Constructive Memory for Situated Design Agents

Authors: Pak-San Liew and John S. Gero

Postal Address: Key Centre of Design Computing and Cognition,
School of Architecture, Design Science and Planning,
University of Sydney NSW 2006 Australia

Courier Address: Room 290
Wilkinson Building (GO4)
148 City Road
University of Sydney
Chippendale NSW Australia

Phone: +61-2-9351 2328

Fax: +61-2-9351 3031

Email: paksan/john@arch.usyd.edu.au

Number of Pages: 45

Number of Tables: 0

Number of Figures: 22

Full Title

Constructive Memory for Situated Design Agents

Abstract

Design is situated. Situatedness in designing entails the explicit consideration of the state of the environment, the knowledge and experiences of the designer and the interactions between the designer and the environment during designing. Central to the notion of situatedness is the notion of design situation and constructive memory. A design situation models a particular state of interaction between a design agent and the environment at a particular point in time. Memory construction occurs whenever a design agent uses past experiences and knowledge within the current design environment in a situated manner. This paper is concerned with the development of an agent-based computational design tool that takes into consideration the notion of situatedness in designing. A key element of this tool is a constructive memory system that supports the dynamic nature of designing where memories of past experiences are constructed as required by the current situation. The behaviours of the memory system are illustrated through a series of experiments that demonstrates the effect of learning across various situations according to interactions with the environment.

Keywords

Situated design agent, constructive memory.

1.0 Introduction

Designing is an intellectual activity. Through designing, humans improve all aspects of their conditions via physical changes. Gero (1990) characterizes designing as "a goal-oriented, constrained, decision-making, exploration and learning activity which operates within a context which depends on the designer's perception of the context". An important characteristic of designing is its unpredictability (Goldschmidt, 1997) due to its dependency on the current state of the environment within which the designer operates, the current state of the designer, and the interactions between the designer and the environment. These dependencies are modelled by the notion of situatedness in designing which emphasizes the interrelationship between designers' knowledge and experiences and their application. Situatedness is concerned with locating everything in a context so that the decisions that are taken are a function of both the situation and the way the situation is constructed or interpreted. This is in contrast to a static world assumption where knowledge is treated as unrelated to its locus of application (Gero, 1998a).

A concept closely related to the situatedness of designing is that of constructive memory. Constructive memory implies that memory is not a static imprint of an experience that is available for recall later through appropriate indexing. Design experiences are stored and the memories of them are constructed in response to a demand on these experiences so that it is possible to answer queries, which could not have been conceived of when that experience first occurred. This view of memory aligns itself well with the concept of situatedness, as the memory of an experience is a function of the situation in which the question which provokes the construction of that memory is asked. Considered together with the situatedness of designing, a constructive memory system based on design experiences facilitates the creation of a dynamic memory system. This flexible, open-ended learning system has the ability to adjust and change itself to reflect its experience.

It is claimed that through the creation of the computational constructive memory system proposed in this paper, the situatedness of designing can be realized to facilitate the creation of a computational situated design agent.

2.0 Model of a Constructive Memory System

Figure 1 illustrates a model of constructive memory that resides within the framework of a situated design agent. Three memory construction processes are illustrated and each carries out the following operations:

- Cueing: The memory system is first cued by a demand from the current situation.
- Activation and selection: A series of experiences is activated and one is selected for use.
- Memory construction: Memory is constructed according to the selected experience.
- Incorporation: The constructed memory is incorporated into the system for subsequent use.
- Grounding: The activated experience is grounded if it is used.

Memory construction commences with a demand for a memory. The memory system is cued according to the current situation. This situation is determined by the current state of the environment and the design agent. The current state of the environment is represented by what information is currently available from the environment. The state of the agent is determined by the knowledge and experiences processed by the memory system. Through cueing, associated design experiences (○, ●, ● or ● in Figure 1) are activated for use during memory construction. One or more of these experiences is selected for subsequent memory construction.

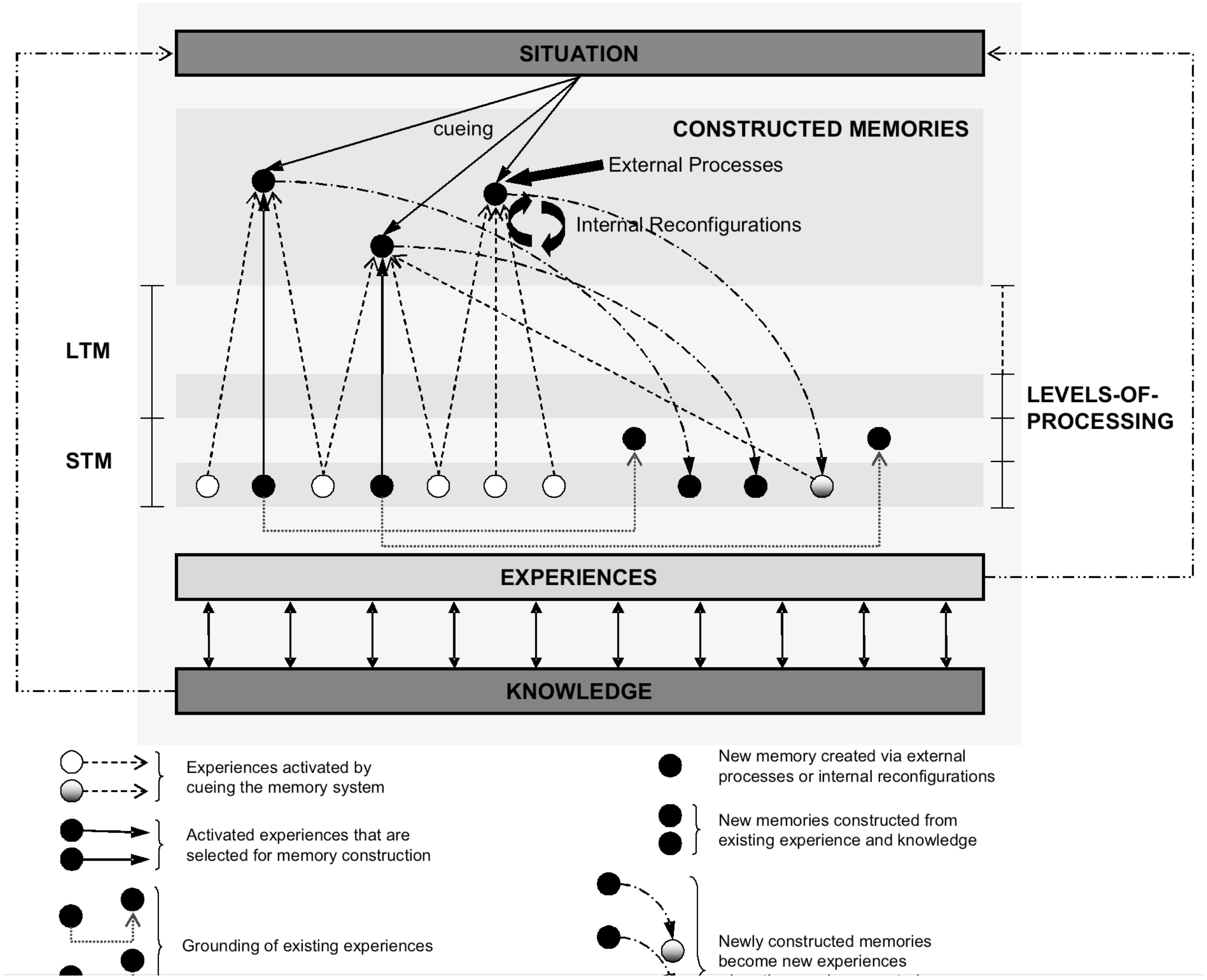

Figure 1. Model of a constructive memory system (extended from (Gero 1999a))

When an experience (or) has been selected, the knowledge associated with this experience is used to determine what gets extracted from the environment and the way in which the extracted information is structured. The knowledge and experiences form part of the situation and affect the kinds of memories that are constructed. The selection of an experience from those activated by the cue is determined by learning that occurred previously. Once an experience has been selected and used for memory construction, it is grounded. Memories that are constructed (and) are incorporated into the memory system as new experiences (and). Memory construction depends on the current number of experiences contained within the memory system and their respective levels-of-processing (described later).

2.1 Learning

The design experiences processed by a situated design agent are not encoded a priori. The agent starts off with a basic set of capabilities and gains experiences as it carries out its design tasks. Learning occurs when the design agent modifies the way pre-existing experiences are used and when the agent gains new experiences through its interactions with the environment. These two forms of learning are modelled by the notions of *long-term learning* and *constructive learning* respectively.

2.1.1 Long-Term Learning

Long-term learning affects the way pre-existing design experiences are used during memory construction over a period of time. This form of learning is effected through the *experiential grounding* of an experience that has been used directly in the current situation. An experience that has been grounded frequently is more likely to be used in subsequent memory constructions. The effects of grounding are modelled by the notion of levels-of-processing.

The process of grounding entails the embedding of an artificial agent into its environment such that its behaviour and the mechanisms, representations, etc. underlying this behaviour are intrinsic and meaningful to the agent itself, rather than dependent on an external designer or observer (Ziemke, 1999). The meaningfulness of these mechanisms, representations, etc. is determined by their usefulness in assisting the agent in attaining any of its goals (Nehaniv, 1999). While *symbolic groundings* deal with the provision of meanings for symbolic representations, *experiential grounding* deals with the provision of meaning for the experiences processed by the artificial agent. In order for an experience to have meaning for an agent, the consequence of actions based on this experience must have an impact on the agent itself (Nehaniv & Dautenhahn, 1998). Underlying this meaningfulness of experience is the applicability of the experience through its usage. The interaction between the agent and the environment plays an important role in experiential-based grounding and has an effect on the use of pre-existing experiences processed by the agent. An agent, with its current experiences, creates a particular view of the external world. If this view matches the other view created by agent through its interactions with the external environment, it becomes useful in predicting the effects of a course of actions. The experience underlying this prediction is labelled useful and is grounded. This process of grounding models the effect of long-term learning.

Different *levels-of-processing* refer to the amount of grounding experiences received through previous use. A highly grounded experience has a higher level of processing and is used more readily in the subsequent operation of the memory system.

The different levels-of-processing are divided into two regions according to a pre-defined value of grounding to demarcate Short-Term Memory (STM) and Long-Term Memory (LTM), Figure 1. An experience starts off in the short-term memory after its first creation by memory construction. This experience progresses towards long-term memory as it gains grounding through its usage in subsequent memory construction. As its usage increases, the groundings of other experiences decay and the other experiences move towards short-term memory. Experiences at the lowest end of short-term memory are purged out of the system eventually. These behaviours treat short-term memory as a gateway for experience to get into and out of the memory system.

To illustrate the effects of long-term learning, consider the case where a similar past experience (◍ in Figure 1) is used directly in the current situation, a new experience (●) is created from the memory (○) constructed from that experience. The constructed memory (○) forms an interpretation of the previous experience (◍) according to the current interactions between the agent and the environment. The experience, ◍, provides the basis for interpreting the environment according to the current objective of the agent. Actions to be carried out are determined by this experience according to a similar situation in the past. After the memory (○) of this experience has been constructed, this experience (◍) is labelled useful and grounded. In subsequent memory construction, this experience has more chance of being used due to its higher level-of-processing resulting from the current grounding. All experiences (pre-existing and those created by incorporating constructed memories into the system) are available for use in the subsequent construction of memories and they are subjected to grounding when they form the basis for subsequent memory construction.

2.1.2 Constructive Learning

Constructive learning is another mechanism that uses similar pre-existing design experiences and creates new experiences called memories during memory construction. The usage of experience is not direct as in the case of long-term learning. During memory construction, analogical processes are used to relate the current situation to some other situations where concepts are found to be similar at some other level of abstraction (Gero & Rosenman, 1989). Elements of past experience are mapped into elements of the current situation to form a coherent memory structure for effecting actions into the environment. The effect of constructive learning is a change in structure of the memory system through a reconfiguration of existing memory components or the addition of new ones.

New information is added via constructive learning through two methods. In the first instance, new experience is added through external reasoning processes such as analogy, combination, mutation and designing from first principles (Rosenman & Gero, 1993) and reformulation processes (Gero, 1998c) that deal with re-parameterization and parameter expansions. An example of this is the addition of a new experience created by drawing an analogy between the current situation and a previous experience. In the second instance, the memory system reconfigures itself to produce new experiences. This reconfiguration takes the form of a reinterpretation of a pre-existing experience where the agent reinterprets its experiences based on the information available from the environment. This occurs when the agent learns about better performing elements from the environment and uses them to modify its past experience.

To illustrate the effects of constructive learning, consider the case when a similar design experience is used as a basis for an external process that produces new experiences. Taking analogy (Qian, 1994) as an example of an external process, elements in the current situation are matched with elements in a similar situation in the past at some abstraction level so that a subsequent mapping process can be carried out. This mapping process connects elements in the current situation to the corresponding elements from a past experience to form a new interpretation of the past experience. The actions carried out in the past experience are transformed into the appropriate form so that it can be used as the basis for action in the current situation. The third memory construction in Figure 1 illustrates this case where a memory (●) is created by the process of constructive learning. The new experience (◎) is created when the constructed memory is incorporated into the memory system for subsequent use. When this experience (◎) is used as a basis for constructing another memory in another design session, it is grounded to a new level-of-processing to model long-term learning.

2.2 Behaviour of a Constructive Memory System

The behaviour of a constructive memory system within a situated environment has the following characteristics:

- memory is modelled as a dynamic process, not as a static imprint to be stored in a specific location and retrieved for use later;
- the operations for constructing memories are influenced by the situatedness of designing;
- construction of memories does not rely on exact matching between what the agent has in the memory system and what is available in the current designing environment; and
- memories that are constructed may not match the original experience exactly as it was first experienced, but change according to when, where and what the memory system is cued with.

The use of past design experiences entails a construction process where memory traces[1] are reconfigured and combined or created through external processes according to the current design situation. Once the memory of a past experience has been constructed with all the necessary memory traces and served its purpose in the current design session, the newly created memory is disassembled into its constituents of memory traces and integrated gradually into the memory system. Subsequent use of the experiences residing within the memory system involves a recollection and reconfiguration of these traces according to the current design situation.

[1] *Memory traces model individual elements of design experiences.*

3.0 Design Experience

Design experiences contained within a constructive memory system play a critical role in designing as they form the basis for the agent's behaviour in the current situation according to similar situations it encountered in the past. A design experience encapsulates the elements, processes and knowledge of a past design session and make them available for use in the current situation. The *design experience* used in this paper encapsulates ideas from the variables and processes operating within a situated design agent (Gero & Fujii, 2000), the situated function-behaviour-structure framework of designing (Gero & Kannengiesser, 2000; Gero & Kannengiesser, 2002) and a design prototype (Gero, 1990). Gero (2000) provides a high-level description of a situated agent for designing. An agent-based approach (Russell & Norvig, 1995) is used to construct a framework for concept formation.

Sensation is an information extraction process. Information from the environment is extracted into the agent according to the current configuration of the sensor. This configuration is dictated by perception that biases the way the sensor works to limit the type and amount of information sensed. Perception, after creating the bias for sensation, structures the sensory data into coherent structures (percepts) as required by conception. Conception operates on existing knowledge and experiences to form concepts relevant for the current design situation.

The variables defining the *concept*, *percept* and *sensory data* in a situated design agent are incorporated directly into equivalent forms within a design experience. The respective processes that produce these variables (conception, perception and sensation) are also used to produce the contents of a design experience. These processes are intrinsic to the design agent itself and they are external to the design experiences contained within the memory system.

The behaviours of conception, perception and sensation are affected by the focused concept contained within a design experience. A focused concept represents a form of knowledge processed by the agent. It provides the initial concept to be used for deriving new ones during conception and biases perception by controlling the way percepts are created from sensory data. Perception, based on the information provided by the focused concept, adjusts the sensitivity of the sensor to limit what is extracted (sensed) from the environment to form sensory data.

To make the contents of a design experience specific to the design domain, the notions of an *external world*, *interpreted world* and *expected world* (Gero & Kannengiesser, 2000; Gero & Kannengiesser, 2002) are incorporated into a design experience.

The external world is the world outside the agent and exists in terms of structure for the agent. The interpreted world is the internal world of the agent created from the knowledge, experiences, percepts and concepts of the agent. The agent interacts with the interpretation of the external and internal worlds. The expected world is the world that the agent anticipates its actions will produce. It is the agent's internal environment for predicting and deciding on actions according to its goals and its interpretations of the current state of the world.

These three worlds are linked together by: interpretations that transform sensed variables from the external world into interpretations of sensory experiences, percepts and concepts within the interpreted world; hypothesizing that utilizes aspects of the interpreted world as goals in the expected world to suggest actions in the external world in accordance to the agent's goals; and action that effects a change in the external world.

