order to analyse self-assembling automata we extend the usual classification used in the theory of machines to a progression of entities commencing with components and structures, then mechanisms and machines, and concluding with mechanical automata and systems. The complex constraints that characterise self-assembling systems of mechanical automata are thus analysed and enumerated progressively. The study considers symmetry and its consequences at each of these levels to develop a catalogue of necessary design elements for classes of self evolving behaviour.

The design elements are then combined with the representation and design methods derived in the prior study and their joint implementation in CAD and SBD software considered.

† EnGen Institute and § The University of Melbourne

3. Components

3.1 Modularity and symmetry

We define modularity as unit structuring, and to analyse unit structuring we consider a brick.

Firstly, every brick is the same as every other brick. This symmetry between units means they are efficient to manufacture, transport and assemble, because the planning, materials sourcing, tooling and fixturing tasks are performed once for all bricks rather than for each item.

Secondly, unit structuring makes a system fault tolerant. Were one wall frame to break during assembly or transport that particular frame would need to be repaired or replaced with considerable inconvenience; whereas a small number of spare bricks fixes the problem at the site of failure. This is component permutation symmetry.

Thirdly, each brick is symmetrical in itself having three orthogonal axes of symmetry. A brick is a space-filling, regular solid (the cube) extended along two axis. The width of a brick is greater than it's height for increased stability, while greater length considerably increases the strength of bonding in brick structures. Significantly these variations from regularity do not effect the space-filling and orthogonal characteristics.

Space-filling components allow stronger structures to be built and regular solids provide the maximum freedom of configuration. Freedom of configuration means the brick can be put to many uses. It has far higher utility than a less symmetrical design such as wall framing where a set of frames can only be assembled to one plan.

Fourthly, the symmetrical, unit structured brick can create an indefinitely large structure. Wall frames aren't extensible unless they contain certain symmetries of shape which determine their connectivity. Thus modularity is unit-structuring with connectivity symmetry.

Modular systems of construction have four characteristics associated with unit structuring and connectivity symmetry - efficiency, fault tolerance, utility and extensibility. In summary, two classes of symmetry – 'symmetry between parts' and 'symmetry of a part' – provide four useful characteristics.

3.2 Modularity and self-organization

Identical components can be interchanged and symmetrical components re-oriented without changing a configuration. The configurations that can be produced with a set of identical components will be called a combination pattern as it is independent of permutation.

Combination patterns can be irregular and asymmetrical or can be arranged to be symmetrical (eg components form a square) and repeating (components form a lattice). A configuration that is symmetrical, repeating and composed entirely of modular components with all connections being identical is such that all components have an identical role in the structure.

The only two dimensional regular symmetrical shapes are the regular polygons and although there are 17 wallpaper groups, the only regular (all angles and sides equal) 2D figures which are space-filling are the equilateral triangle, square and regular hexagon. With appropriately angled magnetic poles for connectors regular polygons will self-assemble in an appropriate closed environment such as the surface of a gently stirred water tank [5].

The Platonic solids are the only three dimensional forms able to be covered with a single type of regular polygon either triangles, squares or pentagons. Only tetrahedrons and cubes are space-filling by themselves [6]. The surface of these forms are closed cells, have finite construction and every component can be assembled regardless of position and orientation.

Archimedean solids are three dimensional surfaces covered with two types of regular polygon and are of interest as a possible component shapes for self-assembling automata. The surfaces of these forms are closed cells and some combinations of these forms are space-filling.

The edges of Platonic solids are regular (all angles and lengths equal) and the edges of Archimedean solids have two lengths and two edges [6]. These edges (mathematical concept) can be realized by components approximating a line (having length considerably greater than cross-section) and end connectors at both ends which we refer to as linkages. Linkages with appropriate multivalent connectors make cell cages.

4. Structures and mechanisms

The means by which components are held in contact are many and varied, however they are of three types - force (eg gravity, magnet, vacuum, electro-magnet), material (nail, adhesive, rivet), or closure (enclosure via contraction of cavity aperture, enclosure by expansion of enclosed object etc).

