DARWIN 270138

Dexterous Assembler Robot
Working with
Embodied Intelligence

SEVENTH FRAMEWORK PROGRAMME
ICT Priority

Deliverable D4.4: Development of a general framework for goal directed reasoning

Due date of deliverable: 19.12.2014

Start date of project: 1st February 2011 Duration: 48 months

Lead Beneficiary: IIT (Vishwanathan Mohan, Ajaz Bhat)
Contributions: KCL, Profactor

Project co-funded by the European Commission within the 7th Framework Programme		
Dissemination Level		
PU	Public	
PP	Restricted to other programme participants (including the Commission	
RE	Restricted to a group specified by the consortium (including the Commission	
CO	Confidential, only for members of the consortium (including the Commission Services)	X

Contents

1. Organization of this Manuscript

This deliverable into three brief sections that summarize:

1) Why reasoning in Darwin and What critical aspects are addressed by the developed framework;
2) How does Darwin reasoning work? The analysis will span from the building blocks to the integrated functional architecture (with emphasis on what/how/where different kinds of learning and reasoning functionality is organized *and why*);
3) Validation of the framework in multiple scenarios, some inspired by studies from animal and infant cognition closely linked to iCub humanoid platform and assembly scenarios for the industrial setup including Y4 demonstrators, with a summary of what is new and interesting

Keeping the text as concise as possible, all the text in this deliverable is continuously linked to both relevant sections in the annexes (publications, user manual, other Darwin publications: DP) to facilitate the interested reader to get further details and also to sections of Y3 review report (to clarify which reviewer recommendations are addressed in the specific text/subsection), hence attempting to maintain the context.

1.1 Why Reasoning in Darwin?

This section briefly articulates the motivation for reasoning in Darwin and various functional/computational aspects that are addressed by the Darwin framework (in particular, describing *why* reasoning in Darwin and *what* does it offer).

Indeed, the most striking feature of any cognitive system (natural/artificial) is its ability to keep learning cumulatively (different things at different times, as necessary) and exploit such experiences flexibly when faced with novel situations, goals. As a simple but fascinating and realistic example (Kacelnik et al, 2002), Betty the new Caledonian crow had playful experiences of bending flexible pipe cleaners. A year later she faced a nontrivial problem of fetching her dinner basket trapped in a transparent vertical tube. A long wire was available nearby. After a short hesitation, she picked up the wire, fashioned a hook out of it using her beak and used the newly created tool to pull out her dinner basket. Even this little tinge of creativity demonstrated by Betty involved ***nontrivial "perception-action"*** *and* ***inferring*** *that the goal is directly unrealizable (given the present environment)*, ***recall*** **of "a specific" learnt past experience** evaluated as ***valuable in the context of the "present"*** (making hooks), some basic knowledge and ***anticipation of physical causality*** (i.e. the consequence of pulling the basket up using the newly created "hook tool" connected to the beak) and ***"flexible" integration*** of all this knowledge in the context of the ***goal***. Such situations (requiring goal directed reasoning) are indeed a commonplace in any unstructured environment (albeit a typical household setup or an industrial assembly line). To ultimately succeed, diverse 'chunks of knowledge' emerging from one's past experiences have to be integrated and exploited flexibly in the context of the 'present' to ensure smooth realization of goals, at the same time keep learning cumulatively. Undoubtedly, these are **critical desirable features** if robots are to become common place assistants in diverse application domains: domestic or industrial (Georg Stork, 2012) given the complexity of our society and economy.

In this context *(i.e. Why/ Why now)*, the Darwin reasoning system (described in this deliverable) offers an integrated biomimetic framework that develops a range of functionalities enriching the state of the art in this direction by:

Taking strong guidance from emerging trends in neurosciences, in particular integrating results from connectomics (Haggman et al, 2008, Sporns 2011, Bressler and Menon 2010, Van den Heuvel and Sporns, 2013), organization of conceptual knowledge (Patterson, 2008, Martin, 2009, Meyer and Damasio, 2009) and default mode network of the brain (Raichle et al. 2001, Buckner and Carroll, 2007, Buckner at al 2009, Suddendorf et al, 2009, Bressler and Menon, 2010, Welberg 2012, Addis et al, 2014) i.e. a network interacting cortical areas consistently activated in different conditions like recalling past experiences, predicting future consequences, goal directed planning and perspective taking.

a) *Enhancing the computational basis* for organization and flexible use of experience in cumulatively learning systems and goal directed reasoning ***in relation to (a)***; Specifically
   - What are the basic neural mechanisms underlying storage and recall of past experiences based on present context in an open ended "cumulatively learning" setup?
   - How remembered past experiences can be combined with explorative actions to learn and memorize something new?
   - How multiple remembered experiences can be "recombined" to generate novel behaviors in a new situation (without the need for explorative actions)?
   - What is the relationship between episodic memories of the robot and the core subsystems directly involved in perception and action when experience is gained originally?
   - Neural basis for forgetting: How multiple episodic memories "compete" to survive in the neural space and thus not be forgotten?
   - Putting all together: What are the basic computational processes governing the incessant interplay between "learning, memory, prospection and abstraction" in a cumulatively developing system?

   Annex 1 provides further details, supplemented with other attachments in the annexures that provide further results.

b) *Formulating and validating the developed computational models* in a range of scenarios ***in relation to (a-b)***: inspired by animal and infant cognition (as conducted on the iCub humanoid platform at IIT) and porting the integrated architecture to enhance flexibility in industrial assembly scenarios (with the robots at Profactor). ***Articles 1-4 in the annex*** (and other referred publications **DP**) address different validation scenarios. This deliverable further reports other ongoing experiments and results in the loop.

c) *Providing a User manual (Annex 5: in relation to a-b-c) that links the core computational* models with their implementation as software modules (mainly, Episodic memory, Observer and Neural PMP), what functionality, neural network, and underlying dynamics each subsystem/software module achieves in relation to the integrated architecture running on both iCub and industrial platforms, with possibility of porting to other robotic platforms. ***User manual in the attached annex*** *goes into the implementation details that are further linked in the next section (summarizing various computational models that lead to the integrated functionality).*

d) Reconciling the developed computational architecture **(b-d)** with the emerging results from neurosciences **(a),**

   (1) To provide further insights in relation to the interrelations between constrained exploration and the role of teacher/user (to demonstrate, instruct, reinforce) hence facilitating swift, green learning;

(2) Why top down and bottom up must share computational/neural substrates to close the loop between learning and reasoning,, trigger meta learning (i.e. abstraction) ;

(3) Our initiative to move away from learning different isolated tasks (in different robots) and instead look at organization of memory and the interplay between “learning, memory, prospection and abstraction” in a cumulatively developing systems, in general. ***These issues are further articulated in section 7 of Annex1.***

In the next section we concisely summarizes how (a-e) are functionally realized, from building blocks to the integrated computational/software architecture at present.

2. How does Darwin Reasoning work?

This section describes the central building blocks in the Integrated Darwin reasoning system, provides information related to the various computational models implementing the underlying functionality, details related to the different neural networks deployed and linked to further sections in the annexes to furnish further information to the interested reader.

2.1 From Building blocks to integrated functional architecture

Figure 1 shows the overall functional organization of the Darwin Reasoning system implemented by means of three core subsystems (namely, Observer, Episodic memory and Neural PMP) that provide different complementary functionalities (summarized inside the boxes). **In this section, we analyze each subsystem, summarizing the computational model, learning in the different neural networks and associated dynamics needed to achieve the claimed functionality.** The description is kept as brief as possible, providing basic information (what does it do, what is learnt, how the functionality is realized) and connected to sections in the attached Annexes **(Annex 1-5)** and other Darwin publications (**DP**), for further details.

2.2 Neural PMP Forward/inverse models: Learning and inferring the consequences of actions

2.2.1 Simulating potential consequences of actions : What and Why?

Forward/Inverse models of action and their relevance in any cognitive architecture has been a topic of recent debate *(Pickering and Clark 2014, **Annex3, DP1**)*. The reason for this is the growing consensus that cortical networks in the predominantly motor areas are activated in several other contexts related to ‘action’ that do not cause any overt movement. Distributed, multi-centered neural activity is consistently detected during different conditions like imagination of movement, observation/imitation of other’s actions, and comprehension of language (Gallese and Sinigaglia, 2011). The general insight emerging is that *the fundamental problems of shaping motor output during action execution and providing the self with critical information related to feasibility, consequence and understanding of potential actions (of oneself or others) are closely intertwined*. Prevalent modelling approaches generally converge on the role of forward models, but diverge on the perspective of how they might be realized in the brain or modelled computationally (see Pickering and Clark, Annex 3 for details). In this context, the neural PMP framework is an extension of old ideas from motor control like equilibrium point hypothesis (Bizzi, Polit and Morasso, 1972) and synergy formation (Bernstein, 1967), emphasizing the fundamental role of a “plastic, expandable” internal representation of the body (i.e. body schema) to realize diverse task specific forward/inverse models (**DP**: Mohan and Morasso 2011).