The *exogenous variable* (Gero & Fujii, 2000), describing that part of the environment the agent is interacting with, is modelled as an *external structure* (forming the external world) within a design experience. The *interpretation* (creating the interpreted world) of this external structure is created by incorporating function-behaviour-structure information to form a concept for later evaluation. The *expectation* (forming the expected world) of the agent is modelled via extracting the function-behaviour-structure information from the focused concept contained within the design experience.

3.1 Representation of Design Experience

The representations of design situation and design experience are based on work on concept formation by a situated design agent (Gero & Fujii, 2000).

A design situation models a particular state of interaction between a design agent and the environment at a particular point in time. It is a snapshot of all the variables that define the internal state of the agent and the external state of that part of the environment that the agent is interacting with. The expectation and interpretation of the agent may or may not be in agreement with each other within a design situation. This agreement is based on the comparison of the behaviours contained within the expectation and interpretation of the environment. A disagreement implies that there is a difference in terms of either behavioural attributes or attribute values and this forms the basis for actions. A design experience models the transition of state from one design situation to another situation in which the expectation is anticipated to agree with the interpretation. The transition is represented by the initial and final state together with the action that facilitates the transition. Essentially, a design experience consists of two design situations and the action that transforms the first situation into the second situation. Within the first situation, the expectation and interpretation are not in agreement. In the second situation, the expectation agrees with the interpretation. Symbolically, a design experience, E, is represented by a collection of constituent parts:

$$E = \{V^i, S^i, P, Q^i, H^i, A, H^f, Q^f, V^f, S^f\}$$

where:

A represents action on the agent or on the environment,

H^i represents the initial focused concept,

H^f represents the final focused concept,

P represents the percept,

Q^i represents the initial expectation,

Q^f represents the final expectation,

S^i represents the initial exogenous sensory data,

S^f represents the final exogenous sensory data,

V^i represents the initial exogenous variable,

V^f represents the final exogenous variable, and

$Q^x = \{F, B, S\}$ with Q^x denoting the initial expectation if $x = i$ or the final expectation when $x = f$, B denoting the behaviour of the expectation, F denoting the function of the expectation and S denoting structure of the expectation.

The *initial exogenous variable*, V^i defines that part of the environment the agent is dealing with. In the case of a design agent, V^i represents the *external structure* (Gero & Kannengiesser, 2002) of a design artefact to be manipulated by the agent. A design agent may or may not operate on the entire external structure. What is focused upon is a function of the current focused concept (H)[2] based on the current design objectives (Gero & Fujii, 2000). The sets of initial exogenous variables[3] encountered by the design agent are different for unfamiliar environments.

An *action*, A, is either a set of operations an agent performs internally on itself, or a set of operations performed externally on the environment (that is, on the design artefact to be manipulated). An internal action changes the behaviour of the agent by altering variables representing its internal states. An external action transforms the initial exogenous variable (V^i) to a final exogenous variable (V^f) according to the current design objectives.

---

[2] *(H) represents Hi and/or H_f depending on the initial or final design situation.*

[3] V^i refers to a single variable that represents a reference to an external structure.

The final exogenous variable, V^f, and sensory data, S^f, model the state of the artefact and its associated data sensed by the agent, after the agent has performed the required action to eliminate the disagreement between its expectation and interpretation of the environment.

For a design agent, a focused concept (H) represents a *design prototype* (Gero, 1990) that the agent is currently using. This prototype can either be retrieved from the memory system or derived from existing prototypes according to the current design objectives through external processes such as analogy.

The design prototype representing the current focused concept contains information pertaining to the function, behaviour and structure (FBS) of a design artefact in term of a dependency network (Gero, Tham, & Lee, 1992). This part of the prototype is extracted via a conception process to form the expectation (Gero & Kannengiesser, 2002), (Q)[4] of the agent according to the current focused concept. Each design prototype has procedural knowledge that defines how the prototype can be used to achieve a specific purpose. This procedure is used to derive the required external action to be effected into the environment for the current situation.

The FBS information from the focused concept is also used in perception to bias the sensation process and structure the exogenous sensory data (S)[5]. The functional information dictates which part of the environment will be focused upon and sensed into the agent. Only parts with the same function as that of the focused concept are extracted from the environment. The sensory data are configured into a complete structure to form percepts (P), a form of interpreted structure (Gero & Kannengiesser, 2002) during perception. The interpretation (Gero & Kannengiesser, 2002) of the environment is created from this percept during memory construction. This interpretation is represented by a dependency network that contains FBS information about the interpreted structure. The behavioural and structural part of the interpretation are either computed from or extracted directly from an external repository of design prototypes.

3.2 Example of a Design Experience

Figure 2 shows an example of a design experience created from redesigning a mechanical assembly for ease of assembly. The memory system is first cued according to the current situation. This situation is dictated by the external structure (V^i_1) from the environment and the current focused concepts (H^i_1 and H^f_1), Figure 3. A series of design experiences are activated by this cue.

Assume that the activated design experience that is selected for use proposes the same focused concept for use in the current environment. The environment is sensed according to this focused concept, Figure 4.

During perception, Figure 4, the sensory data S^i_9 are extracted from the environment and structured according to any applicable functional, behavioural and structural information of the focused concept to form an interpreted structure (percept) P_{10} of the environment.

An interpretation of the environment is formed by conception, Figure 5. This process uses functional information from the focused concept, behavioural information extracted and modified (if necessary) from an external repository of design prototypes and structural information from the environment to create a dependency network that represents the interpretation. The behavioural and structural information of the interpretation is modified to a form suitable for processing during this conception process. This form is dictated by the functional and behavioural information from the focused concept. The expectation of the environment, Q^i_1, is created from the focused concept by extracting the dependency network from the design prototype representation, Figure 5.

---

[4] *(Q) represents either Q_i and/or Q_f depending on the initial or final design situation.*

[5] *(S) represents either S_i and/or S_f depending on the initial or final situation.*

The interpretation and expectation are matched and the required behaviour is compared. Assuming that the expectation has better performance, the two concepts (interpretation and expectation) are mapped to obtain their corresponding structures, Figure 5.

Based on the corresponding structures obtained from the mapping process, the required action, A_1, to redesign the external structure is derived from the focused concept. The action A_1 is effected onto the environment to create a new structure, V^f_2. The sensation, perception, conception and matching process is repeated to form the final situation, Figure 2.

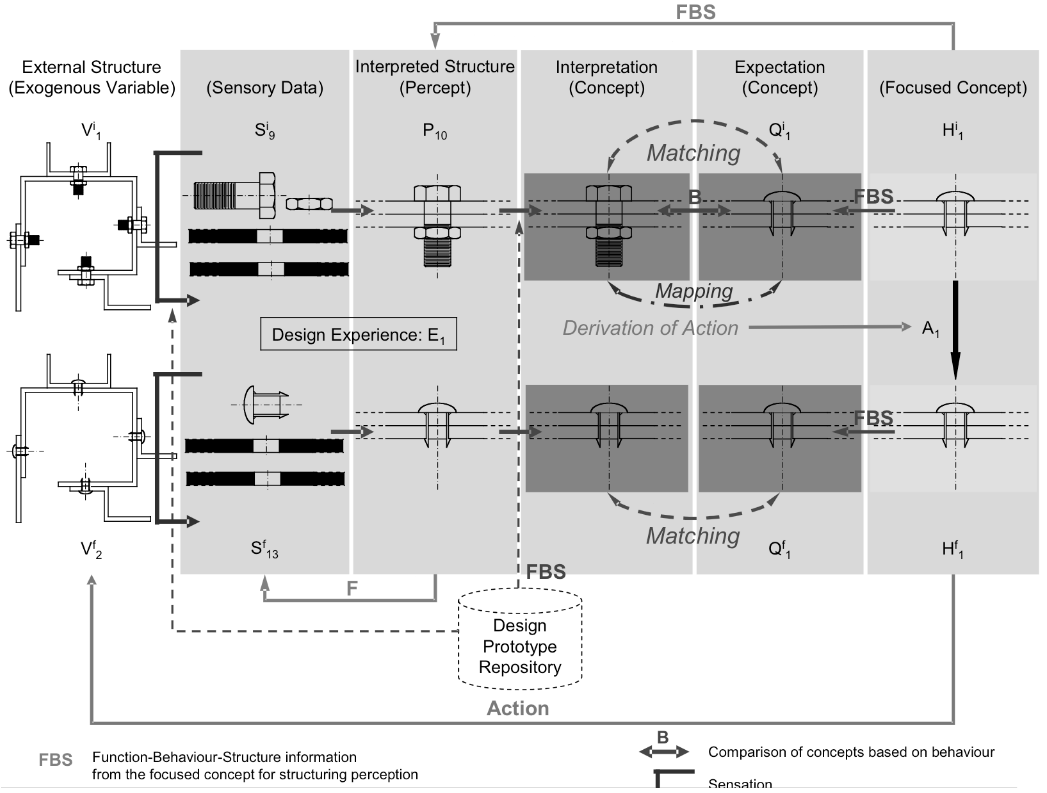

Figure 2. A sample design experience

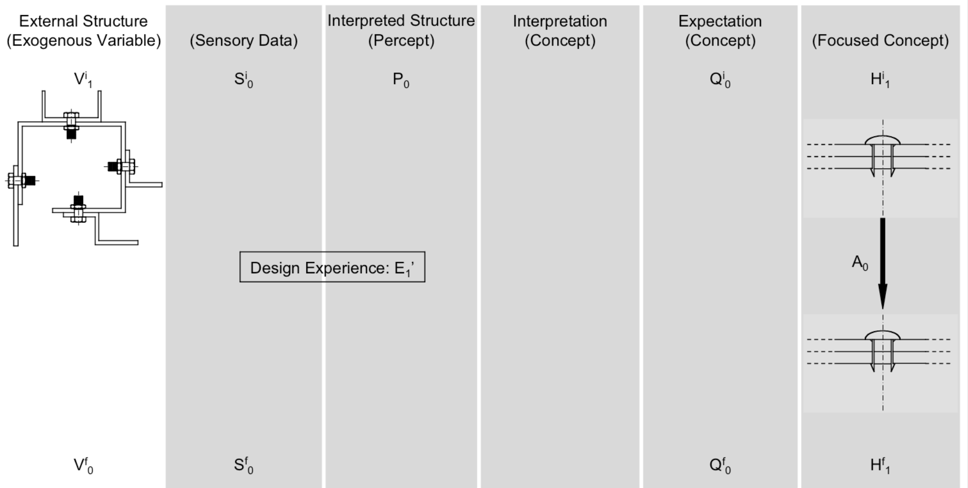

Figure 3. A cue sent into the memory system

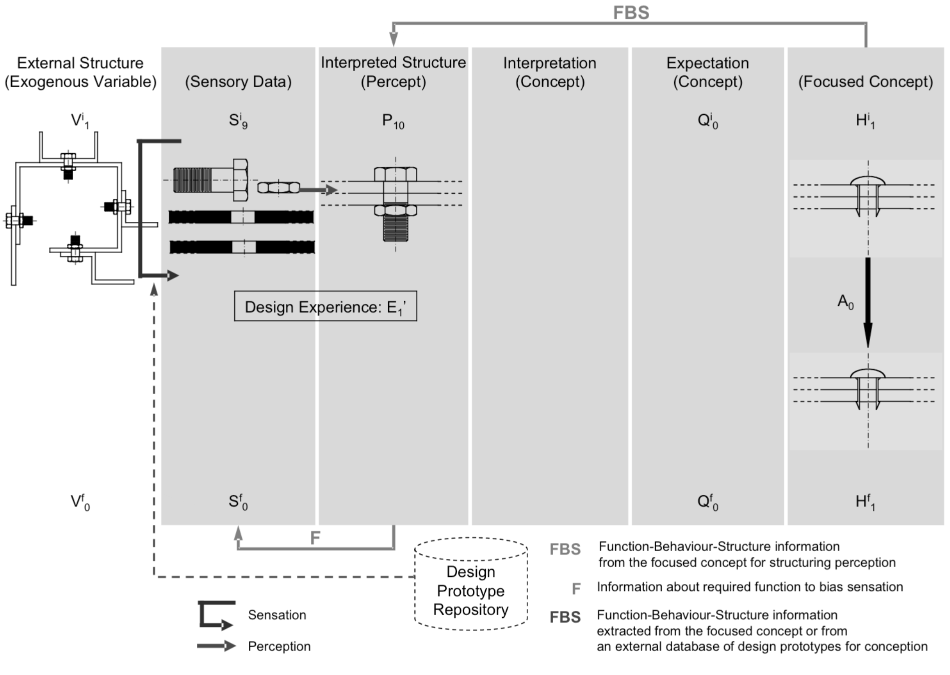

Figure 4. Sensation and perception

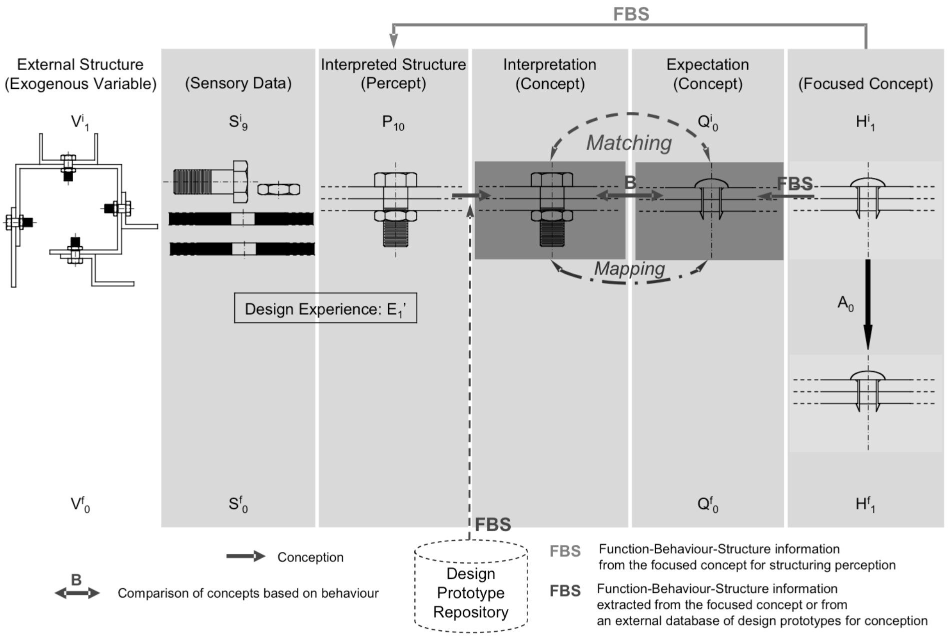

Figure 5. Conception to create interpretation and expectation for matching, mapping and comparison of behaviour

4.0 Overall Structure of a Situated Design Agent

The constructive memory system developed in this paper operates within a situated design agent. Various behaviours that result from cognitive models of memory are incorporated into the memory system through the mechanisms provided by the agent's structure.

Figure 6 illustrates a high-level view of the functional components that make up the situated design agent. The functionalities of these components have been described in terms of the operations of the state-of-affair (SOA) and course-of-event (COE) memory system (Gero & Fujii, 2000).

4.1.1 Architecture of a Constructive Memory System

The design of the various components of the constructive memory system is derived from cognitive studies of the human memory system. In particular the following models of human memory are adopted: modal model of human memory system (Atkinson & Shiffrin, 1968); multiple-component model of long-term memory (Baddeley, 1999); and multiple-component workspace model of human working memory (Logie, 1995). The aim of this model is not to model the behaviours of human memory, but to utilize the components from various models that provide the required characteristics of a computational constructive memory system.

Figure 6 illustrates the architecture of the proposed memory system for the situated design agent. The paths in Figure 6 are shared between processes to avoid overcrowding the diagram. They can be taken as directed associations between two entities, i.e. information or instructions flow in the indicated direction.

The long-term memory (LTM) and short-term memory (STM) system generally deals with experiences and knowledge processed by the agent, while the working memory is a workspace where most reasoning occurs. Memories of design experiences that are constructed in the working memory are transferred to the LTM through the STM.