As our study concerns modular systems of self-assembling automata, we are only concerned with localized forces controllable within the system (in fact programmable components) and reversible operations including reversible joining.

Objects that are attracted towards each other (eg electro-magnets) will contact however they will not form a specific shape by this capability alone. An instance of a specific structure is formed when a set of objects are connected so that specific locations on the objects are held in contact in a specific pattern of connection and orientation. All instances produced by permutation of location and orientation of symmetrical and identical components are considered to be the same structure.

In our current examination of structures and mechanisms we are principally concerned with reversible joining (usually via closure) at sites on rigid components.

All contact observed on a sufficiently magnified scale is surface contact, but if sites of contact are small or linear they are conventionally considered the abstract geometrical entities of points and edges. We will classify reversible joining sites as point, edge or surface connectors (geometric features) on the basis that they have the same symmetries as the abstract entities. Next we consider the ‘symmetries of connectors’ and ‘symmetries between connectors’ at specific surface locations on rigid components.

4.1 Connectors and connector pairs

If a portion of the surface of one object (a geometric feature) can be brought into contact everywhere with a portion of the surface of another object (another geometric feature) then the contacting surfaces (features) are ‘complementary’. These surfaces can be arranged in one, many or an infinite number of orientations when the surface shape has no symmetry, finite symmetries or an infinite number of symmetries.

By considering a square peg fitting a round hole or a square nut fitting a ring spanner it is apparent that surface shapes can be ‘effectively complementary’ and have effective symmetry providing a selection amongst symmetrically arranged possible relative orientations.

When two surfaces in contact allow a finite number of orientations a change from one orientation to another requires disconnection and reconnection. When an infinite number of orientations is possible so is motion. Two surfaces that are effectively symmetrical and have infinite symmetry change the corresponding points in contact of the surfaces in contact when in motion (eg ball and socket joint, wheel and rail). We will refer to surfaces that are held in contact such that motion is allowed as being semi-attached.

The connecting surfaces considered so far have been complementary but not identical (although the components may be identical). Generally, there is a male and a female feature (eg tongue and groove boards). However, a subset of the surface features considered so far are self-complementary, that is, a feature is self-similar so that when rotated it is its own complement. Self-complementary features allow components to have greater symmetry and greater interconnectivity.

The operation of bringing two surfaces into contact is called docking. The operation of holding two surfaces in contact is called locking. Note that self-complementary identical connectors are doubly locked providing additional integrity to a structure.

Semi-attached connectors allow motion when semi-locked (held in contact so that motion is possible). Two types of operation uniquely determine an orientation/location by progression from the semi-locked to the locked state - a change of scale (eg. friction lock due to expansion/contraction of enclosed/enclosing surfaces) or a change of symmetry (eg. rachet closure) of a connector pair.

4.2 Connectivity

We continue to discuss the assembly of configurations of static components but distinguish between configurations composed of connector pairs with finite symmetry and configurations including connector pairs with infinite symmetries. We now define –

<u>A structure</u> is a configuration of components in which no motion is possible. A structure is defined as including all of the finite number of orientations of a single pattern of connectivity of static components.

A given set of static components can be assembled into a finite number of structures. The possible configurations include all combinations of connectivity, permitted by the set of fixed relative locations of connectors on the set of static components, which are not prohibited by connector mating, connector occupancy or component intersection.

For a given set of components with a given set of connectors the maximum connectivity is achieved when all connectors on all components are self-complementary and of one type. If connectors are complementary (male and female) maximum connectivity is achieved when they are of one type of pair.

If the distance separating connectors and/or the orientation of connectors on all components are different and all connectors have finite symmetry the possible arrangements, including closed cycles, are reduced or entirely prohibited leaving branching configurations as the only configurations possible. Thus regular (equal distances and angles) location and orientation of connectors also facilitate maximum connectivity, including configurations with multiply connected components and enhanced structural integrity.