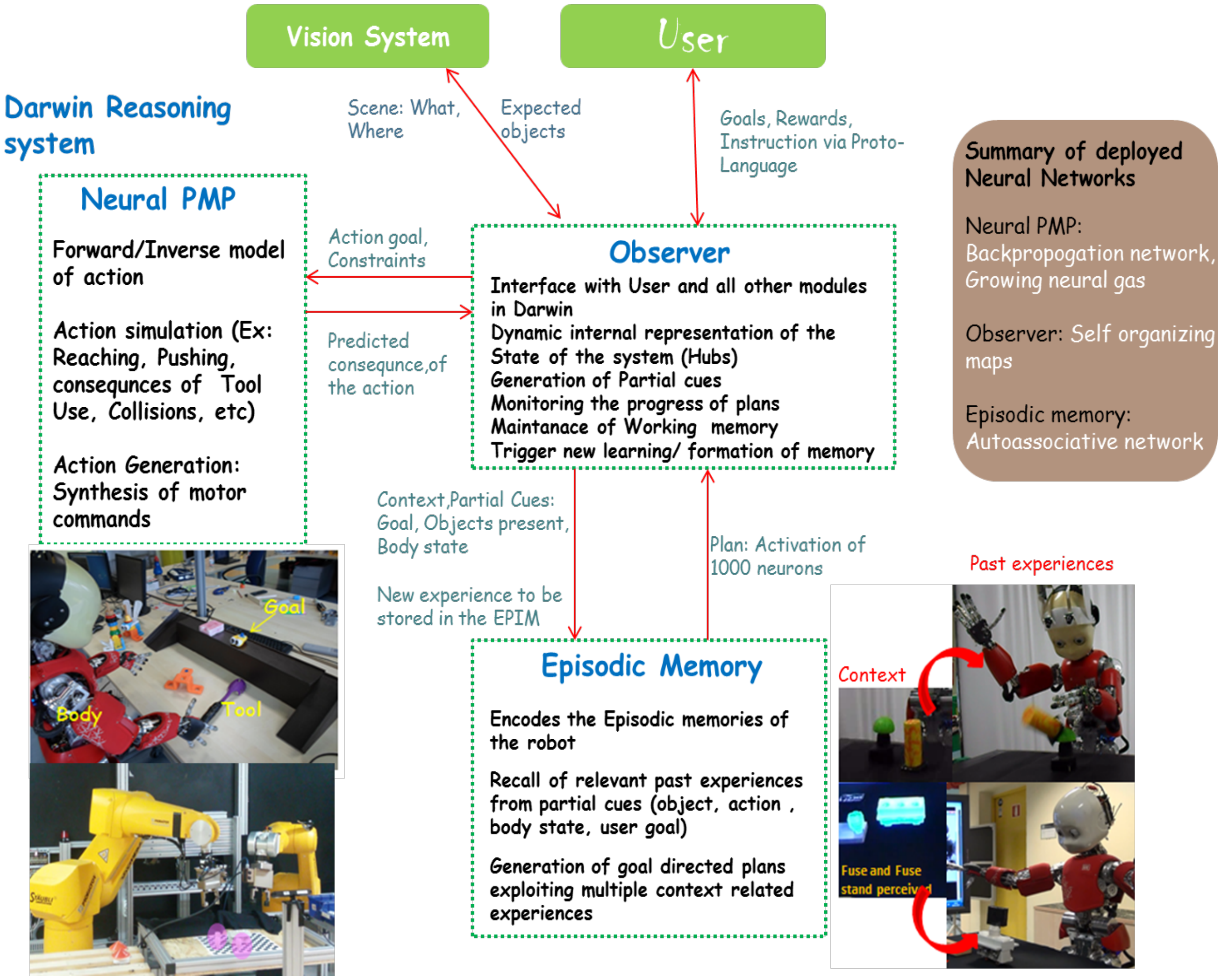


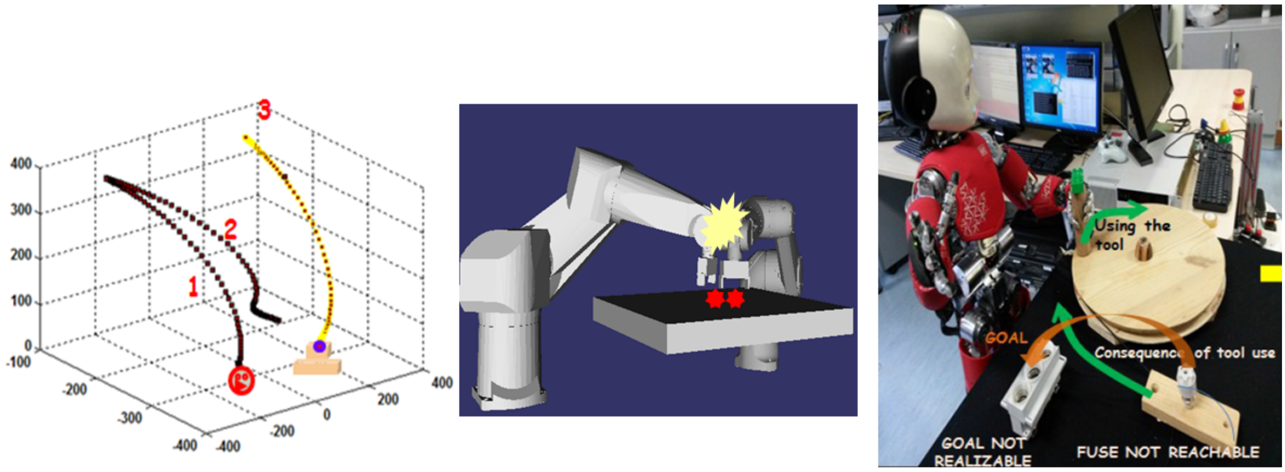
Figure 1. Functional organization of the Darwin Reasoning system implemented by means of three core subsystems (namely, Observer, Episodic memory and Neural PMP) that provide different complementary functionalities (summarized inside the boxes). Some example snapshots showing forward simulation of possible collision, covert simulation of reachability and possible tool use, recall of context relevant past experiences based on context (ex: objects in the scene, user goal), simulating future consequences and generating goal directed plans depict the kind of functionality realized by different subsystems. The box on the top right corner summarizes the different types of ANN's used in different subsystems.

The central features of the framework are:

a. It always operates **based on well posed computations** hence circumventing the need for kinematic inversions and cost function optimizations; In this context it is closely related to approaches from active inference (see, Friston 2011, **DP1**: Mohan and Morasso 2011).
b. Provides a unified framework to **synthesize diverse forward/inverse models on the fly** with different body chains coupled with coordinated tools. In this context, it closely resonates with several results providing evidence that tools during coordination act as an extension to the body schema (see, Iriki and Sakura, 2008, Umilta et al, 2008 for reviews).
c. Provides a framework to **learn** both the transformation between the intrinsic space (joints) and extrinsic space (of the body effectors) and the mapping between end effector to tool effector (for

coordinated tools). The learning is achieved through a **combination of babbling, exploration and imitation** (in the case of tool use).

The cumulative functionality realized by (a-c), endows Darwin platforms to both simulate and predict the consequences of different actions like reaching, tool use, collisions during parallel operation of the industrial robots, pushing and inversely generate the necessary motor commands, to control the robots and overtly execute the actions.



Inferring that the assembly goal is directly unrealizable in the environmental situation: Fuse graspable with the right hand (1) but cannot be inserted (2); Fusebox reachable by the left hand, affords pushing (3).

Forward simulation of collisions during paralell assembly with the industrial platforms

Inferring 1) the assembly goal is not directly realizable and 2) the cosnequence of using the tool (hence changing the environment to become more condusive towards realization of the goal)

Figure 2. Shows few examples where forward simulation of action leads to various inferences related to partially unrealizable goals, predictions related to consequences of alternative actions that have the effect of changing the present environment to become more conducive towards realization of the goal.

2.2.2 How? – Learning in the Neural PMP model

Here we briefly outline two aspects related to this subsystem:

1) Learning: How the forward/inverse models are learnt and represented in the ANN

2) How the underlying computational model makes use of the learnt information to engage in diverse forward simulations

1) **In relation to learning**, there are two different interrelated subtopics:

1) Learning the forward/inverse model for the body and

2) Learning the extension from the body to coordinated tool (and ensuing consequence's).

They are interrelated because the ***data and ANN on which the learning takes place has the same structure*** (i.e. mapping from one motor space to another, like joints to end effector, end effector to tool space) but ***slightly diverge on the ways in which learning itself takes place***. In the former case, data is generated by motor babbling while in the latter i.e. extension to tools, the same data (for linking different motor spaces) can be generated by a combination of imitation of the teacher's demonstration, physical interaction with the tool and reuse of past motor experience.

In relation to 1), figure 3a outlines the basic steps for any robot (iCub, industrial platform, other robots), starting from random babbling to explore the workspace to generate data that is used to train a standard backpropogation network. In relation to use of external objects as tools, the data can be generated as a combination of imitation the teachers demonstration (**DP2:** Mohan and Morasso, 2012, 2011a) hence constraining the domain of exploration (because a spatiotemporal trajectory comes from the teachers demonstration).