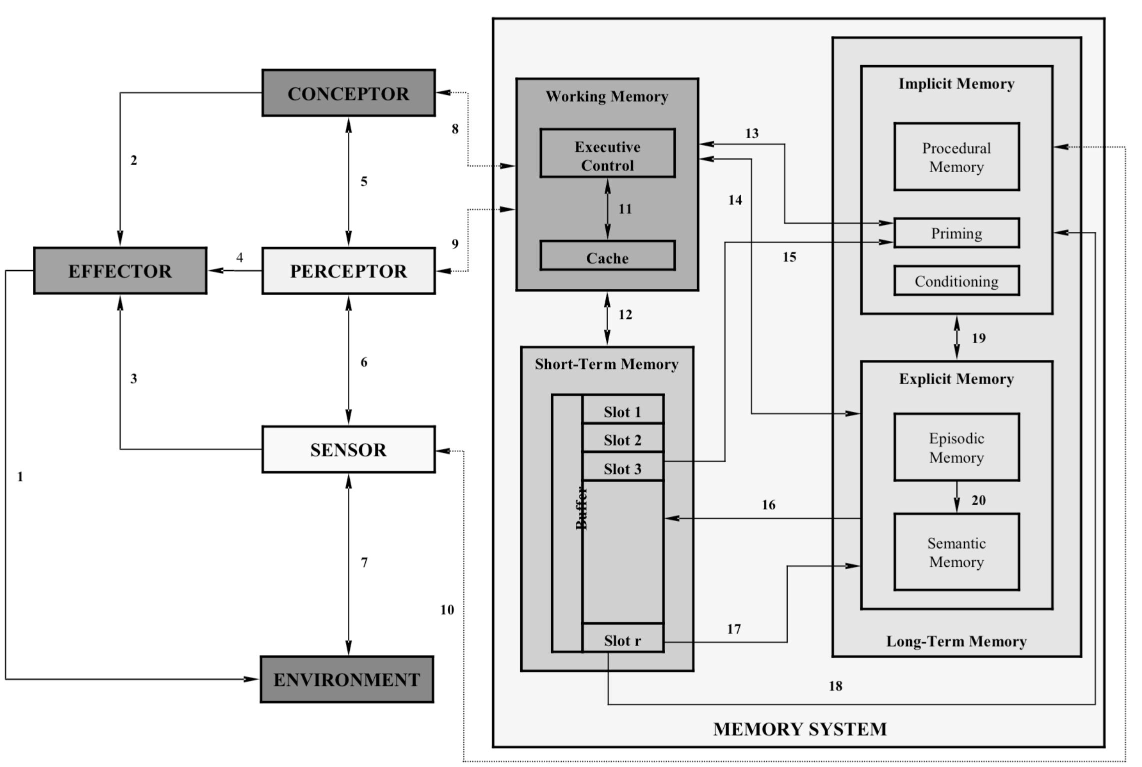

Figure 6. Conceptual architecture of a computational memory system (numbered paths are indicated as (#) in the text where # is the number as shown in the diagram) within a situated design agent (from (Gero & Fujii, 2000))

The working memory is also the working environment where executive control functions are performed. These processes selectively alter and/or add information to the working memory based on the content of the LTM (14) [the numbers in parentheses refer to the paths in Figure 6] or STM (12). Information within the working memory is combined with the stored knowledge and experiences, manipulated, interpreted and recombined to develop new knowledge, assist learning, form goals, and support interaction with the external environment. The reasoning that occurs within the working memory is design-related and is based on three main components: the executive control: where the hub of the agent control processes is located; the STM and LTM: where knowledge and experiences of the agent reside ((14) and (12)); and the cache: that provides the buffering for the reasoning processes (11). The control processes within the executive control are involved in the organization, coordination, manipulation, movement and comparison of information and knowledge (derived from the environment or from the explicit memory) for design-related reasoning. These processes deal with:

- strategy selection and hypothesis-testing procedures associated with concept formation problems during conception, perception, hypothesizing, expectation and devising (Gero & Fujii, 2000);
- the coding of percepts during perception where related stored knowledge and experiences from LTM or STM are activated and constructed within the working memory to allow the interpretation of current percepts in the current design session; and
- tasks where there is a clear solution (reactive process) based on the current bias.

Inputs arrive from the conceptor (8) and perceptor (9) during conception, the perceptor (9) during perception, the LTM (14) and STM (12) when they play host to activated representations and procedures from them.

The path (17) from the STM to the LTM indicates the build up of a long-term trace due to the memory trace's level-of-processing. The slots in Figure 6 are used to indicate the notion of the relative level-of-processing (and thus the likelihood of usage) of the various traces within STM. Both STM and LTM are used in the construction of memories in the working memory ((12) and (14)). The differences between memory traces residing in the STM and LTM lie in that memories in the STM are not as readily recalled in the memory construction process as those in the LTM.

The STM serves as a gateway into and out of the LTM. The path into the LTM occurs when memory traces are transferred from the working memory to the LTM through the STM. As soon as a memory of an experience has been constructed and served its purpose within the working memory, it is transferred into the STM as a new memory trace (12). Memory traces in the STM are integrated into the LTM gradually through their level-of-processing (17). STM acting as a gateway out of the LTM occurs when memory traces in the LTM that are not grounded frequently are transferred into the STM (16) and out of the memory system. These two actions model the phenomenon of long-term learning and the phenomenon of "use it or lose it" in the human memory system.

5.0 System Implementation

The implementation of the constructive part of episodic memory is based on a parallel distributed processing model of the medial temporal memory system by McClelland (1995). The Interactive Activation and Competition (IAC) network (McClelland, 1981) is used to implement this model. An IAC network is a neural network that uses an activation and competition mechanism to operate on a collection of nodes. The knowledge of an individual exemplar is captured in a network of nodes. An *instance node* and a set of corresponding *property nodes* are defined for each of the exemplars that the network knows. Each property node represents an attribute (property) of the instance node. The property nodes and instance nodes are arranged in groups of mutually exclusive inhibitory connections. The nodes are localized representations of concepts. These representations are activated via a spreading activation mechanism where activated exemplars activate the representation of their properties. Mutually exclusive property values compete in such as way that allows properties that are supported by a large subset of the active instance of the category to be reinforced and become strongly active while suppressing those that are not.

In the current implementation, the system's knowledge of an individual design experience is represented as an instance node linked to a set of property nodes through excitatory connections. The connected property nodes represent the constituents of a design experience such as action on the agent or on the environment; initial and final focused concepts; percept; initial and final expected percepts; initial and final exogenous sensory data; and initial and final exogenous variables, Figure 7.

During memory construction, a probe is created and presented to the IAC network as a cue for memory construction. The content of this probe is essentially a collection of available information about the current state of the external environment and the internal state of the agent that indicates which nodes within the network are to be clamped (given an external input). Once the probe has been set, the network is cycled and allowed to settle into an equilibrium state. The resulting instance nodes and their associated activated property nodes highlight similar design experiences that can be used in the current situation. Each activated experience has an associated activation value based on its usefulness in previous design sessions.

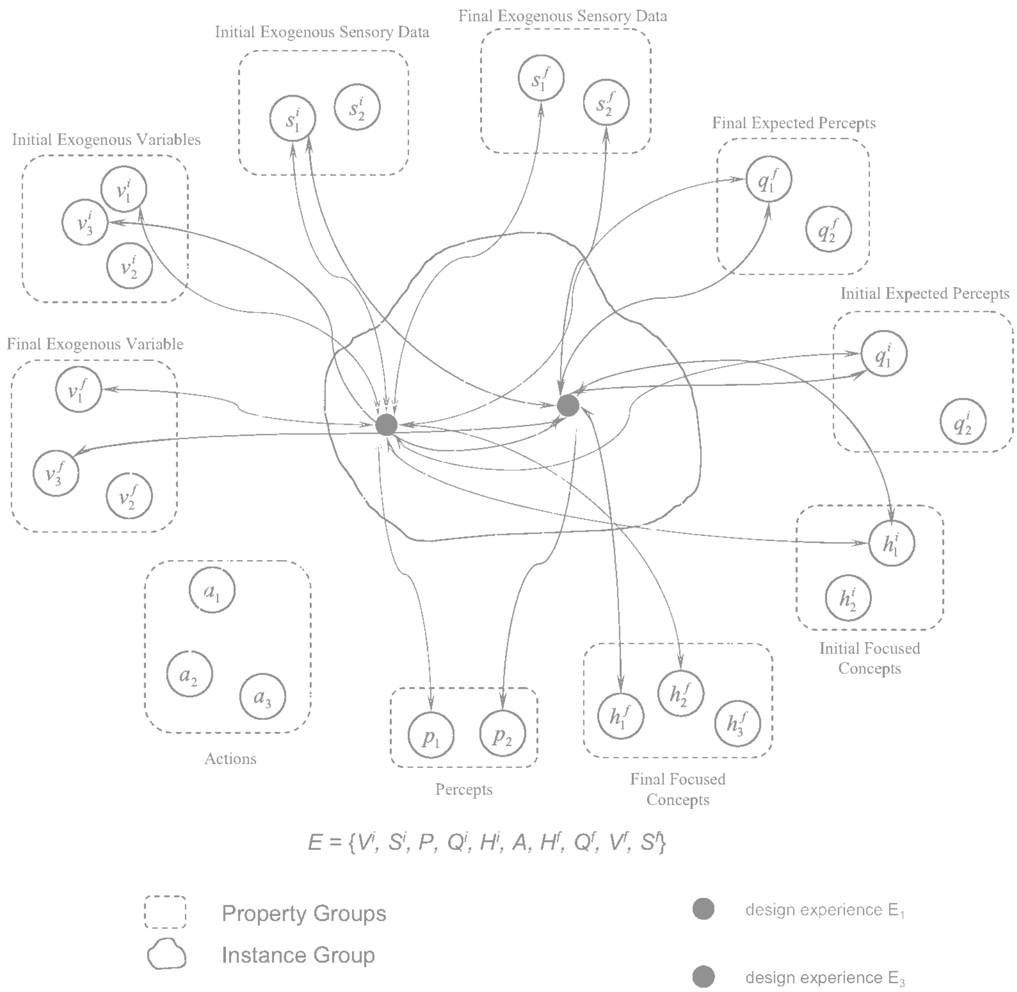

Figure 7. Representations of design experiences within an IAC network

5.1 IAC Formulation

The standard formulation of the IAC network (McClelland, 1988) is used as a base upon which learning mechanisms are incorporated. In an IAC network the total input, t_j, for node j is:

$$t_j = \alpha e_i + \sum w_{ji} a_i$$

where α represents the external input scaling, e_i represents the external input, w_{ji} represents the weight to node j from node i and a_i represents the activation of node i. During cueing, the probe contains the node j, e_j is set to 1.0. α is set to a global value of 4.0 for the entire network.

After the total input has been calculated for all the nodes, activations are updated according to the following equations:

$$(t_j > 0): \quad \Delta a_j = s\{(a_{\max} - a_j)t_j - d(a_j - r)\}$$

$$(t_j < 0): \quad \Delta a_j = s\{(a_j - a_{\min})t_j - d(a_j - r)\}$$

where Δa_j denotes the amount by which to change a_j.

This formulation differs from the original IAC with the additional parameter step, *s*, for controlling the rate at which the node activations are changed. *s* is set to a global value of 0.1 for the entire network. a_{max} is set to 1.0, a_{min} is set to -0.2, rest, *r,* is set to -0.1 and decay, *d*, is 1.0. The network starts off with all nodes having an initial resting activation value of –0.1. Positive links are set to 1.0 and negative weights are set to –1.0[6]. These settings are used for the initial network before any learning occurs. These values form the starting point for all newly created or configured design experiences. The learning process causes the positive links between nodes to increase in value while the decay process reduces the corresponding positive link values.

5.2 Learning Mechanism

The original IAC network (McClelland, 1981) does not provide any learning mechanism. Learning can occur through:

- grounding via adjusting the weights between existing nodes of a design experience;
- constructive learning via adding new experiences in terms of new nodes; or
- constructive learning via reconfiguring existing nodes to represent new experiences.

These learning processes occur in light of the newly acquired experiences by the situated design agent.

5.2.1 Long-Term Learning via Weight Modification

Computer implementations of long-term learning are exemplified in the modification of weights in neural networks (Haykin, 1998; Medler, 1998). Long-term learning in neural networks deals with the change of the interconnection weights within the network of nodes and links.

Learning via changing of weights (long-term learning/grounding) makes memory traces more readily available for the subsequent memory construction process. Only the strengths of excitatory (positive) links are allowed to change in the implementation. Negative links are not changed locally as their global settings are used to control blending.

The learning mechanism is formulated as (Medler, 1998):

$$\Delta w_{ij} = \beta\{(w_{\max} - w_{ij})a_i a_j - \delta w_{ij}\}$$

where Δw_{ij} denotes the amount by which to change the link between node *i* and node *j*, β denotes the learning rate, δ denotes the weight decay factor and w_{max} denotes the maximum weight value. This

[6] These parameters are determined by the configuration of parameter values that produces a set of stable activation values for the nodes and links within the IAC network across different memory construction cycles.

formulation of learning is similar to the one for computing activation values of nodes. The learning process occurs as a separate process only after memory construction has taken place.

The learning processes based on weight adjustments is implemented by presenting the memory system with the memory traces representing the design experience that the design agent found to be useful. This emphasis of particular memory traces strengthens the interconnections between the nodes involved.

5.2.2 Constructive Learning

Computer implementations of constructive learning are exemplified in the modification of neural network structures (Mandic & Chambers, 2001). Constructive learning in neural networks deals with the changing of architecture or interconnections within the network. This occurs with the addition of new nodes into the original IAC network, Figure 8. These ontogenic networks (Fiesler & Beale, 1997) change their topology during learning. There are two basic classes of constructive learning: network growing and network pruning. In the network growing approach (Hoehfeld & Fahlman, 1992), the network starts small and grows as it learns. Network pruning starts with a large network and reduces its size until the desired ratio between accuracy and network size is reached (Reed, 1993). The idea behind these learning mechanisms is similar to the concepts of ontogenic networks (Fiesler & Beale, 1997) where the topology of the network changes over time.

Constructive learning within the IAC via network growth can occur with the addition of new instance nodes created either by reconfiguring an existing instance node with new property nodes or by creating a new instance node from new property nodes. New property nodes are created as a result of external designing reasoning processes such as analogy, combination, mutation and designing from first principles (Rosenman & Gero, 1993) and reformulation processes (Gero, 1998c) that deal with re-parameterization and parameter expansions. Constructive learning via network pruning occurs through the forgetting process based on the grounding of memory traces. One implication of constructive learning is that the memory construction process can produce different design experiences for the same cue at different stages of the agent’s lifetime. What is constructed is a function of both what was originally experienced and what the agent has learned since it gained that experience.

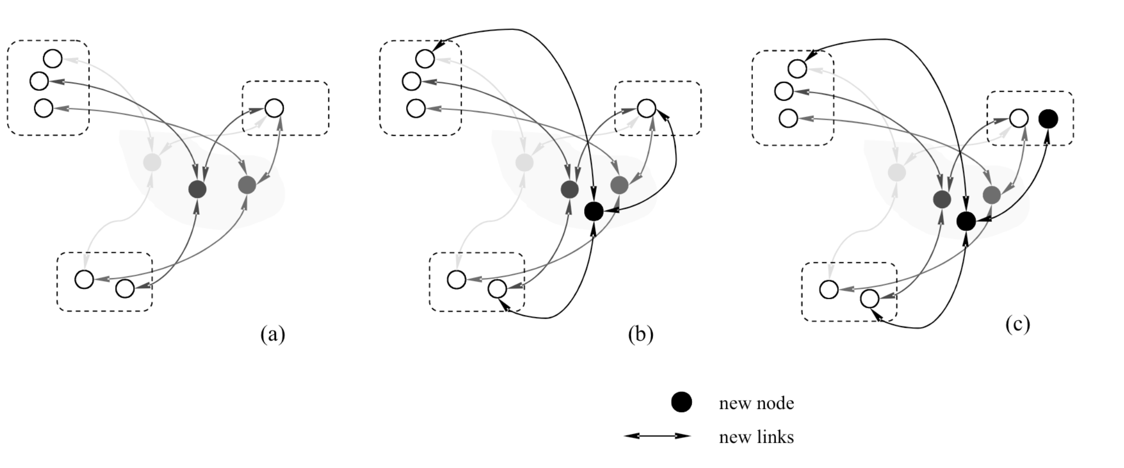

Figure 8. Comparison of the original network before learning (a) and after: (b) learning via adding a new instance node through reconfiguration, (c) learning via adding a new instance through adding a new property node

5.3 Implementation of STM and LTM

The STM and LTM are implemented as a multi-layered IAC network, Figure 9. The separation of instance nodes (together with their associated property nodes) into different layers is based on their levels-of-processing. The common interconnection strength between the instance node and its corresponding property nodes provides the basis for these levels.

In Figure 9, memory traces with high interconnection strengths are placed close to the left to indicate that the respective nodes are more durable within the memory system, while instances with low interconnection strengths are moved towards the right. As the grounded usage of an experience increases, it is repeatedly presented to the memory system to emphasize the relevant experiences through strengthening of the interconnections between the relevant nodes.