<u>A mechanism</u> is a configuration of components in which motion is possible. A mechanism will have an infinite number of orientations associated with a single pattern of connectivity of static components.

We now consider only motion due to rotational freedom (infinite symmetry) of connection which is effected by an outside agency (eg door, tent). This allows flexible structures to be produced (fig 1). This class of mechanism will be referred to as an <u>R-mechanisms</u>.

4.3 Mechanical integrity

A cycle of three components will fix the relative location of the three points of connection in both structures and R-mechanisms. R-mechanisms still allow any component between two connectors with co-linear axes of infinite symmetry to rotate.

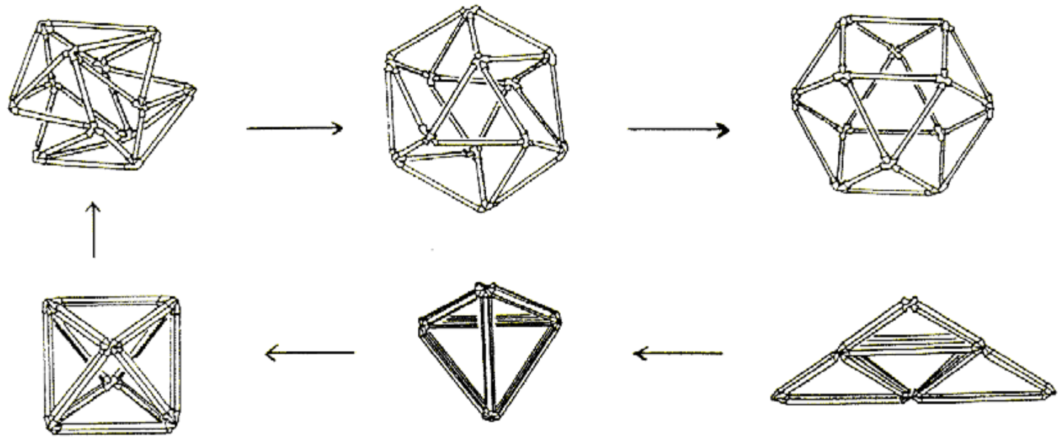

Figure 1 Buckminster Fuller's 'Jitterbug' foldable mechanism [6]

Branching R-mechanisms or R-mechanisms with cycles of four or more connections will readily fold. Branching structures or structures with cycles of four or more connections will have little mechanical integrity as then the orientation of the components depends on the ability of the connections to resist rotation.

The integrity of structures increases with stronger connectors and stronger locking and decreases with larger moments of rotation, however at a cost in strength to weight ratio for a given material.

Integrity is also increased if the component surfaces are in contact and further increased when the contacting surfaces are increased (become complementary). Thus contact can elevate the mechanical integrity of point connection to edge or surface connection and edge connection to surface connection for specific orientations. The more points of connection on an edge (eg hinge) or surface the stronger the structure. Mechanical integrity increases continuously with contact and connection.

Thus surface filling components are structurally advantageous and also allow closed surfaces to be produced, while volume filling components provide the strongest structures for a given material.

However the number of alternative configuration and the number of degrees of freedom (DOF) are least with volume filling components, greater with surface filling components and most with thin linear (linkage) components. As our goal is modular self-assembling structures linkage components are preferable.

4.4 Multivalent linkage and triangulation

We now only consider further mechanisms composed of components which approximate lines which we refer to as 'linkage'.

The 'effective DOF' of individual connectors increase as connectors are moved from surfaces to edges and from edges to vertices, however connectivity decreases as the number of connectors decreases. High connectivity can be achieved if connectors are located in groups at vertices. A grouping of connectors, called a multivalent connector, is most useful when the grouping is symmetrical about the component vertex. Multivalent connectors provide another means, in addition to three cycles, surface or volume filling components and edge connectors, of producing structures with increased mechanical integrity. We now define R-mechanisms –

R-mechanisms are linkage mechanisms in which motion is due only to rotational freedom (infinite symmetry) of connection and effected by an agency outside the system.