For both iCub and the two industrial platforms, a standard backpropogation algorithm. Specifically ***we used a forward neural network (equation 1) with two hidden layers*** to learn the mapping $X = f(Q)$ where $Q = \{q_i\}$ is the input vector (of joint angles), $X = \{x_k\}$ is the output vector (representing 3D position/orientation of the end-effector) and $Z = \{z_j\}$ and $Y = \{y_l\}$ vectors are the output of first and second hidden layer units respectively. Equation 1 expresses the mapping, where $\Omega = \{\omega_{ij}\}$ are connection weights from the input layer to first hidden layer, $O = \{o_{jl}\}$ are the connection weights between two hidden layers, $W = \{w_{lk}\}$ are the connection weights from the second hidden layer to the output layer, $H = \{h_j\}$ are the net inputs to the neurons of the first hidden layer and $P = \{p_l\}$ are net inputs to the second hidden layer.

For iCub the hidden layers contained ***48 and 55 neurons respectively****, while* ***for both industrial robots*** *the hidden layers contained* ***32 and 41 neurons*** *(a smaller network suffices for the industrial robots due to lesser number of degrees of freedom involved).* ***Neurons of the two hidden layers fire using the hyperbolic tangent function; the output layer neurons are linear. Section 6 of Annex 5(User manual)*** describes further detailed procedure for training the PMP related backpropogation network any embodiment, with specific details related to Darwin platforms (data generation, training, testing).

$$X = f(Q) \Rightarrow \begin{cases} h_j = \sum_i \omega_{ij}\, q_i \\ z_j = g(h_j) \\ p_l = \sum_j o_{jl}\, z_j \\ y_l = g(p_l) \\ x_k = \sum_l w_{lk}\, y_l = \sum_l w_{lk} \cdot g\left(\sum_j o_{jl}\, z_j\right) \\ \Rightarrow \quad x_k = \sum_l w_{lk} \cdot g\left(\sum_j o_{jl} \cdot g\left(\sum_i \omega_{ij}\, q_i\right)\right) \end{cases} \qquad (1)$$

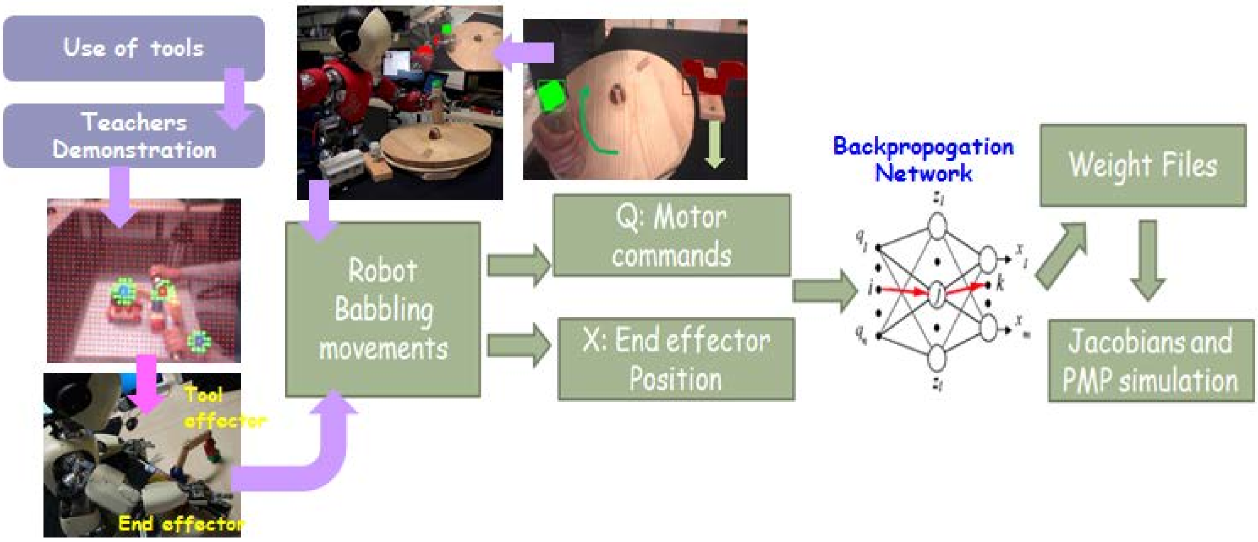


Figure 3. Shows the block diagram of the stages to generate data and learn the mapping between extrinsic and intrinsic space (or from extrinsic space to tool space). The data in the case of the latter can be generated by a combination of imitating the teacher and interacting with the object/tool. A standard backpropogation network is used, and from the learnt connectivity matrix, the Jacobians are extracted.

2) From the learnt neural net to the PMP network: The critical aspect is that from the learnt connectivity matrix of the neural network (based on the generated data by different learning streams:

babbling, imitation, physical interaction), it is possible to extract the Jacobians (encoding the geometric relationship between joint space-end effector space, end effector–coordinated tool effector space) using chain rule (equation 2).

$$J = \frac{\partial x_k}{\partial q_i} = \sum_l w_{lk} \cdot g^{-1}(p_l) \sum_j o_{jl} \cdot g^{-1}(h_j)\, \omega_{ij} \qquad (2)$$

This information is a basic building block to derive task specific PMP networks to facilitate goal directed forward/inverse simulations (i.e. consequences of one's actions and the generation of motor commands to realize it). Figure 4, shows two examples of task specific PMP networks that can be used for both covert simulation as well as overt execution of action. *Note that the Jacobians (i.e. horizontal links in the networks) connecting different motor spaces (joints-end effector-tool effector) are derived by equation 2, using the weight matrix of the learnt neural network.* **Section 6.4 of annex 5** (Neural PMP: computational model vs. implementation in software) further details of the information flow described in the PMP network and how it is implemented in the PMP software module.

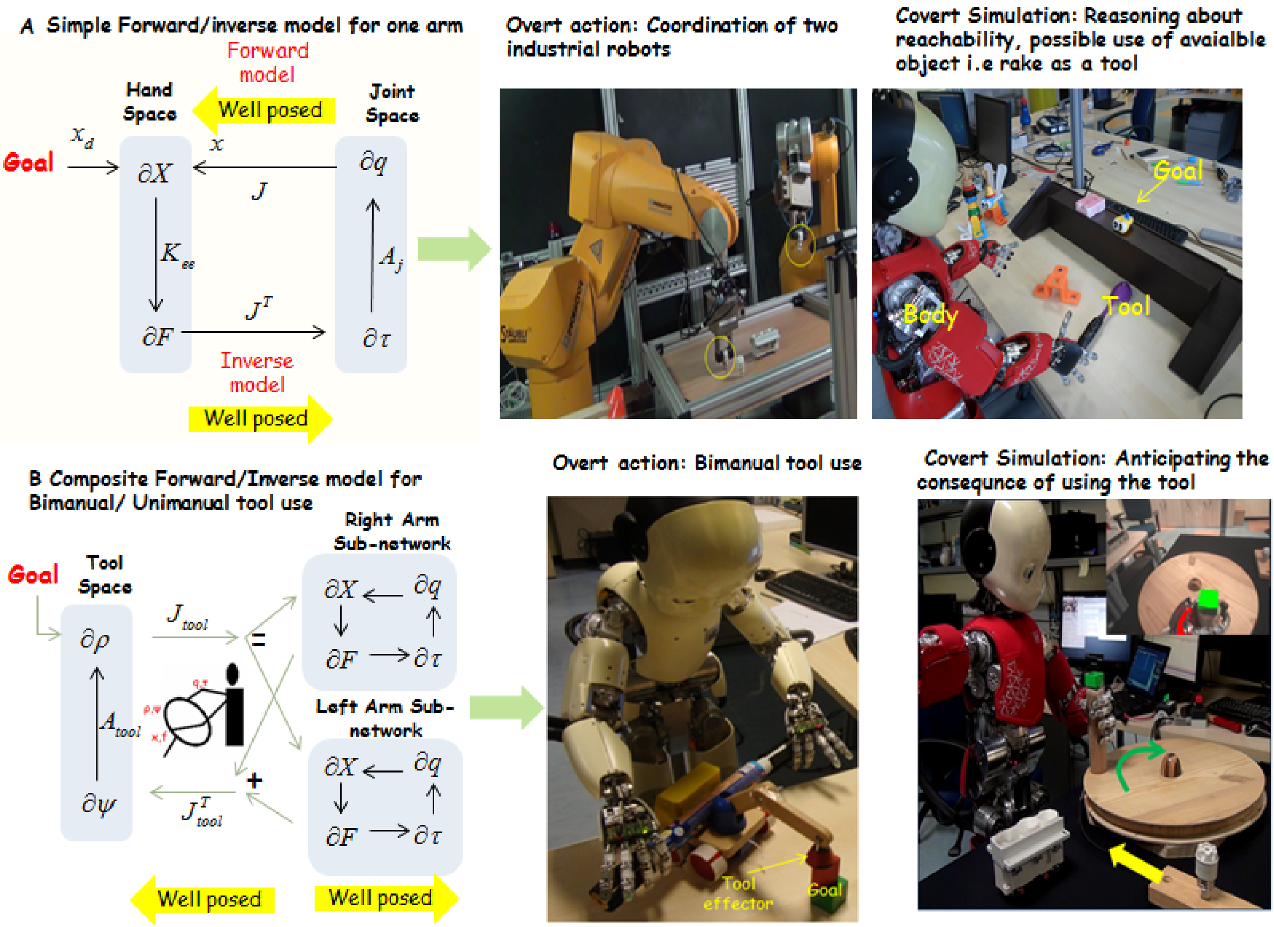


Figure 4: *Shows two examples of PMP based forward/inverse models based on the associated configuration of the body, coupled tools.* ***Panel A*** *: it shows a simple forward inverse model for one arm. In this simple case there are basically two motor spaces hand (or end-effector) space and joint space, both represented by two displacement nodes (dX: hand displacement, dq: joint rotation) and corresponding force nodes (dF: hand force, dT: joint torque). Vertical links characterize the elastic element (i.e. stiffness K and admittance A) while horizontal link characterize the geometric element i.e. the Jacobian encoding the relationship between hand displacement and joint rotation (that is learnt by the feedforward neural network). The network can be used to derive the necessary motor commands (dq) for overt action or anticipate the consequence. This provides vital information to either trigger overt action or reason about exploiting affordances in the environment like a rake tool (****panel E****).* ***Panel B****: it shows a composite forward/inverse model, for bimanual coordination with a connected tool (like a steering wheel or a toy crane). Here the network of panel A for one arm acts as a sub component, with the new addition of tool space represented in the same fashion with a force and displacement node. The tool Jacobian that encodes relation between the end effector and tool effector has to be learnt. Note that all computations are well posed: from the joint*