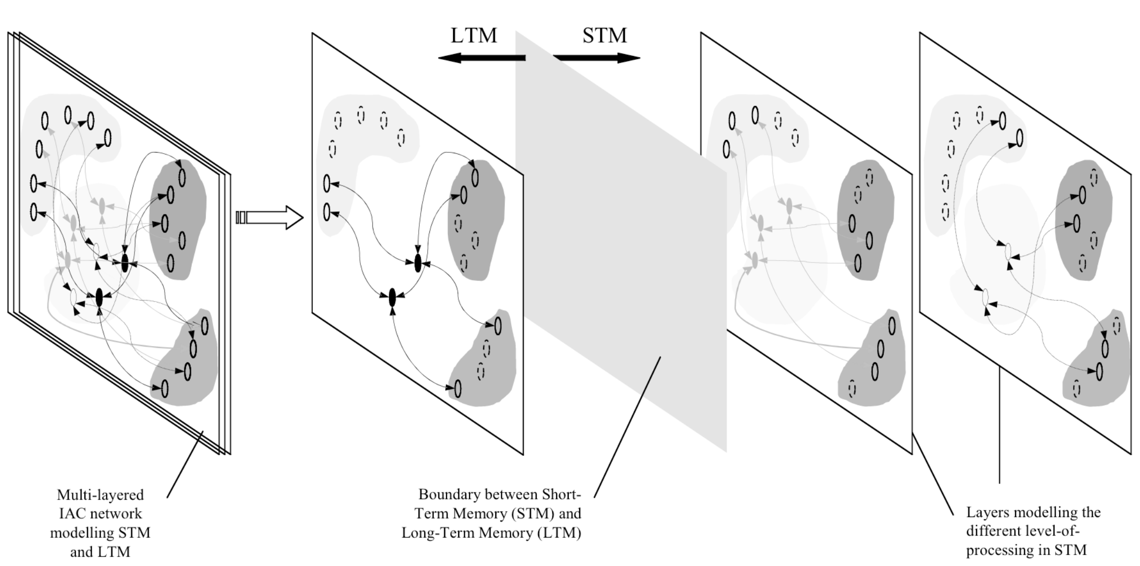

Figure 9. Demarcation of STM and LTM

A threshold value for interconnection strength is set up to delineate LTM and STM. This threshold can be visualized as a cutting plane that separates LTM from STM among the different layers in the modified IAC network. When memory traces are to be removed from the system (by a forgetting process) to increase computational performance, the traces at the lowest level (to the right of Figure 9) within the STM are purged from the system first.

Newly created memories start off in the STM (right side of Figure 9). As their usage becomes grounded they move towards the LTM (left side of Figure 9). This gradual incorporation of new experiences functions to avoid catastrophic interference. Catastrophic interference refers to high activation of the most recently learned experience during memory construction. As a result, the memory system would always respond with the most recently learned experience irregardless of what has been learnt previously.

All memory traces in STM and LTM are subject to a decay process. If the subsequent memory construction process does not reinforce these memory traces, their interconnection strengths will decay and they will move towards the STM region. It should be noted that the notion of STM and LTM is used in this paper to demarcate design experiences residing within the memory system according to their levels-of-processing. STM acts as a gateway into and out of the memory system. Design experiences within the STM and LTM are treated in the same way.

6.0 Experiments to Demonstrate the Effects of Grounding

Situatedness in designing provides a different perspective on the role of machine learning in designing. A common view of the role of machine learning in design is that of "learning to perform existing tasks better using available tools" where these tools are unchanged by their usage (Gero, 1998b). The tool undergoes a training phase where sample solutions are presented to the tool in order for it to extract the underlying relations between input (the problem) and output (the solution). During usage, these tools are not modified in any way to reflect what can be learned through interactions with the environment. Through consideration of the situatedness of designing, computer-aided design tools are no longer seen as passive aids to designers. These tools need to increase their usefulness through their use. Learning is seen as a means to change a tool so that its performance improves with usage through the acquisition or restructuring of knowledge. The quality of the knowledge is constantly improved through its use and as a consequence, when the tool is reused in a similar situation it will have different knowledge than it had before and thus potentially produces better results.

Two forms of learning that fulfil the above requirements are: long-term learning through grounding and constructive learning. Long-term learning is the focus of this paper. Two sets of experiments are conducted to demonstrate the different *performances* and *responses* due to grounding (long-term learning). Different performances are indicated by different activation values of the same design experiences across different memory construction cycles. This difference in activation values cause the memory system to operate in different ways during memory construction. The response of the system is obtained from the system by extracting the design experience with the maximum activated value for each memory construction cycle. The final experiment is conducted to illustrate the mechanism to sustain high activation values through grounding.

The examples are used purely as a vehicle to illustrate the concepts and operations of the memory system and not to justify the use of the system within their domains. The structures can be redesigned in many other ways, some of which may employ mechanisms that are simpler than those proposed here. The memory system offers possibilities that are not directly derivable from these other methods.

6.1 Application Context: Design for Assembly (DFA)

The experiments deal with the redesign of mechanical assemblies for ease of assembly. Redesign is taken as the modification of a pre-existing design according to the current goal of a design agent. The current design goal is modelled through a series of design guidelines, principles or rules that fulfil the various objectives of the goal. For example, an agent performing redesign for the purpose of Design for Assembly (DFA) (Boothroyd, 1994; Boothroyd & Dewhurst, 1989; Boothroyd, Dewhurst, & Knight, 1994; Boothroyd, Poli, & Murch, 1982) will always modify any designs it encounters to allow their assembly operations to be carried out more easily and economically (the goal of DFA) according to a list of DFA guidelines. The focused concept of a design experience (represented by a design prototype) acts as an index to the design guidelines. During redesigning, a list of design prototypes is processed sequentially to emulate the serial processing of design guidelines.

Figure 10 illustrates a series of mechanical assemblies embedded within the design experiences processed by the design agent (Figure 11) to be used for illustrating the different behaviour of the memory system due to grounding.

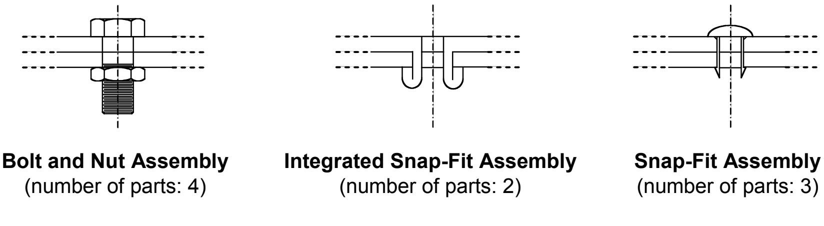

Figure 10. Mechanical assemblies to illustrate the behaviour of the memory system

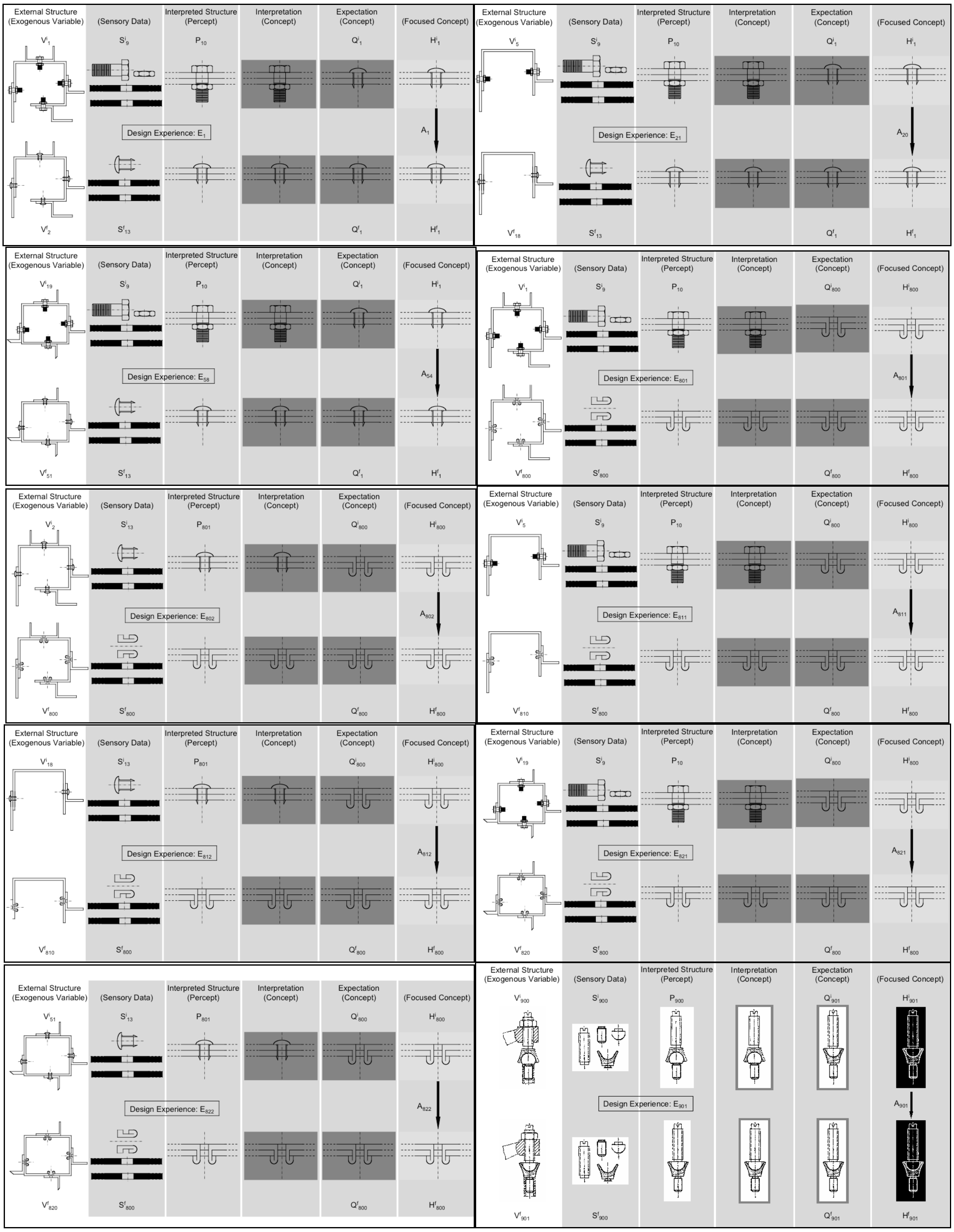

Figure 11. Initial design experiences processed by the agent

6.2 A Simplified Memory Construction Cycle

To demonstrate the system's behaviour for long-term learning, the following memory construction steps are used:

- Creation of partial design experience;
- Cueing of memory system;

- Cycling of network to get activations of pre-existing design experiences and knowledge;
- Selection of a candidate design experience;
- Construction of new design experience by completing the partial design experience; and
- Grounding of the activated experience or knowledge used.

A partial design experience is created according to the current situation (see Figure 3 for an example). The components of this partial experience consist of the current focused concepts (representing the design guideline to be used in the current redesign session) and the representation of the external structure (exogenous variable). These variables are embedded into the memory system and clamped before the system is cycled. A series of design experiences, modelled by the activated instance nodes, is activated when the system stabilises after the cycling process. Relevant candidate experiences are highlighted with positive activation values. A particular design experience is selected for the subsequent construction of memory by completing the partial design experience with its constituents. The activated experience that is used is grounded for subsequent use of the memory system.

A particular memory construction is labelled as P_x^y, which represents the *yth* cycle of a particular run, and the *xth* situation that pertains to the current memory construction cycle. A common value for *x* for different construction cycles implies that the partial design experiences used contain the same focused concepts. Grounding of the activated experience is done separately from the memory construction cycle and is labelled as G_z, where *z* represents the activated experience that was grounded.

To illustrate the use of the above notation, consider two memory construction processes P_1^y and P_2^y, and a grounding process based on the activated experience E_{200}. P_1^1 denotes the first run of a memory construction cycle based on a partial design experience containing information regarding the exogenous variable V^i_{200} and focused concepts H^i_{200} and H^f_{200}. P_1^2 denotes the second run of the memory construction cycle based on another partial design experience containing information regarding the exogenous variable V^i_{300} and the *same* focused concepts H^i_{200} and H^f_{200}. Note that both runs are based on the same focused concept but with different exogenous variables since the agent is looking at two different environments. This similarity (in terms of the same focused concepts) is captured by using the same notation (*x* in P_x^y).

$P_1^1 \rightarrow G_{E_{200}} \rightarrow P_2^1 \rightarrow G_{E_{200}} \rightarrow P_1^2$ denotes the following process:

- P_1^1 : cycle the memory system according to a partial design experience {V^i_{200}, H^i_{200} and H^f_{200}};
- $G_{E_{200}}$: ground experience E_{200};
- P_2^1 : cycle the memory system according to a partial design experience {V^i_{300}, H^i_{300} and H^f_{300}};
- $G_{E_{200}}$: ground experience E_{200} again; and
- P_1^2 : cycle the memory system according to the partial design experience {V^i_{400}, H^i_{200} and H^f_{200}}, where V^i_{400} is the new exogenous variable representing another new structure from the environment.

To give greater emphasis to the long-term learning process, it is assumed that:

- the design experiences contained within the memory system have structures with better performance than those from the environment;
- the activated design experience that is selected can be applied to the current situation; and
- only one activated experience is selected for used in one memory construction.

6.3 Common Response via Different Operating Modes

The first set of experiments demonstrates the different operating behaviours of the memory system due to the degree and type of grounding carried out by the system. The experiments outlined in this section produce the same response across common and different situations although the internal mechanism operates differently.

The response of the system is obtained by extracting the experience with maximum activation from all the activated design experiences during a single cycle of memory construction. Two cycles produce the same response when the same design experience is activated with the maximum values in two separate cycles of memory construction.

The following cases are considered:

- different operating modes across similar situations due to the grounding of different activated design experiences (Experiment 1 and Experiment 2); and
- different operating modes across different situations due to the grounding of different activated design experiences (Experiment 3).

For all the above cases, the following applies:

- cycle P_1 has a cue based on {H^i_1 and H^f_1};
- cycle P_2 has a cue based on {H^i_{901} and H^f_{901}};
- design experience E_{801} has the focused concepts {H^i_{800} and H^f_{800}};
- design experience E_{901} has the focused concepts {H^i_{901} and H^f_{901}}; and
- design experience E_1 has the focused concepts {H^i_1 and H^f_1}.

6.3.1 Different Operating Modes Across Similar Situations

The purpose of the first two experiments (Experiment 1 and Experiment 2) is to highlight the effect of different grounding on the operating mode of the constructive memory system across similar situations. The two experiments are performed to produce similar responses to illustrate the internal changes within the system before the threshold that produces a different response is reached.

The two experiments considered here, each with two runs of memory construction, are:

- Experiment 1:
  - Run 1: P_1^1;
  - Run 2: $P_1^1 \rightarrow G_{E_{801}} \rightarrow P_1^2$.
- Experiment 2:
  - Run 1: P_1^1;
  - Run 2: $P_1^1 \rightarrow G_{E_1} \rightarrow P_1^2$.

6.3.1.1 Experiment 1

Figure 12 illustrates the different states of the memory system based on the two runs for Experiment 1. For the first run, the memory system transits from Memory State 0 to Memory State 1 (P_1^1). For the second run, the memory system transits from Memory State 0 all the way to Memory State 4 ($P_1^1 \rightarrow G_{E_{801}} \rightarrow P_1^2$). The response from the system is the same (E_{801} having the maximum activation value) but the system operates differently for the two runs as indicated by the actual activation values of E_{801} (0.845 in Run 1 and 0.849 in Run 2). The increase in activation value is due to the grounding of experience E_{801} and thus its likelihood for subsequent use increases.