Using linkage components with multivalent connectors four interlinked three cycles (eg like a tetrahedron but not necessarily symmetrical) - called a tetralink - will fix the relative locations of the four points of multivalent connection. The mechanical integrity of structures composed of successively interlinked three cycles (eg triangulation) to produce interlinked tetralinks (eg tetramesh) is sound.

Note that a fully triangulated tetramesh requires 4-valency connectors (like hydrocarbons) and that triangulation is restricted to equilateral triangles if the set of components in an R-mechanism is of one type and hence one length.

An R-mechanism tetramesh will form sound structures which can be articulated (where triangulation is incomplete) and can use rotating links as rollers however it still has only limited useful movement. It is now time to consider connectors with translational movement.

T-mechanisms are linkage mechanisms in which motion is due only to translational freedom (infinite symmetry) of connection and effected by an agency outside the system

Cartesian motion is possible within a T-mechanism, where polar motion was possible within an R-mechanism. Triangulation for structural integrity is restricted as rotational (infinite symmetry) connections are not available but discrete symmetry connections are available. Triangulation is not possible if only parallel and orthogonal connections are available. Triangulation is restricted to congruent (constant angle) triangles. Congruence requires the three components change length (translation) at rates proportional to their length so that the ratios of lengths and hence angles remain constant.

In order to have the mechanical integrity of triangulation while having free movement we require both rotation and translational movement (infinite symmetries) in mechanisms and so we now define –

TR-mechanisms is a linkage mechanism in which both rotational and translational connectors alternate (ie vertices and sides connect alternately) and motion is effected by an agency outside the system.

In order to maintain maximal modularity as linkage unit with length translation and connector rotation at both ends is optimal. In order to maintain mechanical integrity by constructing fully triangulated TR-mechanisms both end connectors will be multi-valent with at least 4-valency.

5. Machines and Automata

Machines are defined as a configuration of components in which motion is possible which is, in part at least, effected by the machine itself. Machines when active transform energy and progress through 'cycles of component orientation' (operation) within the fixed configuration.

Elementary machines obtain energy from and move in response to their environment (eg. door closer).

Basic machines are connected to a source of energy which if not mechanical is converted to mechanical energy (a motor) and have on/off (connect/disconnect) and/or regulation energy control.

Active components are defined as a minimal configuration of components in a machine in which energy is transformed and motion created.

Sensors are defined as devices which change state according to the state of their environment, which may include manipulation by an operator. When sensors are mechanical the change of state is a change of orientation of one or more components is response to the mechanical state of the environment. Sensors are active components.

Sensor machines have alternate operation cycles selected by sensors. Sensor control leads to –

Programmable control occurs when the principal operation cycle senses states recorded in a medium (a program) which select and activate specific subsidiary operation cycles.

Automata are machines in which operations are selected, activated and deactivated (possibly concurrently), under program control. Mechanical automata perform mechanical operations.

In this study, as we are investigating self-assembly, we are considering mechanical automata able to change their external form (external mechanical orientation of components) so as to mechanically operate on their environment to effect assembly. Thus we define -

Effectors are defined as devices which change state under program control, where the change of state effects the external environment. When effectors are mechanical the change of state is a change of orientation of one or more components that effect the external environment.

We now consider the assembly of active components, to form mechanical automata with effectors, that is mechanical automata that have programmable, variable shape. We will from this point forward refer to mechanical automata with effectors simply as automata.

A given set of static and active components can be assembled into a finite number of structures, mechanisms, machines and automata. The possible configurations include all combinations of connectivity, permitted by the set of fixed relative locations of connectors on static components and variable relative locations of connectors on active components, which are not prohibited by connector mating, connector occupancy or component intersection.