angles to the simulated position of the hand to from the force exerted by the two hands to the position of the tool effector (while the inverse will lead to infinite solutions).

2.2.3 Spatial planning using learnt Growing neural gas representing peripersonal space

During assembly in a typical unstructured setup, it is inevitable that based on the spatial arrangement of the objects in the scene (i.e. changing), appropriate action sequences must be generated by both robots to operate in parallel (with or without collision avoidance) and realize as many assemblies as possible. Figure 5 shows three such cases: scene 1 is well structured allowing spontaneous operation of both robots performing different assemblies (Ex: RX assembling FB1 and TX assembling FB2). In scene 2, objects are scattered randomly in the workspace and planning which object to act on (at different time instances) is necessary to facilitate parallel completion of multiple assemblies by both robots. Scene 3 is both unstructured and redundant (more fuse boxes than fuses). In general, to maximize parallelism (with minimizing the need to trigger collision avoidance that increases the assembly time), inevitably requires a flexible spatial planning system that given an environmental scene dynamically allocates tasks to the two parallelly operating robots. Interestingly, this problem can be connected to foraging and navigation experiments (Toussaint, 2008, Mohan et al 2011) considering the analogy of rats navigating for food with Darwin robots foraging for fuses and fuse boxes. In this context, the PMP network was extended with a GNG (Fritzke 1995, Mohan et al, 2011) that explicitly represents the peripersonal space of the robot/embodiment that facilitates a range of spatial planning. *The same data generated during the learning of the Neural PMP (figure 3), in particular randomly explored samples of extrinsic space is used to learn the growing neural gas.* ***Section 6.4 of Annex 5 (user manual) details the process of learning the GNG from the data generated.***

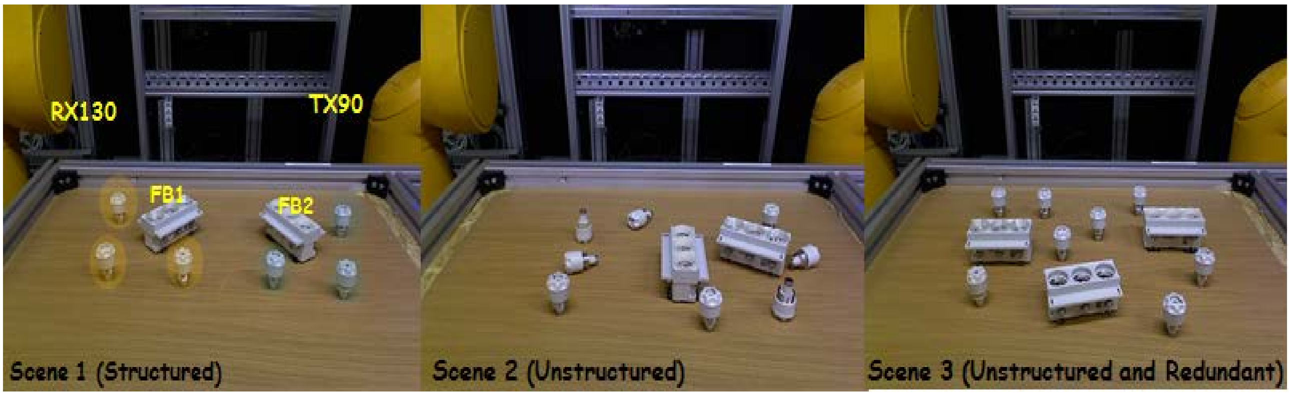


Figure 5. Shows three examples of different spatial arrangements of objects in the scene, requiring both robots to flexibly plan their individual actions so as to operate in parallel and complete multiple assemblies. As seen scene 1 is well structured allowing spontaneous operation of both robots performing different assemblies (Ex: RX assembling FB1 and TX assembling FB2). In scene 2, objects are scattered randomly in the workspace and planning which object to act on (at different time instances) is necessary to facilitate parallel completion of multiple assemblies. Scene 3 is both unstructured and redundant (more fuse boxes than fuses). In general, based on the spatial configuration, appropriate action sequences must be generated by both robots to operate in parallel (with or without collision avoidance) and realize as many assemblies as possible.

Figure 6 top panels show TX robot exploring 10K locations and the resulting learnt GNG network consisting of **478 interconnected neurons** with linear activation functions. The same is done for the RX robot, and figure 6 bottom panel shows the combined representation of peripersonal space for both robots (with intersections i.e. areas where they can both act). A moving neural field superimposed on the GNG based on the anticipated reward fetched (for choosing a particular object in a spatial location) organizes the actions taken by both robots. Figure 6, panel 4 shows a typical environment with objects scatted in the workspace and choosing the right objects to act on at different time instances facilitates parallel operation with or without collision avoidance. Section 3.1 presents results related to realization of 2 assemblies (i.e. 6 fuses in two fuse boxes) in parallel by both industrial robots operating the shared workspace, in a typical unstructured setup. We also refer the **interested reader to (Mohan et al 2011b and Toussaint 2006), for further details, other applications using the same GNG system for spatial reasoning.**