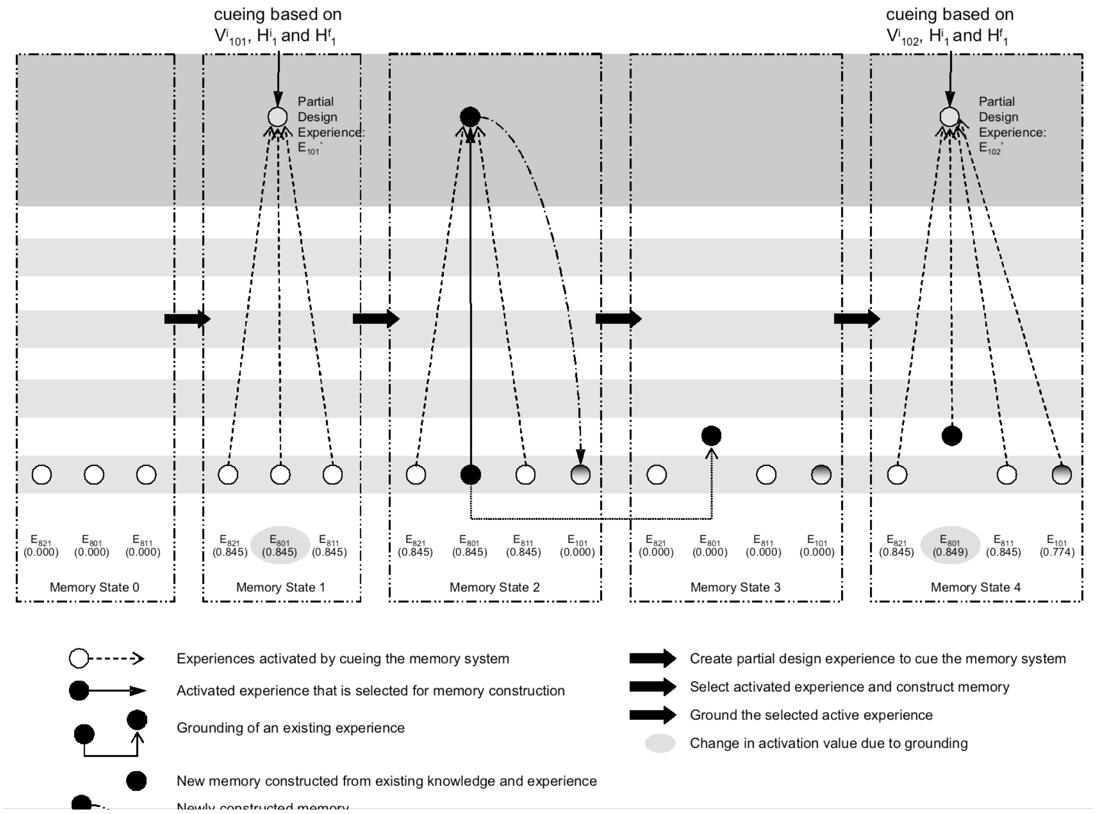

Figure 12. Different states of the memory system for Experiment 1

6.3.1.2 Discussion of Results: Experiment 1

The system commences at rest (Memory State 0 in Figure 12). A partial design experience E_{101}' is created with a variable defining the environment V^i_{101} and a focused concept of a snap-fit assembly represented by H^i_1 and H^f_1 that the agent tries to use in the current situation according to a specific design guideline. This partial experience is sent into the system as a cue (Memory State 1 in Figure 12) for constructing memories of associated experiences that use snap-fits in other situations.

The memory system responds with an experience E_{801} that proposes the use of an integrated snap-fit (H^i_{800} and H^f_{800}) instead of the separate snap-fit used to cue the system. This preference for integrated snap-fit is caused by previous constructive learning that modifies experiences that use separate snap-fits to use integrated snap-fits for better performance in DFA. Based on this activated design experience, the agent structures itself and the environment to look for the application of an integrated snap-fit into the external structure.

When the integrated snap-fit is adopted for the current situation, the original activated experience E_{801} (the response from the system) is grounded (Memory States 2 and 3 in Figure 12). This makes the likelihood of responding with the integrated snap-fit even more likely in subsequent memory construction cycles. This higher probability of usage is demonstrated when the system is cued with another partial design experience E_{102}' in another design session (Memory State 4 in Figure 12). The situation of this new session is similar to that of the initial run since it uses the same snap-fit (H^i_1 and H^f_1) for the partial design experience. The increase in activation of E_{801} (from 0.845 in Run 1 to 0.849 in Run 2) indicates that it has been given greater emphasis due to its previous usage.

The implication of this is that the long-term learning affects the operating modes of the memory system although the same response is obtained from different memory construction cycles. The difference in behaviour across similar situations gives the memory system the ability to operate in a different way even when it is cued with exactly the same cue in two consecutive memory construction cycles.

The increase in activation values of the grounded experience in subsequent operations of the memory system is also indicative of the possibility that the response of the system can be altered if sufficient grounding is provided. The activation value of any activated design experience can be increased through subsequent groundings to enable it to gain maximum activation and alter the response of the system in subsequent similar situations.

This means that it is possible to change the proposed use of an integrated snap-fit to that of a separate snap-fit (response of the system) if the experience containing the separate snap-fit has been given enough grounding. The next experiment illustrates the boundary condition just before this happens.

6.3.1.3 Experiment 2

Figure 13 illustrates the different states of the memory system based on the two runs for Experiment 2. For the first run, the memory system transits from Memory State 0 to Memory State 1 (P_1^1). For the second run, the memory system transits from Memory State 0 all the way to Memory State 4 ($P_1^1 \rightarrow G_{E_1} \rightarrow P_1^2$). This time the grounded experience is not based on the maximum activation value. Design experience E_1 (with activation value 0.839) is grounded instead of the experiences E_{801} or E_{811} or E_{821} (with a maximum activation value of 0.845). The response from the system is still the same (E_{801}, E_{811} and E_{821} still have the same maximum activation value) but the system operates differently for the two runs as indicated by the decrease in activation value of E_{801} (0.845 in Run 1 and 0.843 in Run 2). This decrease in activation value of E_{801} is due to the grounding of experience E_1 that increases E_1's likelihood of usage in subsequent operations of the memory system (0.839 in Run 1 to 0.843 in Run 2).

6.3.1.4 Discussion of Results: Experiment 2

The system commences at rest (Memory State 0 in Figure 13). A partial design experience E_{103}' is created with a variable defining the environment, V^i_{103} and a focused concept of a snap-fit represented by H^i_1 and H^f_1 that the agent tries to use in the current situation. This partial experience is used as a cue (Memory State 1 in Figure 13) for constructing memories of experiences that used the snap-fit in other situations.

The memory system responds with an experience E_{801} that proposes the use of an integrated snap-fit (H^i_{800} and H^f_{800}) instead of the separate snap-fit used to cue the system (Memory State 1 in Figure 13). The proposed integrated snap-fit is not used this time. The system is forced to use the original separate snap-fit in the current situation. This may be due to a change of conditions that renders the production cost of an integrated snap-fit to be higher and thus a separate snap-fit is preferred.

By using the experience E_1 that uses the separate snap-fit instead of that (E_{801}) proposed by the system, E_1 is grounded (Memory State 2 and 3 in Figure 13). When the system encounters another similar situation (Memory State 4 in Figure 13), the likelihood of using the separate snap-fit increases as indicated by the increased activation value (from 0.839 in Run 1 to 0.843 in Run 2).

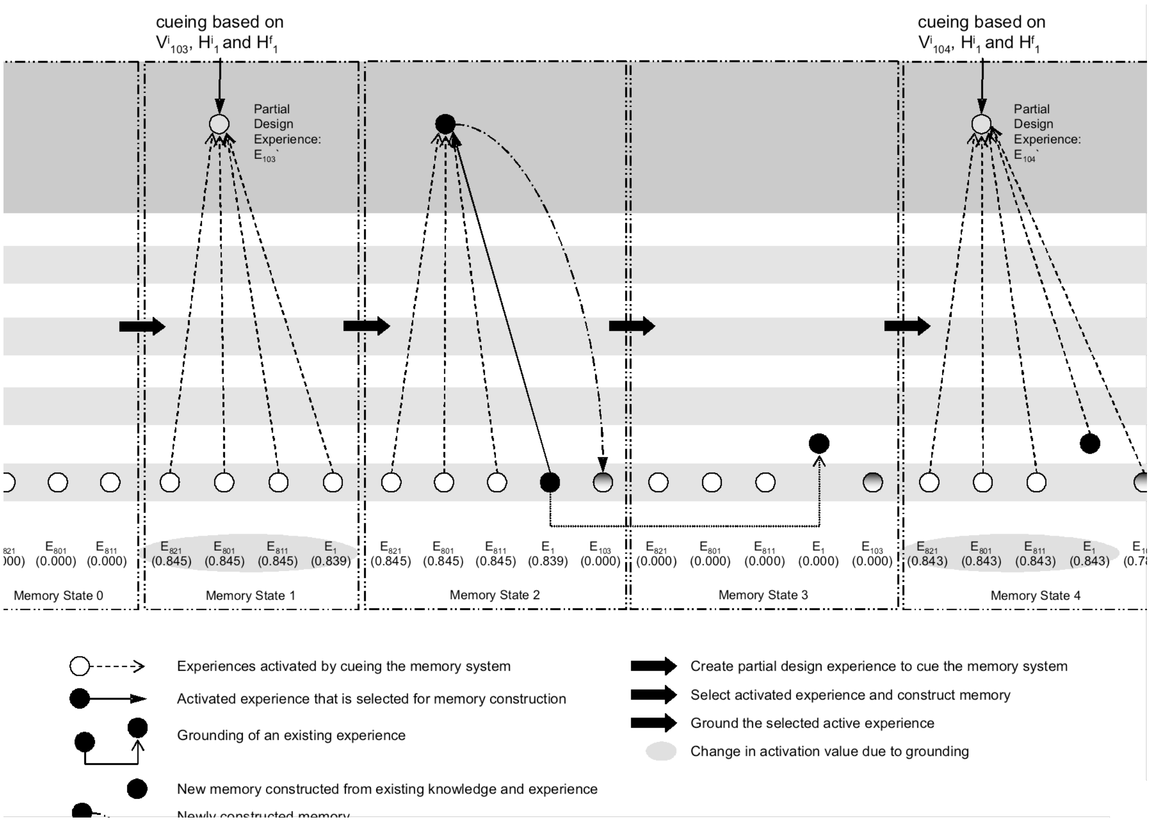

Figure 13. Different states of the memory system for Experiment 2

Although the system still responds with the suggestion of using the integrated snap-fit, this time it is given less emphasis as indicated by the decrease of activation of E_{801}, E_{821} and E_{811} (from 0.845 in Run 1 to 0.843 in Run 2). When the final activation of this second run is compared with the final activation of E_{801} in the second run of Experiment 1, the activation is also observed to decrease (from 0.849 in Experiment 1: Run 2 to 0.843 in Experiment 2: Run 2). This is indicative of the change within the memory system due to the learning of different experiences.

This behaviour of the system confirms the hypothesis stated in Experiment 1 that it is possible to alter the response of the system if sufficient grounding is provided. The activation value of any activated design experience (E_1) can be increased through subsequent grounding to enable it to gain maximum activation and to alter the response of the system in similar situations subsequently. The actual alteration of the response is demonstrated in Experiments 4 and 5.

6.3.2 Different Operating Modes Across Different Situations

The purpose of the following experiment is to highlight the effect of long-term learning on the operating mode of the constructive memory system in different situations through the grounding of different design experiences. The emphasis is on showing the different operating modes of the memory system caused by learning different experiences in different situations. The experiment is performed without crossing the threshold of response change, to produce similar responses in order to illustrate the internal changes within the system.

One experiment with two runs of memory construction is considered here:

- Experiment 3:
  - Run 1: $P_1^1 \to G_{E_{801}} \to P_1^2 \to G_{E_{801}} \to P_1^3$;
  - Run 2: $P_1^1 \to G_{E_{801}} \to P_2^1 \to G_{E_{901}} \to P_1^2$.

This experiment is expected to produce the same response (maximum activated experience remains the same), but the memory system operates differently in each run as the activation values of the highlighted experiences differ for each.

6.3.2.1 Experiment 3

Figure 14 illustrates the different states of the memory system based on the first run for Experiment 3. The memory system transits from Memory State 0 to Memory State 7 ($P_1^1 \to G_{E_{801}} \to P_1^2 \to G_{E_{801}} \to P_1^3$). Figures 15 and 16 illustrate the different states of the memory system based on the second run for Experiment 3. The memory system transits from Memory State 0 to Memory State 7 ($P_1^1 \to G_{E_{801}} \to P_2^1 \to G_{E_{901}} \to P_1^2$). The activation value of E_{801} has reduced from 0.853 ($P_1^1 \to G_{E_{801}} \to P_1^2 \to G_{E_{801}} \to P_1^3$) in the first run of Experiment 1 to 0.846 in the second run of Experiment 2 ($P_1^1 \to G_{E_{801}} \to P_2^1 \to G_{E_{901}} \to P_1^2$). This is due to the grounding of another totally unrelated design experience E_{901} in a different situation. As this new experience is grounded, all other experiences within the system are subjected to a decay process. The effect of the long-term learning that occurs beforehand ($G_{E_{801}}$) is reduced as new situations are encountered. The system's mode of operation changes to reflect the effect of "use it or lose it" as a result of grounding another experience in a different situation.

6.3.2.2 Discussion of Results: Experiment 3

This experiment shows that the increase of activation value due to grounding is not permanent. When an experience is left unused in subsequent memory construction, the effect of its previous grounding will eventually die off. To sustain a high activation value during memory construction, another mechanism is required. It will be shown later that this mechanism is based on the number of new experiences created according to the activated design experience used (grounded) during memory construction (Experiment 6).

The grounding of an activated design experience (long-term learning) produces two effects that influence its subsequent activation values. An immediate effect of grounding is to give more emphasis to the activated experience selected in the subsequent operation of the memory system in similar situations. This effect, however, is only temporary and the increased emphasis will eventually die off when the subsequent use of the experience is not maintained. The second effect of grounding is to produce a new experience that utilizes the activated experience that was selected and grounded. This effect permanently increases the emphasis of the activated experience in subsequent operations of the memory system in similar situations but its influence is much less and slower than that produced by the first case.

Within the domain of DFA this means that emphasis on the use of the separate snap-fit based on grounding alone (Experiment 2) is not sufficient to sustain its maximum activation in subsequent cueing. Only through the increase in the number of new design experiences created with more use of separate snap-fits in subsequent situations will the emphasis on the use of the separate snap-fit be permanent. However, the effect of these new experiences cannot be seen immediately until a sufficient number of them have been created.

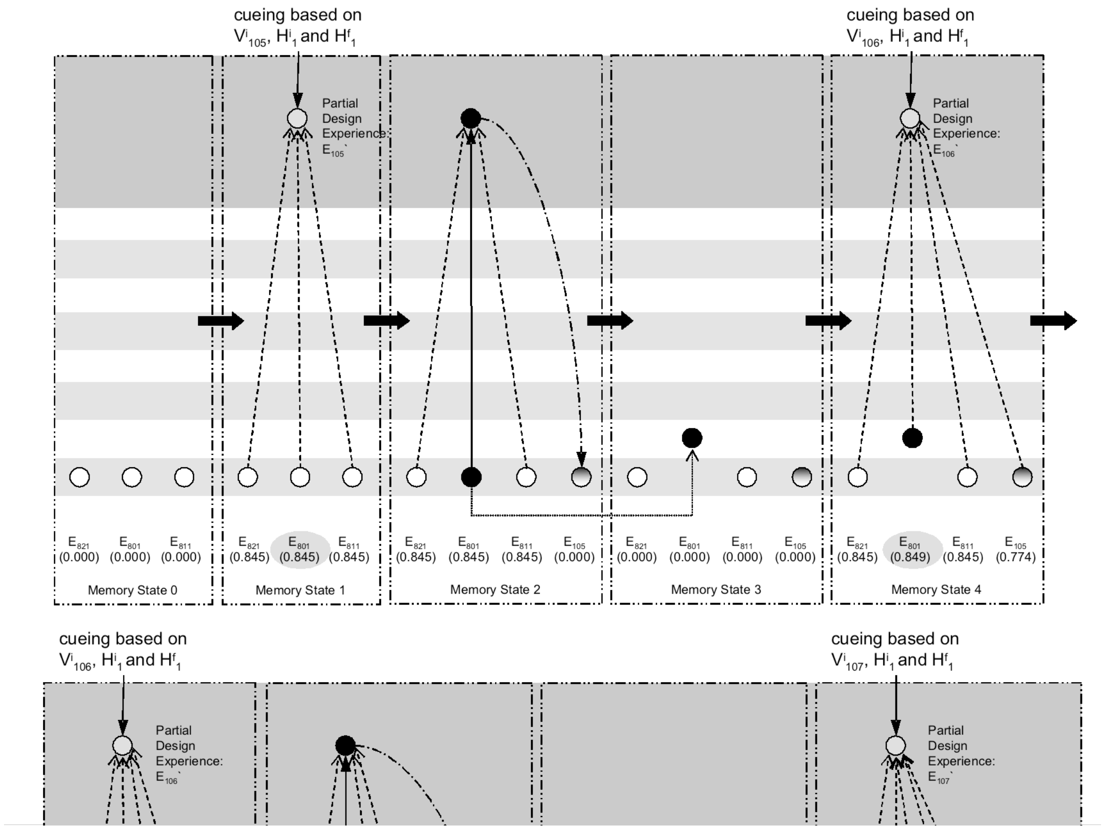

Figure 14. Different states of the memory system for Experiment 3: Run 1 (Memory State 4 is repeated in the bottom row to show the continuation from the top)