5.1 Functionality

Self-assembly of useful structures, mechanisms and machines requires that at least some of the components themselves are active, that is transform energy and produce motion. The functionality of mechanisms able to self-assemble under programmable control requires component assemblies with movement under programmable control including connectors able to attach and detach as required (robot grippers). These active components also require on-board or reticulated power, and must be able to communicate and coordinate requiring position and contact sensors

Whereas various building materials are assembled with permanent bonds, robot grippers are designed to attach and detach. Reversible joining is preferable as this produces a–

Reusable system with the life-like characteristics of
Repair (eg self-repair)
Reconfiguration (eg caterpillar metamorphosis to butterfly)
Development (eg growth and ontogeny)

Whereas building materials whether rigid, plastic or elastic are unable to control their own shape robots are designed to have programmable internal degrees of movement. Mechanisms composed of active components able to change their shape (modular robotics) have –

> Movement and hence can be
> Mobile.

To facilitate description, the operation of these mechanisms can be analysed into four modes–

1. Contact: The components must be brought into contact (dock) or near proximity as a prerequisite to other modes.
2. Traverse: Components in contact may traverse other components (eg. via lock, move, unlock, move cycles of limbs) in order to reconfigure a component arrangement.

It is often only a matter of perspective as to whether this is motion within one entity, or the relative motion of entities in which case it is again a matter of perspective as to which is transferor and which is transferee.

3. Connect: When components have transferred to the site at which they are to be located in a defined configuration the component and structure connect (lock) to reconfigure or extend the structure.

The mechanical requirements of these connections are determined by the structural requirements at the particular site, in the configuration whether final or temporary during the processes of construction and operation. Hence the mechanical characteristics of a set of components and of the connections they make is a determining factor of the set of possible configurations.

4. Operate: In a final configuration some components may operate so as to produce movement within the configuration or even mobility of the entire configuration.

5.2 Self-assembling automata

A first solution is a robotic assembly line, however this is not self-assembly unless the assembly line builds a duplicate assembly line, in which case this is hardly a minimal core self-assembling system.

A second solution is a modular construction system (eg scaffolding, bricks) over which a construction robot traverses – a ‘spider web’ solution. “Spider’s web” designs have been built as robots to lay railway track and bricklaying robots traversing walls. The ultimate system of this type is the “Universal Beam Constructor” developed for NASA by Grumman Industries. It folds, stamps and crimps foil from rolls to produce very lightweight beams in much the same way as spouting is now formed on site. The beam constructor can then traverse this beam, rotate and produce a beam at an angle to the original beam. It also joins beams. Spider web systems are good as construction robots but they do not self-assemble as only a static structure is constructed and the heart of the system the robot is not duplicated.

The solution is to use modular automata as the components. Each such unit can traverse its neighbours and lock into place as part of the structure. The structures composed this way are also capable of dynamic movement. Note also that organic self-assembly at the most fundamental level the replication of RNA/DNA operates this way where the components are available (not formed) and all that is replicated is their arrangement..

As shown in the analysis of section 4, the optimal modular automata for self-assembly is the linkage unit with length translation and connector rotation (compliant or active rotation wrist) at both ends, which we refer to as TR-automata. In order to maintain mechanical integrity by constructing fully triangulated TR-automata meshes both end connectors will be multi-valent with at least 4-valency. The connectors are optimal if of one type and self-complementary (but could also be complementary male / female pairs).

This modular linkage robotics can form space frame structures and 'parallel architecture' assembly line robots. These linkage units can also have attached modular panels which connect to adjacent panels. This form of unit structure is capability of forming both space frame and shell structures which are mobile.

6. Design environment architecture

Having analysed and summarized constraints on self-assembly of systems of automata from a symmetry perspective, we now consider how these constraints can be combined with results from our prior study [1] and implemented as a design environment for machines made of self-assembling automata.

6.1 Specification and extensible generative grammars

In evolving a design from an identified need through conceptual design to detailed design, while considering all requirements and constraints throughout a systems life cycle, it is accepted practice to undertake systems analysis to derive a specification which is then refined by iterative design (fig 2).