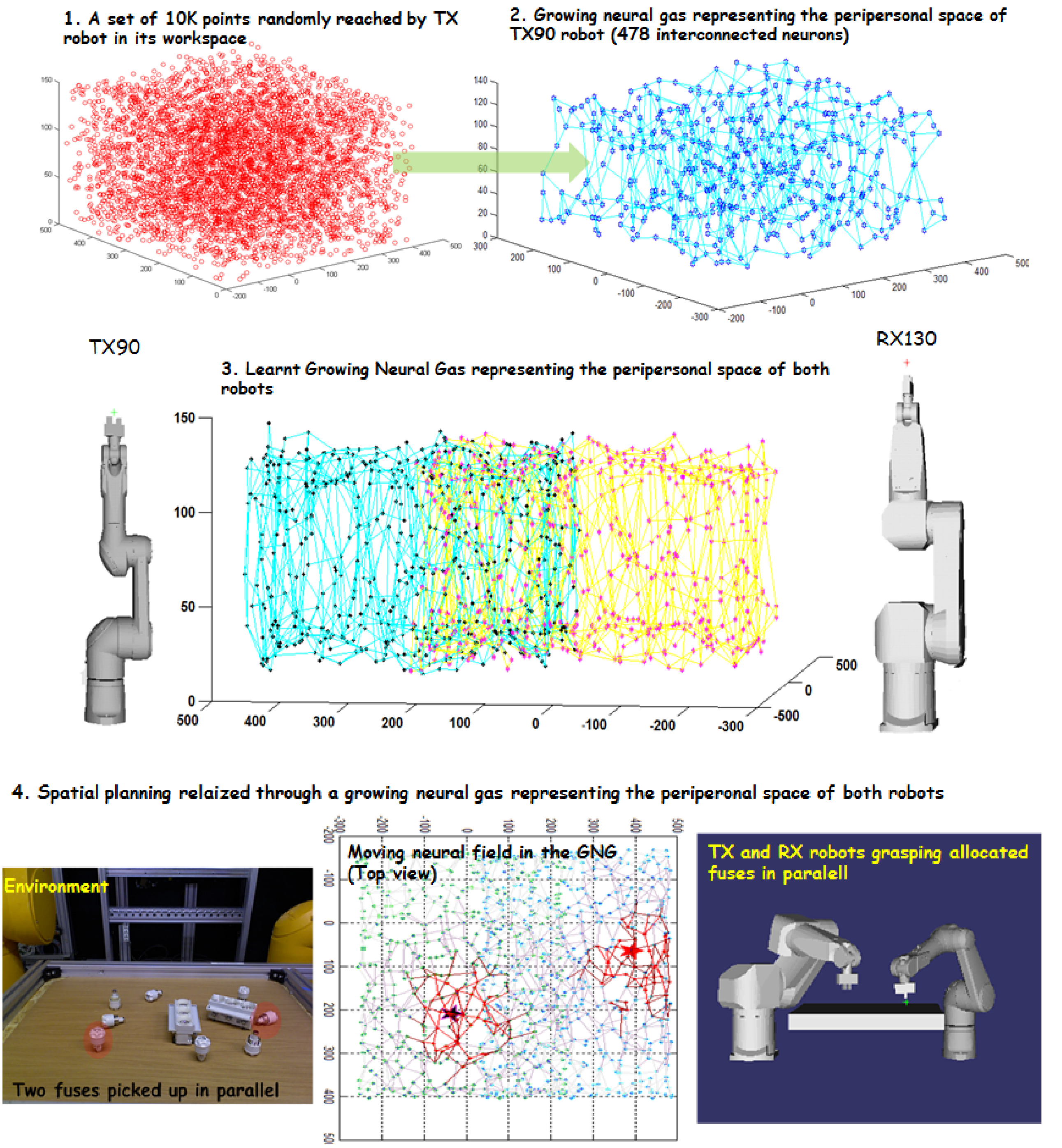


Figure 6. Panels 1-2 show the random points in space explored by TX robot (while learning the neural PMP) and the growing neural network representing its peripersonal space. Panel 3 shows the learnt PPS of both robots on which reward based spatial planning is realized, to facilitate parallel operation to complete multiple assemblies. Panel 4, shows a typical assembly environment where objects are scattered in the workspace and choosing the right objects to act on at different time instances facilitates parallel operation with or without collision avoidance. Middle panel, shows the moving neural field on the learnt GNG that organizes the spatial planning and right panel shows the corresponding simulation of the two robots parallelly approaching to grasp the allocated fuses in the scene

2.3 Observer-Episodic memory loop

While the previous section outlined the systems that enable Darwin robots to infer about actions (and consequences of potential actions), this section describes the other two core building blocks in Darwin reasoning architecture namely, the Observer-Episodic memory loop (see figure 1, for a summary of the functionality realized by this subsystem). This loop implements multiple functionalities from the being main interface catering to the user, to recalling past experiences in relation to present context (ex. objects in the environment, user goals, internal state of the system), generate goal directed plans, anticipate future consequences, monitor the execution of the plans, detect inconsistencies hence triggering new learning, formation of new memories. In sum, this loop performs the crucial function of efficiently channelizing information related to perception, action, body, knowledge (i.e. past experiences of the robot encoded in its memory) towards realization of goals (and learn new things as necessary in that process). The architecture is strongly inspired by emerging trends in neurosciences (***see section 1.1 in Annex 1***) and basically deploys two different kinds of neural networks 1) interconnected SOMs in the Observer module to organize sensorimotor information and 2) Auto-associative memory network in the Episodic memory module to recall context relevant experiences and from goal directed plans. In the sections that follow, we summarize the underlying rationale and biological inspiration, how sensorimotor information is organized with details related to the deployed neural nets, how learning takes place and experiences are organized and remembered (**linked further to sections in the annexes and user manual to furnish further details**).

2.3.1 Organization of Sensorimotor information in the Observer: Role of Hubs

The higher level organization of sensory, motor and cognitive information in the reasoning system architecture both connects and is guided by three recent trends in neurosciences:

A. Functional imaging studies providing evidence that conceptual information is organized in a “distributed fashion” in “property specific” cortical networks that directly support perception and action and that were active during learning;

B. Studies from the field of connectomics suggesting the presence of a small set of Hubs that enable swift global integration and cross-modal communication between multiple subsystems that process/represent various sensory, motor and cognitive information and organized in a distributed fashion (as substantiated by 1);

C. Converging evidence that a core network of “highly connected” brain areas (Hubs) called as the Default Mode network is consistently activated when subjects perform diverse cognitive functions like recalling past experiences, simulating possible future events (or prospection), goal directed planning, imagining fictitious scenarios, interpreting perspectives of others. This suggests that disparate cognitive functions often treated as distinct, might share common neural processes that crucially deal with the “memories” of one own individual experiences and how it is exploited in various “contexts”. Such a perspective urges to view memory not just as “past oriented” but also “future oriented” in other words as a key component of the prospective brain that actively facilitates simulation of future events, formation of flexible plans and predictions.

Figure 7 shows a block diagram of how basic information related to perception and action are organized in the observer and their link to the neural episodic memory of the robot. As seen, bottom up perceptual information coming from visual analysis (color, shape) activate level 1 SOM’s organized in a property specific fashion. The rationale behind this choice is both emerging results related to organization of conceptual information (**A**) and the fact that such an organization facilitates learning which properties are causally dominant for different tasks (**see, section 3.5 in Annex 1 and Annex 4** for two different scenarios). Word inputs are entered by the user via key board and are converted into stimulus vectors on the basis of letter usage frequencies in English language as is done in (Hopfield, 2008). The activity in the level 1 map feed the Object hub that cross-modally integrates information coming from the individual maps (inspired by emerging studies from connectomics: **B**). As a simple example, a novel word input like "Red Container" can activate the object hub (bottom up) that in turn causes activity in the color and shape map (top down) corresponding to what the robot anticipates a 'red container' to be (see **DP**: Mohan et al, 2013). Further, Object hub activity acts as partial cue for the episodic memory network to recall context relevant past experiences (we deal with this aspect in the next sections). While **DP: Mohan et al 2013**, provides precise details related to how "color-word-shape-hub" network is learnt, and **section 7.3 in Annex 5** describes the implementation details, here we provide a summary below giving details related to the different maps and associated learning method.

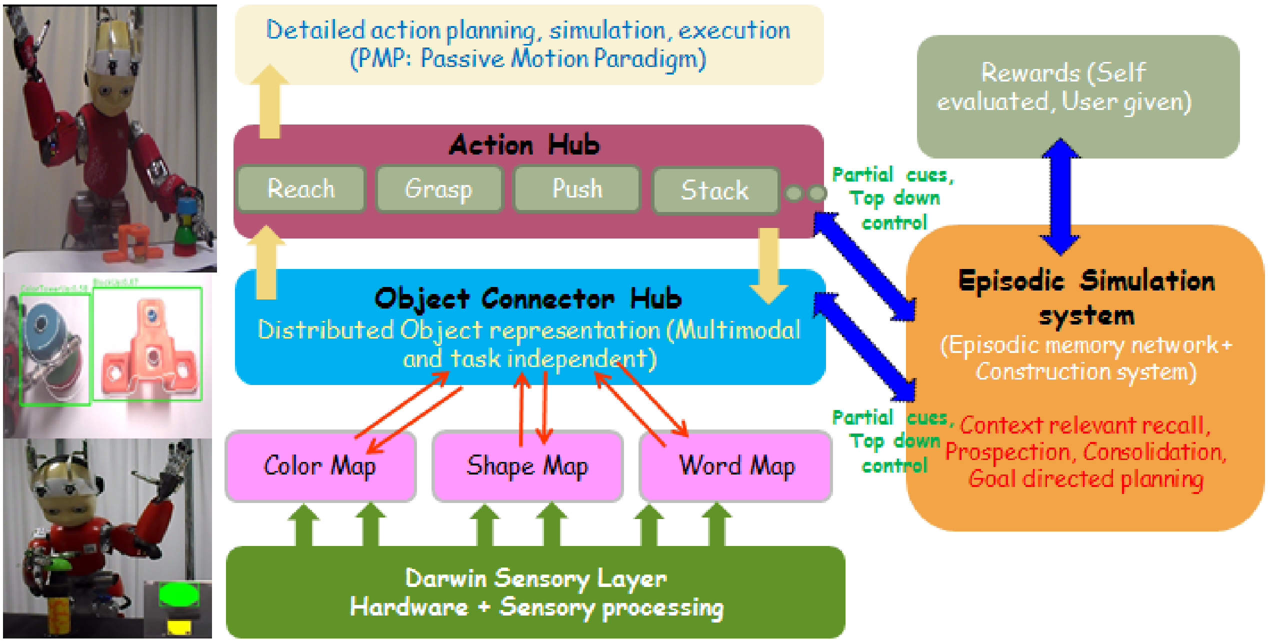


Figure 7. Shows a block diagram of how information related to perception and action are organized and their link to the episodic simulation system. There is a distributed "property specific" organization of sensorimotor information, integrated through a small set of hubs. Temporal sequence of activations in the hubs when experience is originally gained is used to form the episodic memory trace. At the same time, bottom up activations in the hub provide partial cues to trigger context related recall. Activations in the episodic memory network in turn modulates top down the hubs to mediate fundamental processes like combining past experiences with exploration, flexibly connecting multiple experiences in a novel situation, consolidation and forgetting.