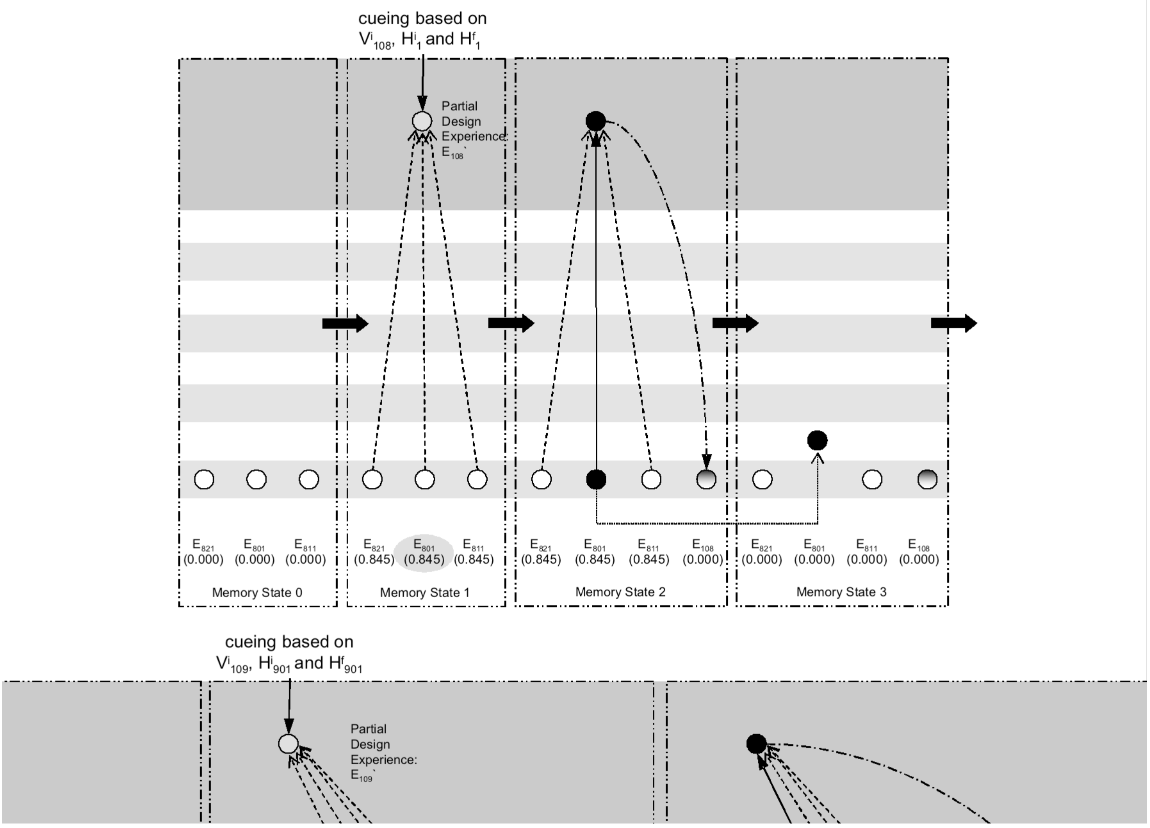

Figure 15. Memory State 0 to Memory State 5 of the system for Experiment 3: Run 2 (Memory State 3 is repeated in the bottom row to show the continuation from the top. Continued in Figure 16)

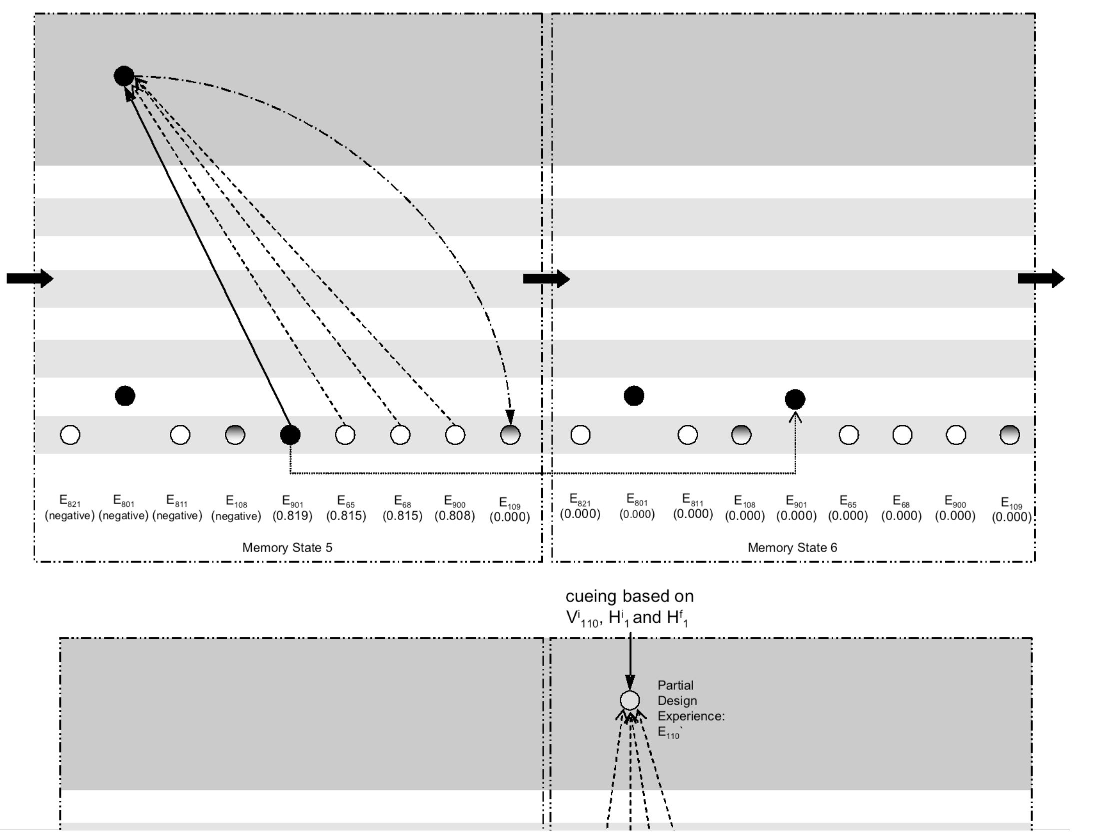

Figure 16. Memory State 5 to Memory State 7 of the system for Experiment 3: Run 2 (Memory State 5 is repeated in the top row to show the continuation from the previous diagram, Figure 15, and Memory State 6 is repeated in the bottom row to show the continuation from the top row of the current figure)

6.4 Different Responses via Different Operating Modes

The second set of experiments performed on the implemented constructive memory system demonstrates the different output produced by the system due to the amount of grounding. The experiments outlined in this section produce different responses across similar and different situations due to long-term learning. The notion of similarity is based on the use of the same focused concepts within the cue to the system. The response of the system is obtained by extracting the activated experience with the maximum activation during a single cycle of memory construction.

The difference in maximum activation values between each run for the same activated design experience is indicative of the different operating modes of the system. However, in this section, not only is the maximum activation value of the activated experience (operating mode of the system) made to change, the response (that particular activated design experience with maximum activation) is also altered between different construction cycles of the memory system through grounding. For the experiment described here, a common activated design experience is emphasized repeatedly to produce the different response required.

The following cases are considered:

- different response due to repeated grounding of an activated design experience for similar situations (Experiments 4); and
- different response due to repeated grounding of an activated design experience for different situations (Experiment 5).

For each of the above cases, the following applies:

- cycle P_1 has cue based on {H^i_1 and H^f_1};
- cycle P_2 cue based on {H^i_{800} and H^f_{800}};
- design experience E_{801}, has the focused concepts {H^i_{800} and H^f_{800}};
- design experience E_1, Figure 11 has the focused concepts {H^i_1 and H^f_1}.

6.4.1 Different Responses in Similar Situations

The purpose of the experiment in this section is to demonstrate that with enough grounding, the memory system can be forced to respond with another activated design experience when the threshold for response change has been reached in similar situations. This behaviour is important to situate the operations of the memory system according to the current interactions with the environment.

One experiment with two runs of memory construction is considered:

- Experiment 4:
  - Run 1: P_1^1;
  - Run 2: $P_1^1 \rightarrow G_{E_1} \rightarrow P_1^2 \rightarrow G_{E_1} \rightarrow P_1^3$.

The system is expected to respond differently after the experience E_1 has been grounded twice repeatedly (the threshold for response change in this case). The original response E_{801} is suppressed by E_1.

6.4.1.1 Experiment 4

Figures 17 and 18 illustrate the different states of the memory system based on the two runs for Experiment 4. For the first run, the memory system transits from Memory State 0 to Memory State 1 (P_1^1). For the second run, the memory system transits from Memory State 0 all the way to Memory State 7 ($P_1^1 \rightarrow G_{E_1} \rightarrow P_1^2 \rightarrow G_{E_1} \rightarrow P_1^3$). The response from the system has changed from E_{801} to E_1. The activation value of E_{801} has decreased from 0.845 to 0.843 and finally to 0.841, while the activation value of E_1 has increased from 0.839 to 0.843 and finally to 0.846. The system's response has effectively changed from suggesting an integrated snap-fit to suggesting a separate snap-fit assembly for redesign.

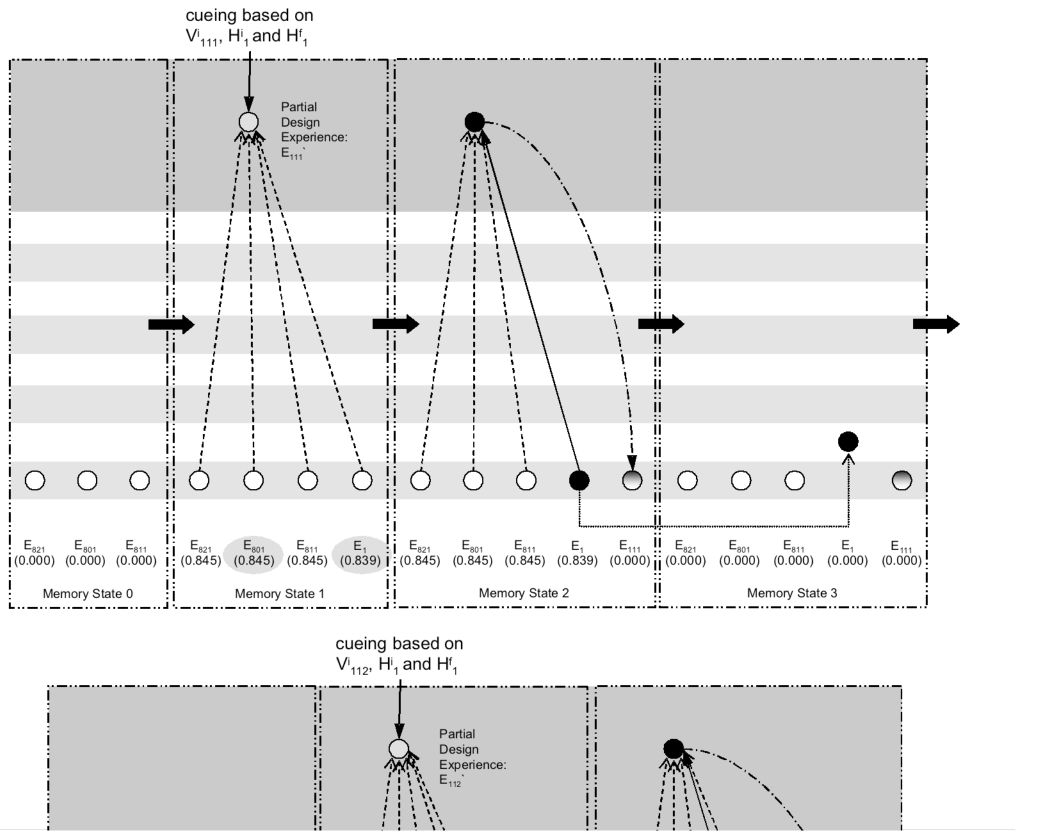

Figure 17. Memory State 1 to Memory State 5 of the system for Experiment 4 (Memory State 3 is repeated in the bottom row to show the continuation from the top row. Figure continued in Figure 18)

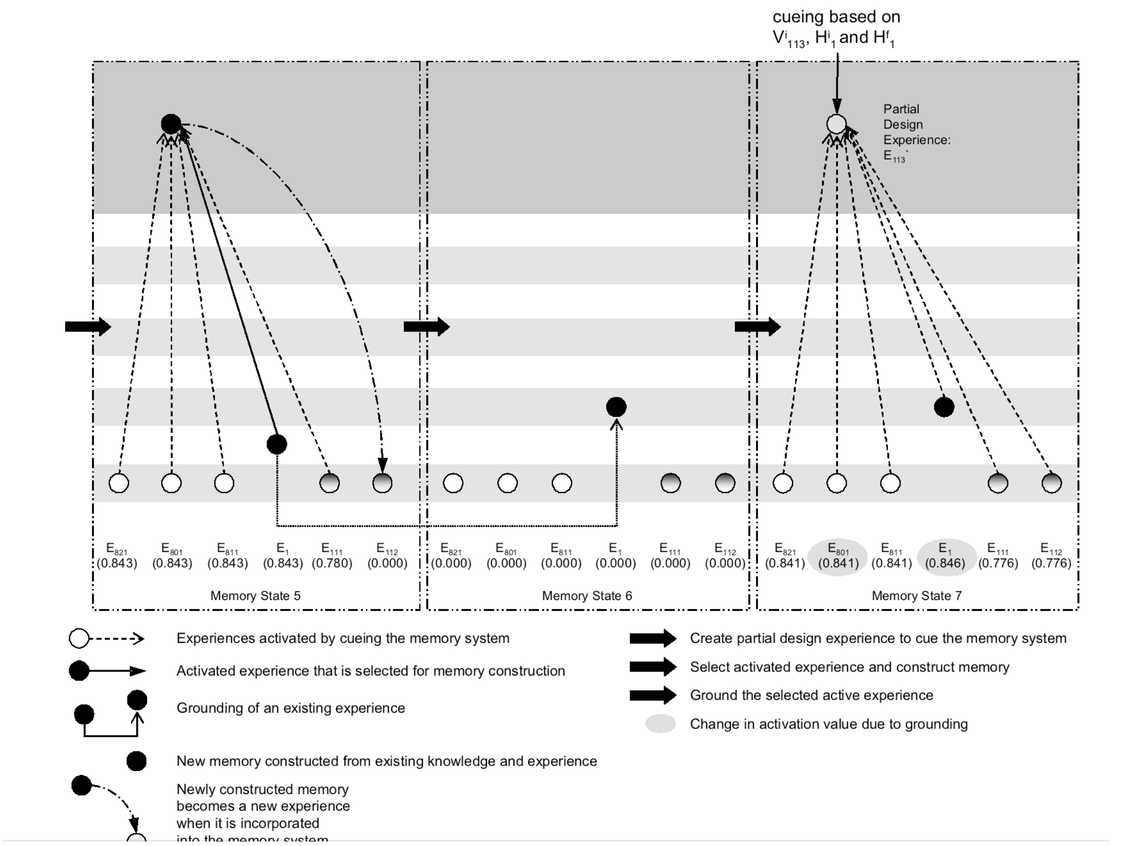

Figure 18. Memory State 5 to Memory State 7 of the system for Experiment 4 (Memory State 5 is repeated to show the continuation from the previous diagram, Figure 17)

6.4.1.2 Discussion of Results: Experiment 4

This experiment demonstrates that the system can be made to respond differently according what is used frequently. Initially the cue with the separate snap-fit as the focused concept (Memory State 0 in Figure 17) triggers the integrated snap-fit as the focused concept (Memory State 1 in Figure 17).

However, as the separate snap-fit is repeatedly used, the design experience E_1 eventually gets enough grounding to change the response of the system to E_1 that uses a separate snap-fit as the focused concept (Memory State 7 in Figure 18). The grounding of the design experience E_1 has effectively altered the suggested use of an integrated snap-fit (H^i_{800} and H^f_{800} in E_{801}) to a separate snap-fit (H^i_{800} and H^f_{800} in E_1) as proposed initially by the system for a cue based on the use of a separate snap-fit.

This behaviour implies that even though the memory system may have learned that a focused concept such as an integrated snap-fit (H^i_{800} and H^f_{800} in E_{801}) is useful in previous situations, the use of this learned concept is not absolute as it is subjected to the current interactions between the agent and the environment. The application of knowledge (focused concepts) is situated. Situations may exist that render the use of the integrated snap-fit infeasible and these situations will cause the system to respond differently in other similar situations. The system updates what it had learned previously according the most recent interactions with the environment. This is in contrast to the learning paradigm in normal machine learning where learning occurs as a process separate from interactions with the environment.