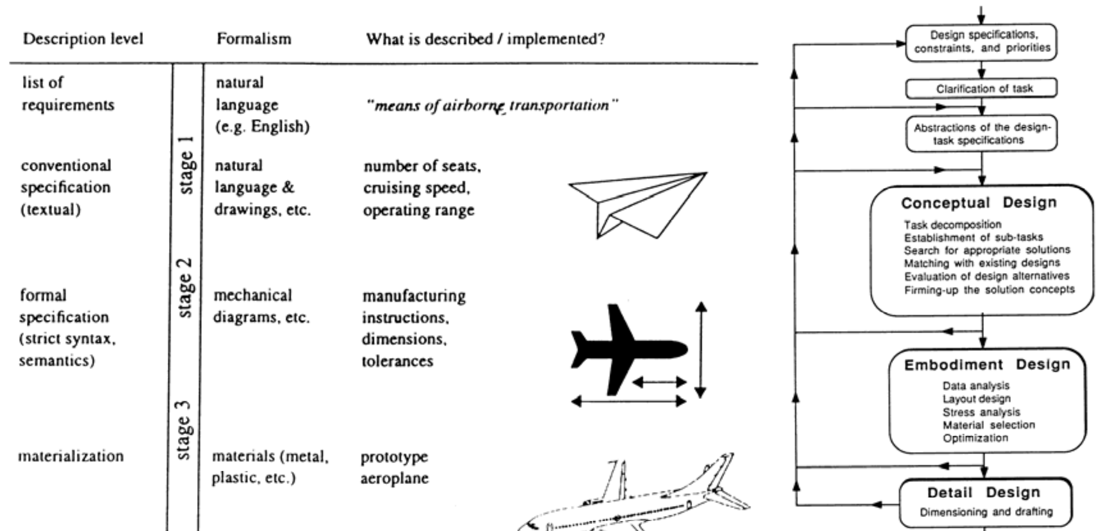

Figure 2 –Conventional View of Specification and Design Flow [7]

The conventional view of the specification and design processes is readily understood but for our purpose an 'information flow' view (fig 3) showing the transformations of information type and representation and the hierarchical, cyclic synthesis & analysis process is preferable.

This hierarchical, cyclic process uses a hierarchy of languages whilst progressing a simple, abstract statement into a complex, concrete statement. Each step in the synthesis process takes one language as an input specification and translates to a more concrete language while at the same time introducing additional constraints relevant to the concrete representation. If these new constraints are incompatible with the previously accumulated constraints, the conflict is analysed and translated into the input language and the previous design process reiterated, progressing up the language and decision hierarchy as required until a modification of specification resolves the conflict.

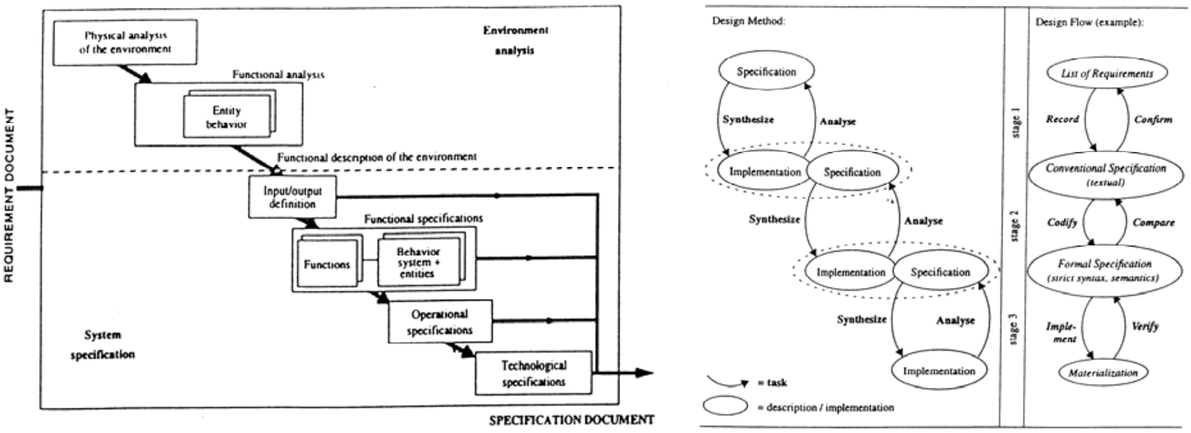

Figure 3 Information Flow View of Design Flow [8, 7]