2.3.2 Details of Neural networks and Learning in the Observer

Both the level 1 maps and the hubs of figure 7 deploy standard self-organizing maps (Kohonen 1995), with novel concepts introduced in relation to learning the inter-connectivity between the maps and underlying neural dynamics (that allows activations in one map to trigger other connected maps). Let N be the number of neurons in any SOM and S be the dimensionality of the bottom up stimulus feeding the map. Then the connectivity matrix has a dimensionality of N x S. Since we are dealing with multiple maps here, for clarity we address N_C, N_S, N_W, N_H as the number of neurons in the color, shape, word, Object hub respectively. In the present implementation **N_C=30, N_S=30, N_W=30, N_H=42,** respectively. This choice was made also in relation to the structure of the neural episodic memory (i.e. consisting of 1000 neurons arranged in a sheet like structure 20x50) that will be described in the next section. Only the activations in the hubs (object, action and body) enter the episodic memory (and not color, word or shape that just feed the Hubs). Since color, word and shape SOM activity forms the bottom up input to the object hub as shown in figure 7, the connectivity matrix of the Object hub has a dimensionality of N_H x (N_C+N_S+N_W), i.e. **42x90** here. Figure 8, top two rows illustrates examples of learning the individual SOM and interconnections between them (see **DP: Mohan et al, 2013** for several other examples). In each case, the robot is presented with a new object followed by the linguistic input of what it is from the teacher. In the examples of figure 8, the robot is presented with a fuse and a holder respectively, both recognized by vision feeding the Shape map, while the linguistic input coming from the teacher feeds the word map*.* **Layer 1 maps are trained in parallel** using the SOM procedure that is fairly standard (see Kohonen 1995). In short, this consists basically of two steps:

1) Finding the neuron 'i' that shows maximum activity for the observed sensory stimulus S^t at time t. This also implies that neuron 'i' sensory weights s_i such that $||(s_i-S^t)^2||$ has the smallest value, among all neurons existing in the respective SOM at that instance of time.;

2) Adapting the sensory weights of the winner in a Hebbian fashion by bringing the sensory weights s_i of the winner "i" closer to the stimulus S^t. This simply has the effect that in future instances the neuron "i" actively codes for the particular sensory stimulus S^t. In this way neurons in different property specific maps of layer 1 that have sensory weights closest to the incoming input sensory vector start representing these signals.

The net activity in the color, word and shape SOM's forms the bottom up input to the Object hub. The connectivity is of dual dyad type, and **weights between layer 1 maps and hub are adjusted as follows**:

'if neuron "i" and neuron "j" winning in the shape and word SOM's respectively manage to activate neuron "k" in the provincial hub , make W_{ik} =1 and W_{jk} =1.

This has a net effect of enabling neuron "k" , "i" and "j" in three different SOM's (operating on their own local sensory streams) to retroactivate each other in "bottom up", "top down" and "cross modal" fashion. The internal weights of the provincial hub can either have random initialization or a winner "k" can be randomly chosen from the subset of neurons in the provincial hub that have internal weights zero. The net effect is that in both cases there is some neuron in the' provincial hub' that responds to activity in two different SOM's processing different sensory streams. In the future, both the perception of an object (for example fuse) or teacher inputting the word "fuse" can activate a global interconnected network. This behavior is very common in infants (show them a "dog" for example and say the word "bow bow", next time the child sees a dog, we often see it playfully pointing to it with the word "bow bow"). *Studies in functional imaging go even further providing evidence that even if it was a toy dog, a real dog or a cartoon or just the word "bow bow" should activate the global network as was experienced during learning (Martin, 2007).* This is indeed also the case in our computational framework: perception of an object like fuse can activate its word representation or vice versa. **DP: Mohan et al, 2013** illustrate how even novel combinations of words are handled in the proposed framework demonstrating how this small network of SOMs have their local inferential capabilities. ***The more interesting aspect is the interconnection between the hubs (in the observer module) and the episodic memory network (in the Episodic memory module) dealt with in the next section (and illustrated in the bottom panel of figure 8).***

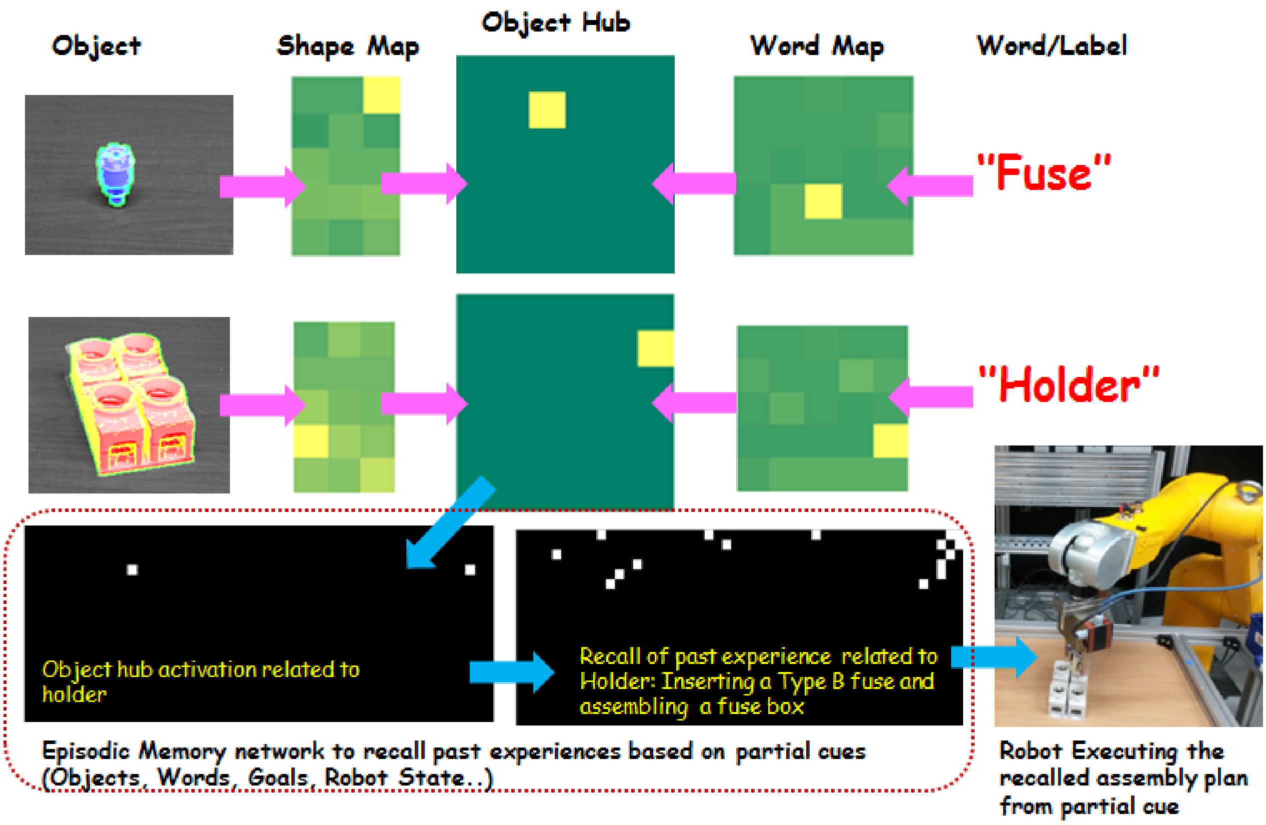


Figure 8. Top two roes show two examples jointly training the Shape-word and Object Hub, by presenting an object along with a linguistic label inputted by the teacher, using the method described in 2.3.2. In the proposed framework there is a direct connection between activations in the hub and the episodic memory (the content of the robots episodic memory is the temporal sequence of activity either in the object, action hubs or reward received from user, when real experience was originally gained by the robot and encoded in the episodic memory).The power of such distributed multimodal organization is that in the future either the word “Holder” or perception of the object in the scene can activate the object hub, hence generating partial cues in the episodic memory. The dynamics of the episodic memory then allows the robot to recall past experiences in relation to the context: Inserting a Type B fuse into the holder to assemble a fuse box, and the anticipated rewards for such actions (bottom panel).

2.3.3 Link between Hubs and Episodic memory network

In the previous section, we summarized how bottom up perceptual information or linguistic input form the user is used to jointly train layer 1 SOM’s and the Object Hub to which they are connected. In this section we describe the relationship between activations in the Hub and the neural episodic memory (**Section 3, Annex 1** provides detailed information that we summarize here). In general, experiences are either gained through direct interactions between the body and the world or socially through experiences of others (for example: User of Darwin), and this must reflect in the episodic memories of Darwin robots. Here comes the link between Hubs and the neural episodic memory of the robot in our computational framework (figure 7). We clarify this here and point out the rationality behind our proposition. To decide what information to encode, we looked at emerging results related to the recently discovered default mode network (DMN). It’s been established through numerous studies that central function of this network is to generate “self-referential episodic simulations”: that includes re-construction of the past experience in relation to a present context (partial cues), simulation of possible future scenarios and planning, consolidation of new experiences with the existing knowledge, some forms of spatial navigation, perspective taking. Such episodic simulation activates the highest level brain hubs involved in object, action and value processing. This makes sense because while operating in a real world, partial cues (related to perception, action, goals and rewards) emanate from the environment activating bottom up the highest level neural maps (like, object connector hub, action hub in **figure 7**). **Thus, there is both biological grounding and computational simplicity in considering that activity in the highest level object-action hubs fill in partial "contextual" information, triggering a reconstruction process of remembering the complete past experiences (figure 8, bottom panel). Thus, the content of the robots episodic memory is the temporal sequence of activity either in the object, action hubs or reward received from user, when real experience was originally gained by the robot and encoded in the episodic memory (this is further illustrated in figure 10).**