6.4.2 Different Responses in Different Situations

The purpose of the experiment in this section is to demonstrate that with enough grounding, the memory system can be forced to respond with another activated design experience when the threshold for response change has been reached in different situations. This behaviour is important to situate the operations of the memory system according to the current interactions with the environment. In contrast to having different responses in similar situations (described in section 6.4.1), the situation at the final memory construction stage is altered to highlight the influence of grounding in settings that would normally cause the system to behave differently without grounding.

One experiment with two runs of memory construction is considered:

- Experiment 5:
  - Run 1: $P_1^1 \rightarrow G_{E_1} \rightarrow P_2^1$;
  - Run 2: $P_1^1 \rightarrow G_{E_1} \rightarrow P_1^2 \rightarrow G_{E_1} \rightarrow P_1^3 \rightarrow G_{E_1} \rightarrow P_2^1$.

The system is expected to change its response in light of learning through the interactions with the environment. This behaviour is observable even if the memory system is subsequently cued with a focused concept that was useful in the past but is less useful in the current situation.

6.4.2.1 Experiment 5

Figure 19 illustrates the different states of the memory system based on the first run for Experiment 5. After the system has gone through one cycle of P_1, based on the cue containing H^i_1 and H^f_1 (separate snap-fit assembly), design experience E_1 is grounded instead of E_{801} (the activated experience with the maximum activation). This grounding is attempting to change the subsequent response of the memory system from proposing an integrated snap-fit assembly (H^i_{800} and H^f_{800} contained within E_{801}) to proposing a separate snap-fit assembly (H^i_1 and H^f_1 contained within E_1). However, the effect of this grounding is not strong enough to suggest a separate snap-fit assembly, via making the design experience E_1 the maximum activated experience, when the system is cued with (H^i_{800} and H^f_{800}). The system responds with E_{801} as the maximum activated experience with a value of 0.847 over with E_1 with a value of 0.840.

Figures 20 and 21 illustrate the different states of the memory system based on the second run for Experiment 5. The system undergoes three cycles of P_1 and three consecutive groundings of design experience E_1. This grounding changes the subsequent response of the memory system from proposing an integrated snap-fit assembly (H^i_{800} and H^f_{800} contained within E_{801}) to proposing a separate snap-fit assembly (H^i_1 and H^f_1 contained within E_1). The system responds with E_1 as the maximum activated experience with a value of 0.847 over E_{801} with a value of 0.843 at the end of Run 2 for this experiment.

6.4.2.2 Discussion of Results: Experiment 5

The activated design experience E_1 is grounded and used repeatedly to complete the partial design experience used as a cue to the system in this experiment. These groundings change the response of the system by changing the maximum activated experience from E_{801} to E_1. This has the effect of forcing the use of a separate snap-fit assembly over the use of an integrated snap-fit assembly as proposed initially by the system, even when the cue is based on the use of an integrated snap-fit.

This behaviour highlights the strong influence of grounding on the response of the system. Without grounding, the cue based on the integrated snap-fit (Memory State 10, Figure 21) would normally produce a response that uses an integrated snap-fit, E_{801} (Memory State 4, Figure 19). With the repeated grounding of E_1, the response from the system is altered to one that proposes the use of the separate snap-fit.

This behaviour implies that the use of design experience and knowledge is situated and highly dependent on what has been learned recently through the interactions between the agent and the environment. What was termed useful in the past is not absolute but varies according to current interactions with the environment.

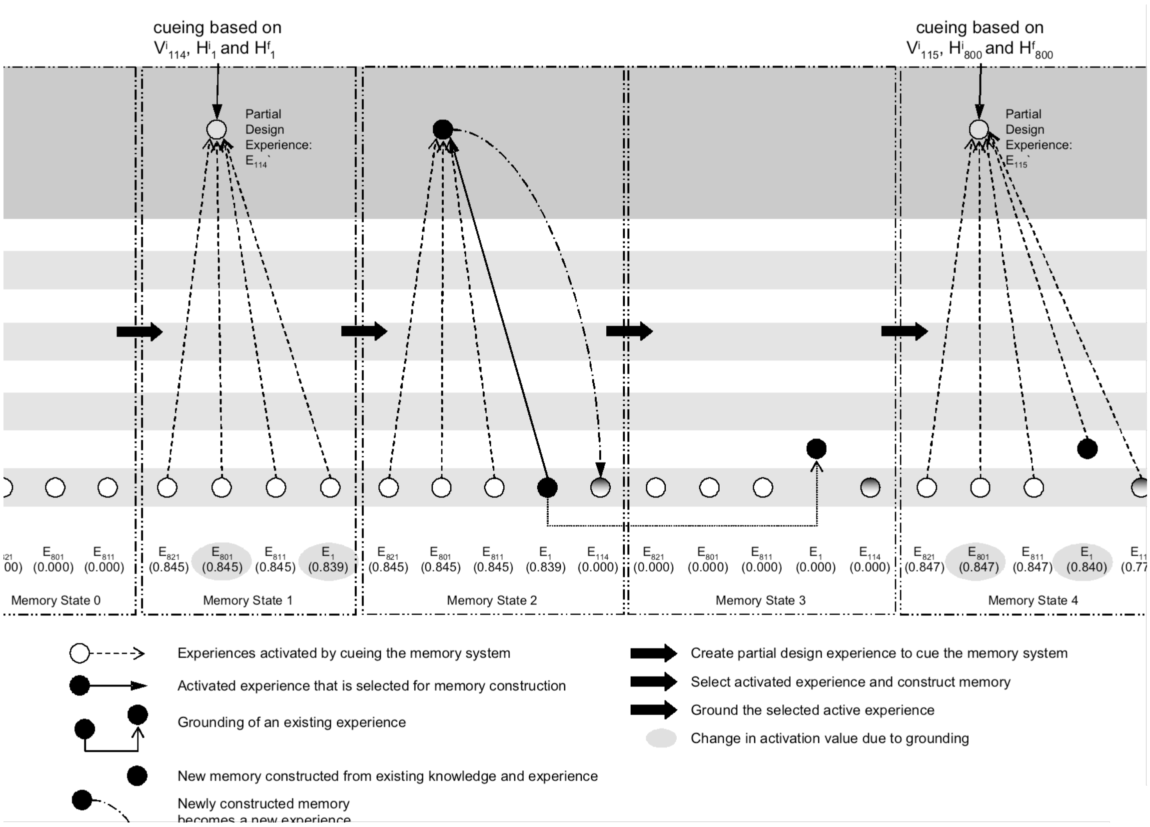

Figure 19. Different states of the memory system for Experiment 5: Run 1

6.5 Mechanisms for Sustaining High Activation Values

Grounding has two effects on the subsequent operations of the memory system. The immediate effect is to give a greater emphasis to the grounded design experience that was used recently so that its likelihood of usage in subsequent memory construction cycles is elevated. This effect of grounding is achieved by increasing the interconnection strengths (weights) of the grounded experience within the neural network implementation of the constructive memory system. However, elevated emphasis is only temporary as the weights are subjected to a decay process once other design experiences are grounded.

Another effect of grounding that increases the likelihood of usage of an experience is caused by the increase in the number of experiences that share common focused concepts. For example, the grounding of the experience E_1 (containing H^i_1 and H^f_1 as focused concepts) causes a new experience with the same focused concepts, H^i_1 and H^f_1, to be created within the memory system. This new experience contributes to the activation of the experience E_1 via these common focused concepts in subsequent memory construction cycles. Compared to the first effect of grounding, this indirect contribution to increasing the likelihood of usage is permanent and does not decay with the grounding of other experiences. However, this effect is less immediate, and behavioural changes in the memory system that result from by this effect are evident only after a sufficient number of new experiences have been created.

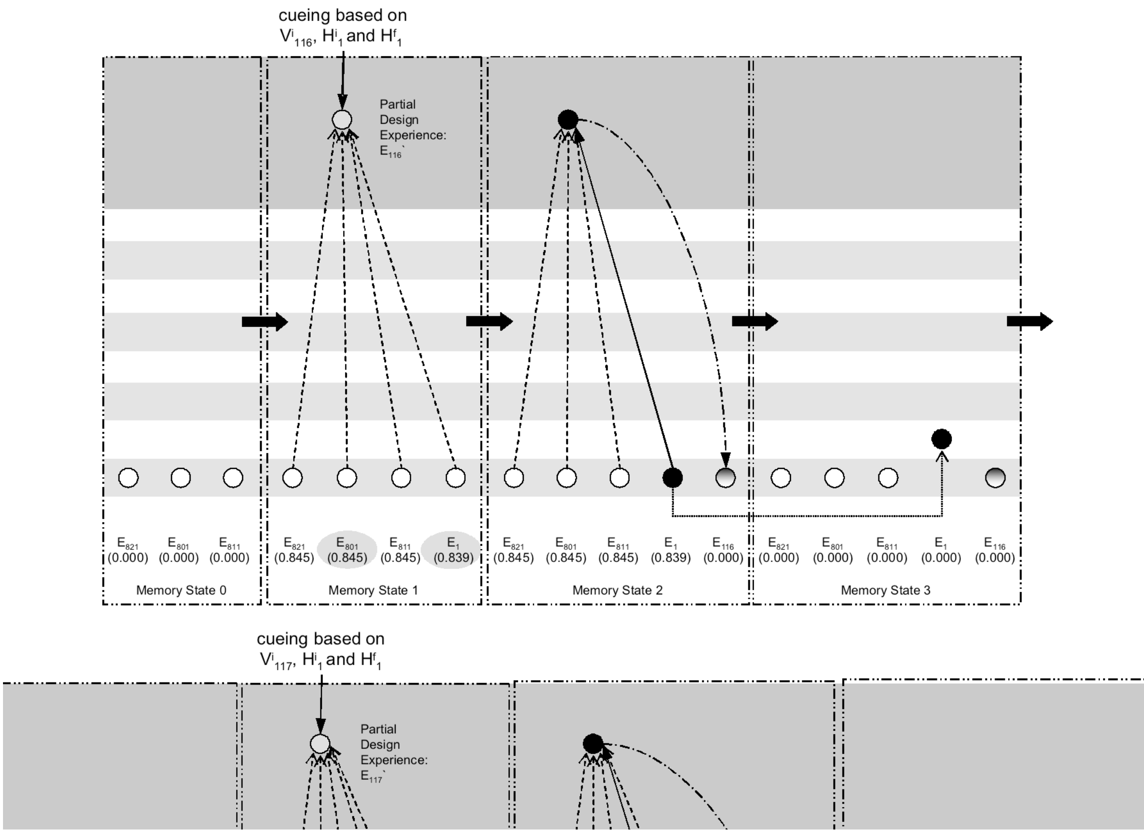

Figure 20. Memory State 0 to Memory State 6 of the system for Experiment 5: Run 2 (Memory State 3 is repeated in the bottom row to show the continuation from the top row of the current figure. Continued in Figure 21)

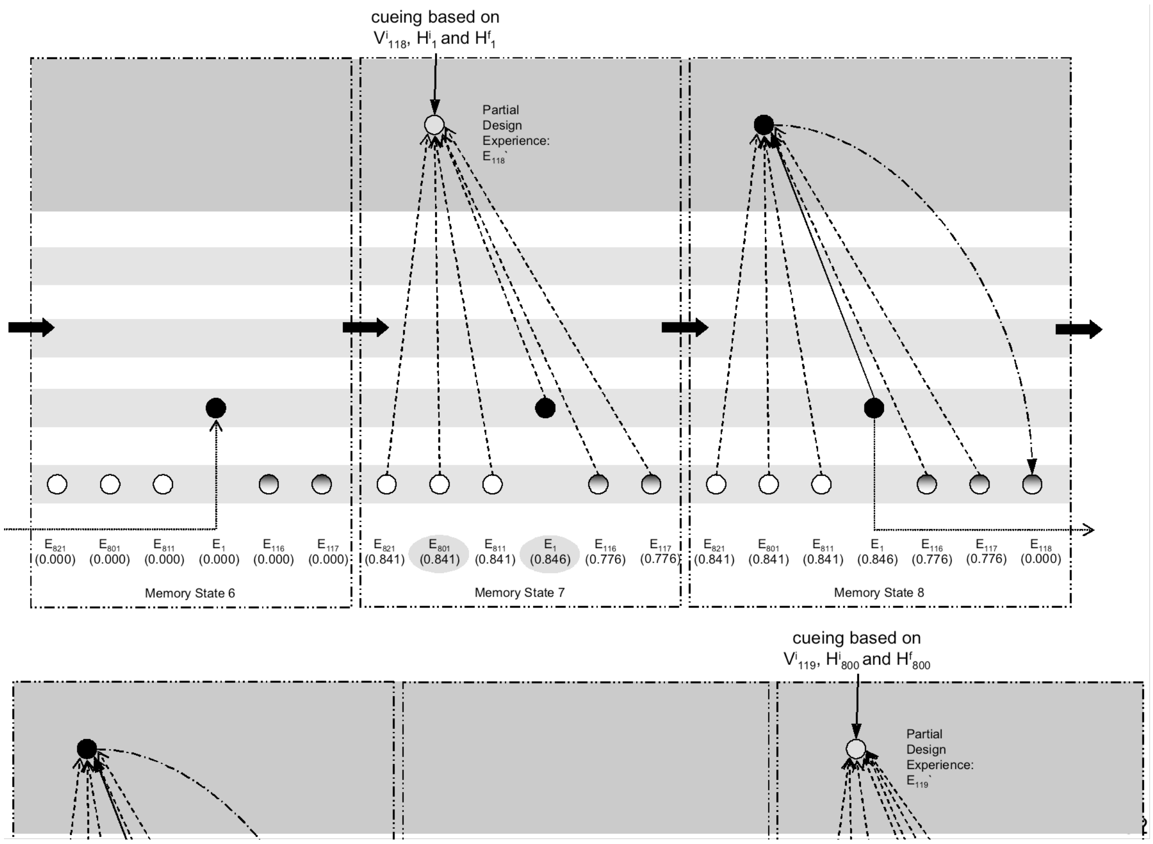

Figure 21. Memory State 6 to Memory State 10 of the system for Experiment 5: Run 2 (Memory State 6 is repeated in the top row to show the continuation from the previous diagram, Figure 20, and Memory State 8 is repeated in the bottom row to show the continuation from the top row of the current figure

6.5.1 Likelihood of Usage

The notion of the likelihood of usage needs to be quantified if the two effects of grounding are to be illustrated through an experiment with the constructive memory system. This likelihood of usage for an activated design experience can be obtained by comparing the actual activation of that experience against all other activated experiences that result from cueing the memory system:

$$Likelihood\ of\ Usage = \frac{activation_{actual} - activation_{\min}}{activation_{\max} - activation_{\min}}$$

where:

$activation_{actual}$: the actual activation value of the design experience of concern;

$activation_{max}$: the maximum activation value extracted from all the activated experiences; and

$activation_{min}$: the minimum activation value extracted from all the activated experiences.

A value of 1 indicates a maximum likelihood of usage in the current memory construction cycle. The likelihood of usage for the most grounded experiences is shown as a spike with value 1 at the time when grounding occurs within a plot of likelihood of usage and time. As other design experiences are grounded, this elevated likelihood of usage decreases in value to indicate the decreasing effect of grounding. The second effect of grounding is shown as a permanently elevated likelihood of usage after the spike in the same plot. This effect is permanent and maintains a constant likelihood of usage regardless of the grounding of other design experiences.

6.5.1.1 Experiment 6

The purpose of this experiment is to illustrate the two effects of grounding and show their temporal characteristics as other experiences are grounded. The experiment is carried out via the following steps:

- Ground design experience E_1 X number of times;
  - Create X number of new design experiences, according to the focused concepts H^i_1 and H^f_1 contained in E_1;
  - Ground with another design experience, E_{901}, 10 times;
  - For each grounding of E_{901}:
    - Cue system with partial design experience V^i_{600} with focused concepts H^i_1 and H^f_1;
    - Compute likelihood of activation for E_1;
- Repeat cycle by increasing the value of X from 1 to 9.

6.5.1.2 Discussion of Results: Experiment 6

Figure 22 illustrates a plot of the likelihood of usage against time for this experiment. In each cycle, at time t, the system is grounded with design experience E_1 for a specific number of times and then another experience E_{901} is grounded 10 times at 10 time intervals to illustrate the wearing off of the immediate effect of grounding. The constant elevated likelihood of usage after time t is sustained by the new experiences that were created with the same focused concepts as that of E_1. The number of times E_1 is grounded varies from 1 to 9 times for each plot to show the immediate and long-term effects of grounding.