In prior work [1], hierarchies of extensible generative grammars with closure were found to be a suitable basis for describing mechanical developmental systems with self-evolving behaviour. We have also noted the use of language hierarchies in design environments for software and electronics. Closure can be described using formally defined semantics to link languages.

6.2 Formal Grammar Semantics

Typically, the languages are not precisely specified nor constrained so that the language is extended as required with new definitions. However formal methods of specification and design have been developed for computer science and computer design where the languages used are formally defined using mathematics and logic. Formal languages can be defined semantically as well as syntactically in which cases translation between languages and error checking can be automated. Such translations are not restricted to requirement and specification languages, and may include translations to graphical languages for visualization and simulation (fig 4) and/or command languages of robotics and machinery for production, assembly and testing.

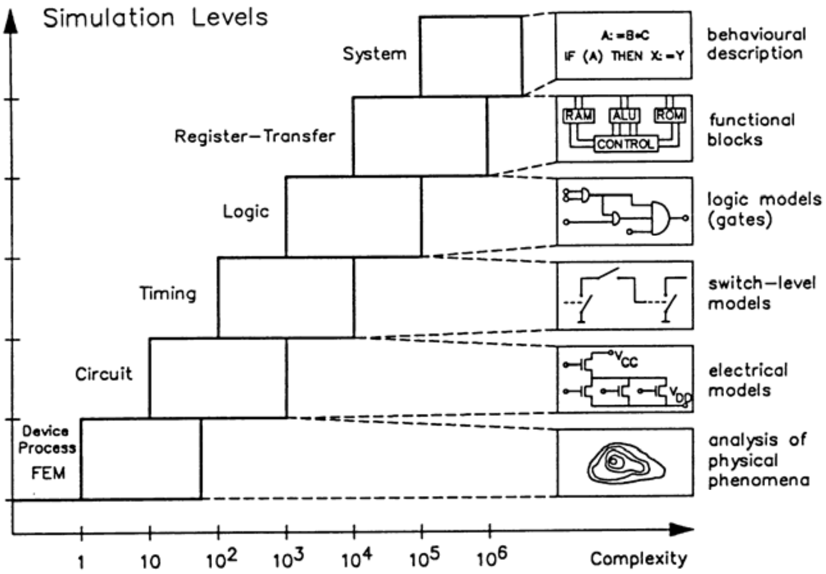

Figure 4 Simulation and Language Hierarchy [9]

Whitney [10] has documented a number of the significant barriers that are to be overcome if formal methods are to be applied successfully in engineering design environments and we now address some of these considerations.

We note that the use of formal grammars in design environments for computer programs and electronics is restricted to the definition and translation of languages describing requirements, specifications and design – that is the end results of each stage of design. Design decisions are made by designers using operations research and formal logic software tools embedded in design environments that amplify their productivity. Hierarchical attributed grammars and associated decision processes are commonly used in software design and translation while for electronics attributed graph grammars are used.

Whereas, in the shape grammar approach to architectural and engineering design, formal grammars have frequently been used to directly generate design alternatives, although augmented graph grammars and decision processes are now beginning to be used [11].

The co-design of software and electronics is currently an active research field and we hope to extend this to include mechanical design for mechatronics using formal methods. The initial goal is to develop a design environment for self-assembling cellular robotic automata. This design environment will use a hierarchy of extensible tree and graph grammars attributed with symmetry information.

6.3 Symmetry based design

The consideration of symmetry is well established practice in engineering design. The degrees of freedom between all possible pairings of components where movement is possible are known and classified into lower (surface contact) and upper (edge and point contact) pairs. Liu [12] has studied the use of symmetry in robot assembly planning.