2.3.3 Details of the neural network in the episodic memory network

The architecture of the Darwin episodic memory builds up on a recently proposed excitatory-inhibitory neural network of autoassociative memory (Hopfield, 2008). This network which deals with basic "storage and retrieval" mechanisms is taken as a starting point and further enriched in the context of a cumulative developmental "learning/ reasoning" framework where (a) experiences are cumulatively acquired by the robot by interacting with the world; (b) number of memories grow with time, some eventually forgotten, some consolidated (c) multiple memories of past experiences need to be retrieved based on context and goals, may have to be "causally combined" to generate novel creative behaviors**'. Section 2 of Annex 1 provides full details that we summarize here.**

In the present computational model (Mohan et al, 2014), we deal with a small patch in the sheet like neocortex, consisting of 1000 pyramidal cells (N=1000). The 1000 neurons are organized in a sheet like structure with 20 rows each containing 50 neurons. Every row may be thought as an event in time (related to activation in object hub, action hub or received end reward) and the complete memory as an episode of experience (for example, picking a cylinder and placing it on a mushroom and getting a reward of 0 from the user: **Annex 1 Section 3.1**). Importantly, in the memory network of 1000 neurons, multiple episodic memories can be represented and retrieved (approximately 230 episodes as reported in Hopfield 2008). So activity in the 1000 neurons are not just used to exclusively represent one episodic memory, but rather many different episodic memories in a cumulative learning set up (for example, building the tallest stack on *day 1* with mushroom and cylinder, playing on *day 2* with cubes and containers, learning to assemble a Fuse box on day 3: **Annex 3 , section2**, so on). The memory circuit is characterized by "all-to-all" connections between the N excitatory neurons (thus the connectivity matrix is of the order **NxN**). *Memories are stored in the network by updating the connections between different neurons using Hebbian learning. In addition, there is an inhibitory network that is equally driven by all N excitatory neurons and in turn inhibits equally all excitatory units (**Section 7.2 of Annex 5** provides implementation details). **A rate-based model is used, in which the instantaneous firing rate of each neuron is a function of its instantaneous input current (Hopfield, 2008, Mohan et al 2014).** In the next section we summarize how memories are encoded and recalled based on context.*

2.3.4 Learning and encoding experiences in the Observer-Episodic memory network

Building up on 2.3.1-2.3.3, this section describes how experiences are learnt and encoded in the episodic memory network. In relation to the ongoing open ended design of our reasoning architecture, we summarize three crucial aspects related to learning and encoding of experience into the episodic memory network

a. ***When does learning take place*** i.e. the situations that trigger learning in the present architecture;
b. ***How does learning take place*** i.e. the various input streams with which experience can be encoded in the episodic memory;
c. ***What is represented and how*** encoded past experiences are reconstructed from partial cues in the present to form goal directed plans;

To keep the description brief, we further point the interested reader to sections in the attached annexes that provide detailed examples and validation of (a-c) in Darwin robots.

Figure 9 illustrates our approach to issues (a) and (b). In particular, learning takes place under the following circumstances in the present Darwin architecture

1. **No past experience** exists and the robot explores to gain experience (**Annex 1, section 3.2**)
2. **Novel object** in the environment along with past experiences related to other objects present: Robot combines past experience with exploration to learn new action sequence encoded into the episodic memory (**Annex 1, section 4, Annex 4 section 3**)
3. **Dissonance:** Contradiction between the present situation and what the robot anticipates (**Annex 5, Section 7.4**), robot reports the inconsistency to the user and learns by user intervention. Dissonance might also trigger exploration that leads to successful realization of goal without learning (see section 3.3).
4. **User Instruction via proto language:** to teach a new assembly plan to facilitate quick switch over to novel objects, already learnt and encoded in the shape-hub network of 2.3.1. This procedure is described in **Annex 2, Annex 5, section 7.6 and a new example** presented in section 3.4 of this deliverable.
5. **User demonstration and robot exploration** as in the iCub demonstrator planned for year 4, further described in section 3.5 and **DP: Mohan et al 2011, 2012.**

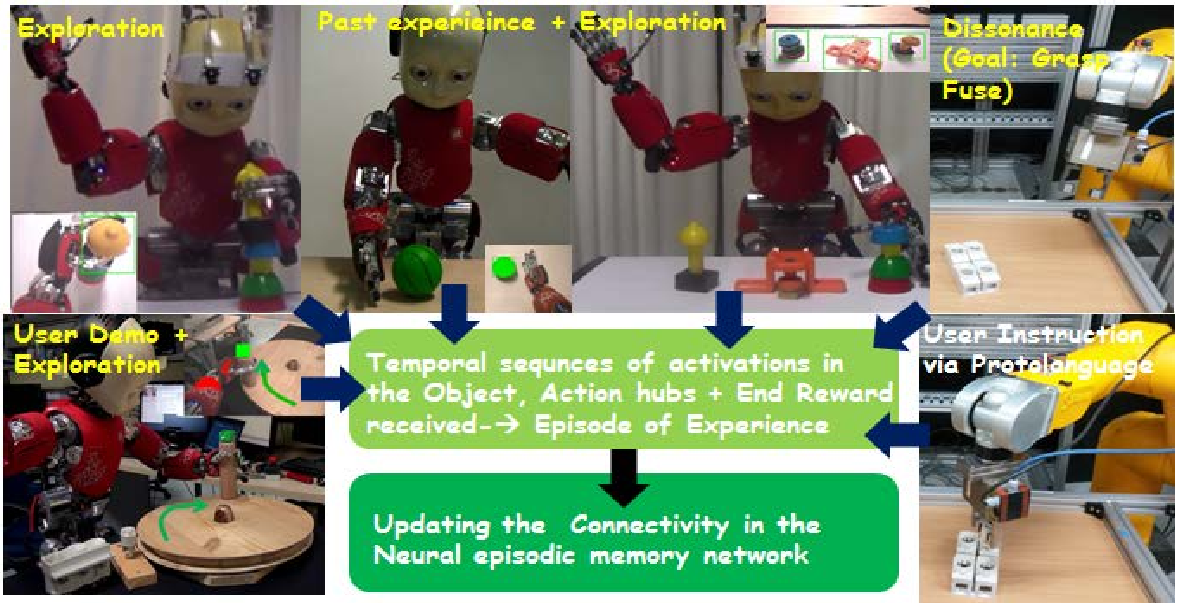


Figure 9. Illustrates few Darwin instances of learning and encoding experience into the episodic memory network, depicting the generality in the proposed approach: multiple things can be learnt at different times and stored in the same neural network and remembered based on context to realize goals. Further note the underlying rationale of the link between Hubs and Episodic memory: weather the knowledge is gained by exploration, or combining past experience with exploration, user instruction or demonstration, all of them lead to a temporal sequence of activations in the hubs. This temporal sequence of activations + the end reward received is taken as an episode of experience. Such experiences are encoded into the episodic

memory network by adapting the weights. In the present network, approximately 250 such experiences can be encoded and recalled based on context.

What is represented?

As has been already described above in 2.3.2, the content of the episodic memory is the temporal sequence of activity either in the object, action hubs or reward received from user, when real experience was originally gained by the robot (via multiple learning streams as illustrated in figure 9). The 1000 neurons in the network are organized in a sheet like structure with 20 rows each containing 50 neurons. Every row may be thought as an event in time (related to activation in object hub, action hub or received end reward) and the complete memory as an episode of experience (for example, picking a cylinder and placing it on a mushroom and getting a reward of 0 from the user. Figure 10 illustrates the idea with a simple example. The same applies in more complex cases, for example of inconsistency between what the robot anticipates and what it observes, engages in exploration to acquire new experience (**Annex 5**: merging partial plans with explorative actions) or learning by instruction via User-Observer protolanguage (in which case, the words inputted by the user activate the word map hence activating the hubs in a temporal order as instructed by the user). The present experience (i.e sequence of activations in the hubs) is always recorded in the Observer and can be encoded into the episodic memory network by updating the connectivity between neurons as described in the next subsection.