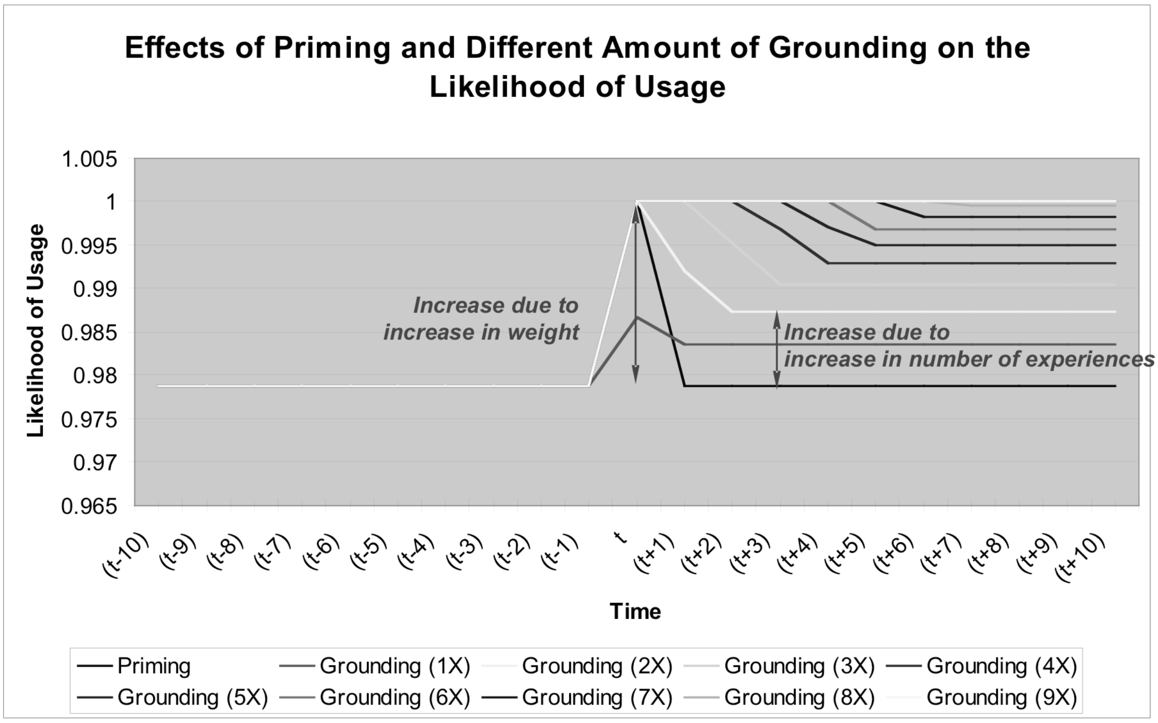

Figure 22. Variation in the likelihood of activation due to priming, grounding and number of experiences in the memory system (increases in likelihood due to increased weight and due to the increase in the number of experiences are compared for Grounding (2X))

It can be observed from Figure 22 that as the number of groundings for experience E_1 increases, the indirect effect of grounding effected through increasing the number of experience that share the same focused concepts provides the necessary mechanisms for sustaining the elevated likelihood of usage for the experience E_1. In contrast, the immediate effect of grounding causes a spike in the plot to elevate the likelihood of usage at the time of grounding. This effect dies off quickly as another experience is grounded. In the extreme case, the elevated likelihood of usage of a grounded experience behaviour as though the system has been primed with that experience before the network is cycled. The two effects of grounding capture the important idea that long-term learning through interactions with the environment (through grounding of activated design experience) alters what the design agent was given initially when it was first spawned. This alteration is temporary until a substantial number of interactions provide a stronger basis for the alteration. This form of learning is different from that of constructive learning where the effect of learning is permanent, immediate and based purely on the creation of new experiences to provide new behaviours for the memory system.

6.6 Summary of Results

From the experiments described so far, it can be seen that different types and amounts of grounding can affect the behaviour of the system across the same and different situations through the gradual suppression of experiences according to the current interactions with the environment. This is indicative of the situatedness of the memory system, as its constructive operations are determined by its current interpretations of past experiences based on what it had learned so far. To sustain high activation values across different memory construction cycles, new experiences are required on top of the elevated weights across the constituents of a design experience.

7.0 Comparison with Case-Based Design Systems

The constructive memory system described in this paper has many operations that are similar to other experiential-based design system. A comparison is made in the following between a constructive memory system for design and a case-based design system.

Case-based design is a form of analogical reasoning that uses previous design episodes to generate new designs (Maher, Balachandran, & Zhang, 1995). The episodes are usually represented as a repository of design cases. These cases are recalled during designing to provide the bases upon which solutions are created. The fundamental processes that are involved in using Case-Based Reasoning (CBR) for designing are:

- case recall where relevant cases are retrieved from the case repository; and
- case adaptation where the selected case is modified to produce a solution according to the current situation.

The operations of case adaptation are very similar to those of memory construction described in this paper. Both process uses an analogical process that matches and subsequently maps structures from the environment to those contained within the system and both learn through the application of old experiences in new situations.

There are, however, fundamental differences between the two approaches to designing. Case-based design is based on pattern matching between specific features from the environment and indices contained within the design cases (Maher et al., 1995). An index (or a set of indices) used to retrieve a case is not modified as it is used during the retrieval process. The retrieved case will definitely contain the index (or a subset of indices). A constructed memory, on the other hand, may or may not contain the same information used initially to cue the system. What gets constructed is not just a function of the cue to the system but also depends on what the system has learned to be associated with the original contents of the cue to propose alternative experiences as a basis for interpreting the environment.

Although a case-based design system and a constructive memory system may employ the same analogical techniques of matching and mapping, the frameworks within which they operate are fundamentally different. The constructive memory proposed in this paper emphasizes the situatedness of designing. The notion of situatedness contains the two fundamental ideas of *interaction* and *construction*. Interaction implies that what the system knows is not encoded a priori and indexed for use later, but rather the knowledge is developed through interaction with the environment. This interaction interprets the experiences processed by the system according to the current situation in such a way that the interpretations of past experiences are done through a “filter” defined by the present situation. Compared to a situated computing system, systems based on CBR lack this interaction (Clancey, 1997).

An idea closely related to interaction is the concept of memory construction about previous experiences. This construction process provides the basis for interpreting past experiences according to the current situation. A past experience is not “copied” into the present but rather it is interpreted based on the current interactions with the environment. What is interpreted eventually becomes part of the memory system (when the new memory is incorporated into the system as a new experience) that influences its subsequent interpretations.

To illustrate the effects of the interaction and construction not present in design systems based on CBR, consider the case where a case-based design system has adopted one of its cases for the current situation. During the recall of this case, a cue based on available information from the environment may be used to retrieve the closest match from the case repository. After the case has been adopted and incorporated into the memory system, if the same cue is presented to the system again, the originally retrieved case is activated with the same ranking. The newly adopted case is also activated with a better ranking, since it matches the situation much more closely because of the same cue. The lack of change in the ranking of the original case is indicative of the fact that the case-based design system does not change its operating mode through case adaptation. The cases processed by the system do not become part of the situation within which it operates.

In contrast, the design experiences (the equivalent of design cases in case-based design systems) processed by the constructive memory system readily become part of the situation within which they will operate in subsequent operations of the system. This is indicated by the feedback arrow originating from the EXPERIENCES block to the SITUATION block in Figure 1. If the same cue is presented to the system again, the system has the potential to operate and respond differently, depending on how big a change in the experiences it processes has taken place. These differences are caused by the interactions between the system and the environment through the grounding (long-term learning) and reinterpretation (constructive learning) of pre-existing experiences. A distinctive feature of the constructive memory system proposed in this paper is the way it allows different operating modes and responses to be produced according the same cue presented to the system. This is caused by varying the ways in which the same cue is framed within a system according to the current situation.

To summarize, the main difference between the constructive memory system proposed in this paper and case-based design system lies in the fact that the systems are based on two fundamentally different ideas. A case-based design system operates according to pattern matching between the case indices and information from the environment. A constructive memory system operates according to its interactions with the environment. During these interactions, memories about past experiences are constructed and incorporated into the system as new experiences that influence its subsequent operations. This is in contrast to design systems based on CBR where cases are indexed, retrieved, used and passively adapted.

8.0 Conclusion

Memory is modelled as a dynamic process, based on the model of constructive memory described in this paper. It is not a static imprint to be stored in a specific location and retrieved for use later. The memory construction processes are not predefined but are influenced by the situatedness of designing. Memories that are constructed may not match the original experience exactly as it was first experienced, but change according to when, where and what the memory system is cued with. These behaviours of the memory system are dependent on the current external environment, the current internal state of the design agent and the interactions between the agent and the external environment. The work presented in this paper provides a perspective of constructive memory within the domain of designing that allows a computational memory system to be developed. A computational framework for designing based on the notion of situatedness in designing has been created to demonstrate the various behaviours of the memory system not easily producible by normal design case retrieval systems.

Such a memory system forms one basis of a situated design agent. Without the ability to be situated a design agent requires increasing amounts of knowledge that has be to either encoded or learned. However, learning knowledge without also learning the situation produces only static generalisations and does not address the issue of the impact of the environment.

Acknowledgements

This research has been supported by a University of Sydney Postgraduate Research Award. Computational resources have been provided by the Key Centre of Design Computing and Cognition.

References

Atkinson, R.C., & Shiffrin, R.M. (1968). Human memory: A proposed system and its control processes. *The Psychology of Learning and Motivation: Advances in Research and Theory* (Spence, K.W., Ed.), pp. 89-195. New York: Academic Press.

Baddeley, A.D. (1999). Memory. *The MIT Encyclopedia of the Cognitive Sciences* (Wilson, R.A. & Keil, F.C., Eds.), pp. 514-517. Cambridge, Mass.: MIT Press.

Boothroyd, G. (1994). Product design for manufacture and assembly. *Computer-Aided Design, 26*(7), 505-519.

Boothroyd, G., & Dewhurst, P. (1989). *Product design for assembly*. Wakefield, R.I.: Boothroyd Dewhurst, Inc.

Boothroyd, G., Dewhurst, P., & Knight, W. (1994). *Product design for manufacture and assembly*. New York: M. Dekker.

Boothroyd, G., Poli, C., & Murch, L.E. (1982). *Automatic assembly*. New York: M. Dekker.

Clancey, W.J. (1997). *Situated cognition: on human knowledge and computer representations*. Cambridge, U.K. ; New York, NY, USA: Cambridge University Press.

Craik, F.I.M., & Lockhart, R.S. (1972). Levels of processing: A framework for memory research. *Journal of Verbal Learning and Verbal Behavior, 11*, 671-684.

Davis, D.N. (2001). Control states and complete agent architectures. *International Journal of Computational Intelligence, 17*(4), 621-650.

Fiesler, E., & Beale, R. (Eds.). (1997). *Handbook of Neural Computation*. New York: Oxford University Press.

Gero, J.S. (1990). Design prototypes: a knowledge representation schema for design. *AI Magazine, 11*(4), 26-36.

Gero, J.S. (1998a). Conceptual designing as a sequence of situated acts. *Artificial Intelligence in Structural Engineering* (Smith, I., Ed.), pp. 165-177: Springer, Berlin.

Gero, J.S. (1998b). Design tools that learn. *Advances in Engineering Software, 29*(10), 755-761.

Gero, J.S. (1998c). Towards a model of designing which includes its situatedness. *Universal Design Theory* (Grabowski, H., Rude, S., & Green, G., Eds.), pp. 47-56. Aachen: Shaker Verlag.

Gero, J.S. (1999). Constructive memory in design thinking. *Design Thinking Research Symposium: Design Representation* (Goldschmidt, G. & Porter, W., Eds.), pp. I.29-35: MIT, Cambridge.

Gero, J.S., & Fujii, H. (2000). A computational framework for concept formation in a situated design agent. *Knowledge-Based Systems, 13*(6), 361-368.

Gero, J.S., & Kannengiesser, U. (2000). Towards a situated Function-Behaviour-Structure framework as the basis for a theory of designing. *Workshop on Development and Application of Design Theories in AI in Design Research, Artificial Intelligence in Design'00* (Smithers, T., Ed.), pp. gk:1-5. Worcester, MA.

Gero, J.S., & Kannengiesser, U. (2002). The situated function-behaviour-structure framework. *Artificial Intelligence in Design'02* (Gero, J.S., Ed.), pp. 89-104. Kluwer: Dordrecht.

Gero, J.S., & Rosenman, M.A. (1989). A conceptual framework for knowledge-based design research at Sydney University's Design Computing Unit. *Artificial Intelligence in Design* (Gero, J.S., Ed.), pp. 361-380. Southampton and Berlin: CMP/Springer-Verlag.

Gero, J.S., Tham, K.W., & Lee, H.S. (1992). Behaviour: A link between function and structure in design. *Intelligent Computer Aided Design* (Brown, D.C., Waldron, M.B., & Yoshikawa, H., Eds.), pp. 193-225. North-Holland, Amsterdam.

Goldschmidt, G. (1997). Capturing indeterminism: representation in the design problem space. *Design Studies, 18*, 441-455.

Haykin, S.S. (1998). *Neural Networks: A Comprehensive Foundation*. New Jersey: Prentice Hall.

Hoehfeld, M., & Fahlman, S.E. (1992). Learning with limited numerical precision using the cascade-correlation learning algorithm. *IEEE Transactions on Neural Networks, 3*(4), 602-611.

Logie, R.H. (1995). *Visuo-Spatial Working Memory*. Hove: Lawrence Erlbaum.

Maher, M.L., Balachandran, M.B., & Zhang, D.M. (1995). *Case-based reasoning in design*. Mahwah, N.J.: Lawrence Erlbaum Associates.

Maher, M.L., & Gero, J.S. (2002). *Agent models of 3D virtual worlds.* Proc. ACADIA 2002: Thresholds, 127-138, Pamona.

Mandic, D., & Chambers, J. (2001). *Recurrent neural networks for prediction: Learning algorithms, architectures and stability*. Singapore: John Wiley and Sons.

McClelland, J.L. (1981). Retrieving general and specific information from stored knowledge of specifics. *Proceedings of the Third Annual Meeting of the Cognitive Science Society*, pp. 170-172. Hillsdale, NJ: Erlbaum.

McClelland, J.L. (1988). *Explorations in parallel distributed processing: a handbook of models, programs, and exercises*. Cambridge, Mass.: MIT Press.

McClelland, J.L. (1995). Constructive memory and memory distortion: a parallel-distributed processing approach. *Memory Distortion: How Minds, Brains, and Societies Reconstruct the Past* (Schacter, D.L., Ed.), pp. 69-90. Cambridge, Massachusetts: Harvard University Press.

Medler, D.A. (1998). A brief history of connectionism. *Neural Computing Surveys, 1*(1), 61-101.

Nehaniv, C., & Dautenhahn, K. (1998). Embodiment and memories - Algebras of time and history for autobiographic agents. *Cybernetics and Systems '98, Proceedings of the 14th European Meeting on Cybernetics and Systems Research Symposium on Embodied Cognition and Artificial Intelligence (Vienna, Austria, 14-17 April 1998). Austrian Society for Cybernetic Studies, volume 2* (Trappl, R., Ed.), pp. 651-656.

Nehaniv, C.L. (1999). Meaning for observers and agents. *IEEE International Symposium on Intelligent Control / Intelligent Systems and Semiotics, ISIC/ISAS'99 - September 15-17, 1999 Cambridge, Massachusetts, USA*, 435-440.

Qian, L. (1994). *Creative Design by Analogy.* Unpublished PhD Thesis, University of Sydney, Sydney.

Reed, R. (1993). Pruning algorithms: A survey. *IEEE Transactions on Neural Networks, 4*, 740-747.

Rosenman, M.A., & Gero, J.S. (1993). Creativity in design using a design prototype approach. *Modeling Creativity and Knowledge-Based Creative Design* (Gero, J.S. & Maher, M.L., Eds.), pp. 119-148. Hillsdale, New Jersey: Lawrence Erlbaum.

Russell, S., & Norvig, P. (1995). *Artificial Intelligence*. New Jersey: Prentice-Hall.

Sloman, A. (1996). *What sort of architecture is required for a human-like agent?*: Invited talk at Cognitive Modeling Workshop, AAAI96, Portland Oregon, Aug 1996.

Ziemke, T. (1999). Rethinking grounding. *Understanding representation in the cognitive sciences* (Riegler, A., Peschl, M., & Stein, A., Eds.), pp. 177-190. New York: Plenum Publishers.

This is a copy of the paper: Liew, P and Gero JS (2004) Constructive memory for situated agents, *AIEDAM* (to appear)