As noted in the introduction our study of Self-Evolving Behaviour (SEB) [1] has identified symmetry as underlying such behaviour and proposed two techniques, ‘multi-stage extensible generative grammars with closure’ and ‘endo-symmetry’, to represent and design mechanical developmental systems with SEB. However we have so far only described (a) using a hierarchy of extensible grammars augmented with symmetry information and (b) the closure of the grammar hierarchy using formal semantics, and are yet to discuss endo-symmetry.

Endo-symmetry is symmetry linkage across levels in the language hierarchy. Each language represents a different level of abstraction and detail of the one design. Each language is used to describe design constraints (in terms of symmetries) at one level of abstraction. It is not surprising then that symmetries are related across levels, for as we progress down the hierarchy each language is more concrete and refines the design and as we move up the hierarchy each language is more abstract and more general.

7. Results

This investigation has correlated the architecture of self-assembling automata and geometric features with localised and specific symmetry. Parametric prototypes incorporating these geometric features can be placed in a library and accessed from a feature based computer aided design (CAD) systems. These features also suggest primitive functions, such as point, line and space-groups, which could be added to the command structure of CAD systems for self-assembling structures.

When such a CAD system is based on a hierarchy of models each defined with formal grammars and linked via formal semantics the model hierarchy can be integrated and closure defined. Constraints between models (across abstraction levels) allow self-assembling systems to be defined and provide a tool to further investigate self-evolving behaviour.

In future work this CAD + Formal Language system could be combined with a compositional simulation tool [3] to produce simulation based design (SBD) software tools, and studies of self-assembling structures extended to also consider collision detection and dynamic issues.

The study has not investigated requirements of components and connections beyond geometry and kinematics (e.g. acceptable strength and deflection limits). These requirements are determined by the structural requirements at each site within particular configurations, whether final or temporary, during the processes of (re)construction and operation. The development of strategies to keep components within specification will be considered in a separate study.

References

1 Hollier, W., Burvill, C. and Davoren, J. (2005) *The Design of Self Evolving Behaviour: Self-evolution-I*, 15th International Conference on Engineering Design, these proceedings

2 Shah, J.J. and Mantyla, M. (1995) *Parametric and Feature-Based CAD/CAM*, Wiley-Interscience Publications

3 Paredis, C., Diaz-Calderon, A., Sinha, R. and Khosla, P. (2000) *Composable Models for Simulation-Based Design*, Tech. Report ICES-04-21-00, Institute for Complex Engineered Systems, Carnegie Mellon University

4 Hollier, W. (1991) *Self-Assembling Bridge: Modular Automation and Construction*, Concept Demonstrator proposed to Australian Defence Industry Development Program, EnGen Institute

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

6 Edmondson, A. (1987) *A Fuller Explanation: The Synergetic Geometry of R. Buckminster Fuller*, Burkhauser

7 Van den Hurk, J., Jess, J. (1998) *System Level Hardware / Software Co-design: An Industrial Approach*, Kluwer Academic

8 Waxman, R., Berge, J., et al (1996) *High-Level System Modeling: Specification and Design Methodologies*, Kluwer Academic

9 Reichl, H. (1990) *Micro System Technologies 90*, Springer Verlag

10 Whitney, D. (1996) *Why Mechanical Design cannot be like VLSI design*, MIT available at: http://esd..mit.edu/esd_books/whitnet/whitney_online.html

11 Chase, S. (2002) *A Systematic Method of Re-design: Using Function, Behaviour and Structure to facilitate grammar transformation*, in *eCAADe19, Architectural Information Management: Proceedings of the 19th Conference on Education in Computer Aided Architectural Design in Europe* Ed H Penttilä (Otamedia Oy, Espoo, Finland) 18-24, available at: www.hut.fi/events/ecaade/ E2001presentations/01_02_chase.pdf

12 Liu, X (1990) *Symmetry groups in Robotic assembly Planning* PhD Thesis, University of Massachusetts

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