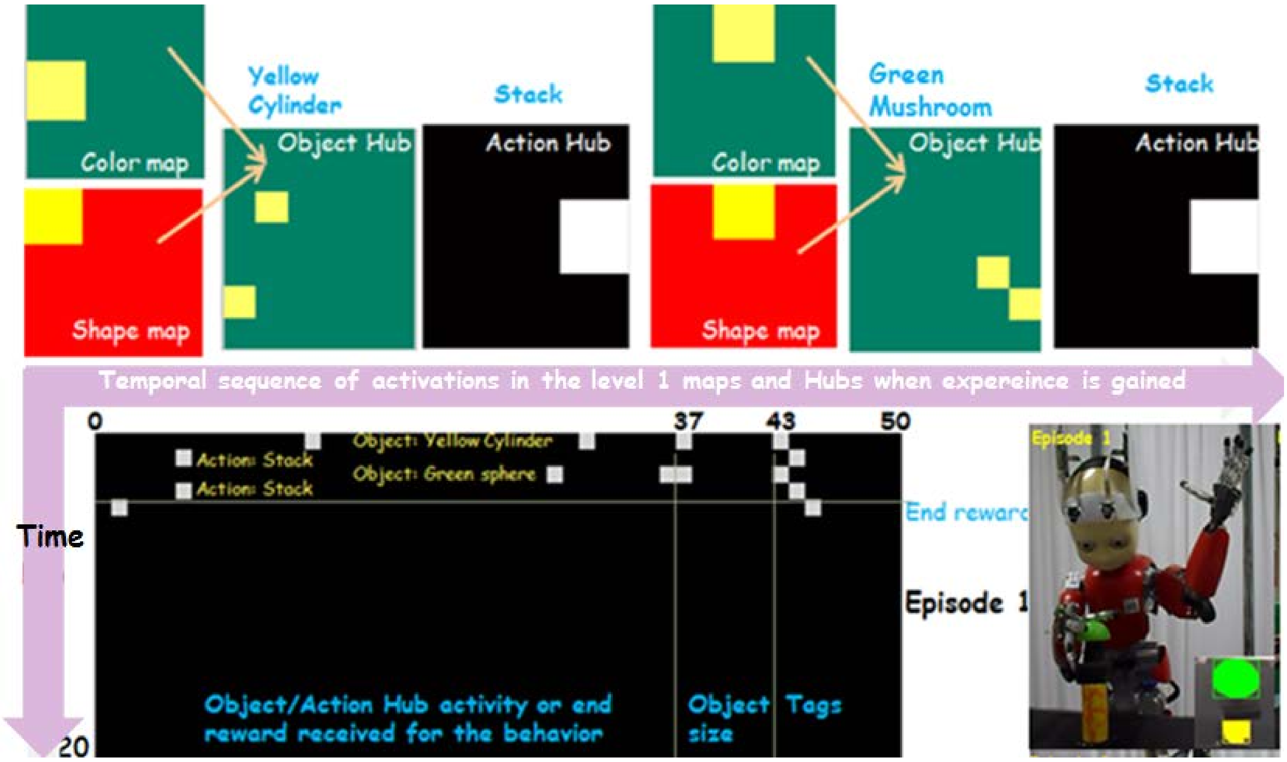


Figure 10. Top panel shows the temporal sequence of activations in the hubs (and level 1 maps that trigger the hubs) when experience is originally gained, while the bottom panel shows the same sequence represented in the episodic memory network organized in a 20x50 sheet, every row representing an event in time or the end reward received.

Updating the Neural connectivity in the Episodic memory to learn new experiences

New experiences gained by the robot (through 1-5) are learnt by updating connections between the neurons in the episodic memory network in the following way. Let V_{new} be a one dimensional vector representing the activity of N (N=1000, here) neurons shown as a 20x50 matrix (figure 2 top panel). Let T denote the connectivity matrix between the N neurons. Since there are 1000 neurons, the dimensionality of T is 1000x1000, that represents the strength of the connection between any neuron "i" to any neuron "j". *T is a null matrix to start with as nothing is known.* Consider that a new episode represented by activity V_{New} has to be stored in the memory network. This is done by updating all the connections T_{ij} between the N neurons, using a simple rule summarized below:

If V_i=1 and V_j=1, then make T_{ij}=1(regardless of what its value was before);
Else, make no change to T_{ij}. *(3)*

Remembering past experiences from partial cues*:* The interesting feature of the proposed computational framework is that the content of the episodic memory correspond to activity in the object or action hubs (when experience was gained). This has additional advantages in terms of generation of partial cues to trigger recall of context relevant past experiences, because objects in the real world or user goals are often the sources of partial cues. For example, on day X if the green mushroom is placed in front of the robot, bottom up perception will lead activations in the object hub (corresponding to the mushroom) that fill in partial information: or activate a sub set of neurons in the 50x20 neural episodic memory network. This serves as initial condition to trigger recall of past experiences (in this example, in relation the object mushroom). The network dynamics to recall past experiences from partial cues is given by equation 2: V_K is the activity in the K^{th} neuron (a subset of neurons in the 50x20 neural episodic memory layer will be active based on the partial cue or initial condition). T is the connectivity matrix between the neurons learnt using equation 1 when the episodes of experiences are encoded in

$$\tau_{rel}\dot{V}_k = -V_k + \sum_{j=1}^{N} T_{k,j} V_j + I_{inhib}$$
$$I_{inhib} = g(-\alpha^{in} + \beta \sum_k V_k) \qquad (4)$$
$$g(i) = 0, if\,(i<0), else, g(i) = i.$$

the network. 'I' is the current coming from the inhibition network that is modeled as a single neuron. The function of the inhibitory network is to keep the excitatory system from running away, to limit the firing rate of the excitatory neurons. At low levels of excitation the inhibitory term generally vanishes. **For all experiments α^{in} was chosen as 30, τ_{rel} as 1000 and β as 3.5. Sections 7.2 annex 5 describes details of how the dynamics of episodic memory is implemented in software. Section 6 Annex 1, presents an analysis of the performance of the neural network if the critical parameters in the neural network are modified.**

In sum, multiple experiences of acting and learning from different scenarios, objects over time acquired through multiple learning streams, in a cumulative fashion can be encoded in the same network of 1000 neurons, organized in a 50x20 sheet (every row an event in time). The critical feature of the framework is that the content of the episodic memory corresponds to the temporal order of activations in the object and action hubs or the received reward, when experience is originally gained. The straight forward advantage is that any time in the future, if the hubs are activated bottom up (example, the robot perceives a fuse), the activations in the hubs automatically act as partial cues, providing context and triggering the recall of context relevant past experiences. The recall of past experiences also allows the robot to anticipate the future perceptual consequences in the present context, anticipate the fetched reward and form goal directed plans (as a simple example, from perception of a mushroom to recalling past experience of placing a cylinder on top of it and getting a reward of 0, or from the user input "fuse" to the assembly plan of inserting it in the fuse holder to assemble a fuse box). This is in line with the emerging results in relation to the Default mode network (), that recalling the past, simulating the future and forming goal directed plans share cortical substrates in the brain. The next section presents further results from different Darwin scenarios exploiting the computational architecture.

3. Integrated Darwin Reasoning in Action

This section presents a collection of results highlighting different interesting aspects of the integrated reasoning system as validated on the Darwin robots (both iCub and industrial platform) in different Darwin scenarios. Specifically 5 different tasks (Assembling the tallest possible stack, Fuse Box assembly, Spatial planning for parallel Assembly with two robots in a shared workspace, From Proto Language to episodic memory to learn a task with new objects, ongoing iCub demonstrator for Y4) are chosen to illustrate that the computational/software architecture is general enough to go beyond isolated examples demonstrating the functionality. This section is further supplemented with other examples in the attached annexes that we link appropriately to point the interested reader to other examples of reasoning in action, the resulting system behavior/functionality. Finally, a comprehensive discussion on the novel features in the Darwin Reasoning architecture, connecting to emerging trends in neurosciences and insights gained form the computational model is described in **section 7 of Annex 1** and hence is not reiterated here**.**

3.1 Connecting multiple experiences to generate novel goal directed behaviors

As has been outlined in the previous section, based on partial cues emerging from the environment (objects in the scene, user goal etc.), the neural episodic memory is able to recall context relevant past experiences in relation to the task at hand. This section presents an interesting example of how multiple isolated experiences (learnt in the past) can be creatively recombined to generate a novel behaviors to realize the goal. **A variation of this scenario is presented in section 6.3: Annex 1, Further section 6.2 Annex 1** *details how the demonstrated behavior of combining multiple memories/remembered experiences to generate novel behavior is independent of which experience is acquired when, during cumulative learning. In the present scenario, the* user puts all four objects (cube, small cylinder, large box and sphere) in front of the robot, and issues the goal to assemble the tallest possible stack using them. iCub has isolated past experiences with all of them. *However, it has never encountered all of them together. This is an interesting scenario because none of the "past experiences" of the robot has the full information to realize the goal (all of them have partial chunks of sequences), but still if the robot is able to "combine" knowledge from multiple experiences to come up with a "novel action sequence" that too without any further learning, it is indeed interesting.* With the help of figure 9 we discuss how multiple past experiences remembered in the context of the present can be recombined to generate novel behavior (without any exploration).

The process initiates with bottom up information coming from the world activating the object hub, generation of "partial cues" that enable recall four past experiences (EM1-EM4), summarized in the figure. All the remembered memories (EM1-EM4) have some information related to a "subset" of objects present in the world. *This summarizes the bottom up process, from objects in the world to remembering past experiences encountered with them (panel **11A**).* The temporal evolution of the top down influence of these competing memories is shown in **11B (the formal details of top down completion between recalled memories is described in full detail in section 4.2.2 of Annex 1)**. Briefly, EM1 and EM2 are completely wiped out in the competition, because there are other memories that encode more information (in the context of the present situation). EM3 encodes information related to not just cylinders and spheres (encoded by EM1 and EM2) *but also about cubes*, hence is a stronger competitor. *But as seen, in addition to EM3, EM4 also manages to stay alive (it encodes experiences with large objects that none of the others encode).* Further, we see also that EM3 and EM4 encode something in common (i.e. cubes), to control which they are basically inhibiting each other (the overlapping neuron is shown in blue box). Importantly, note that in this interesting case, the sum of the activity imposed top down on the hub by EM3 and EM4 is equal to the activity in bottom up object

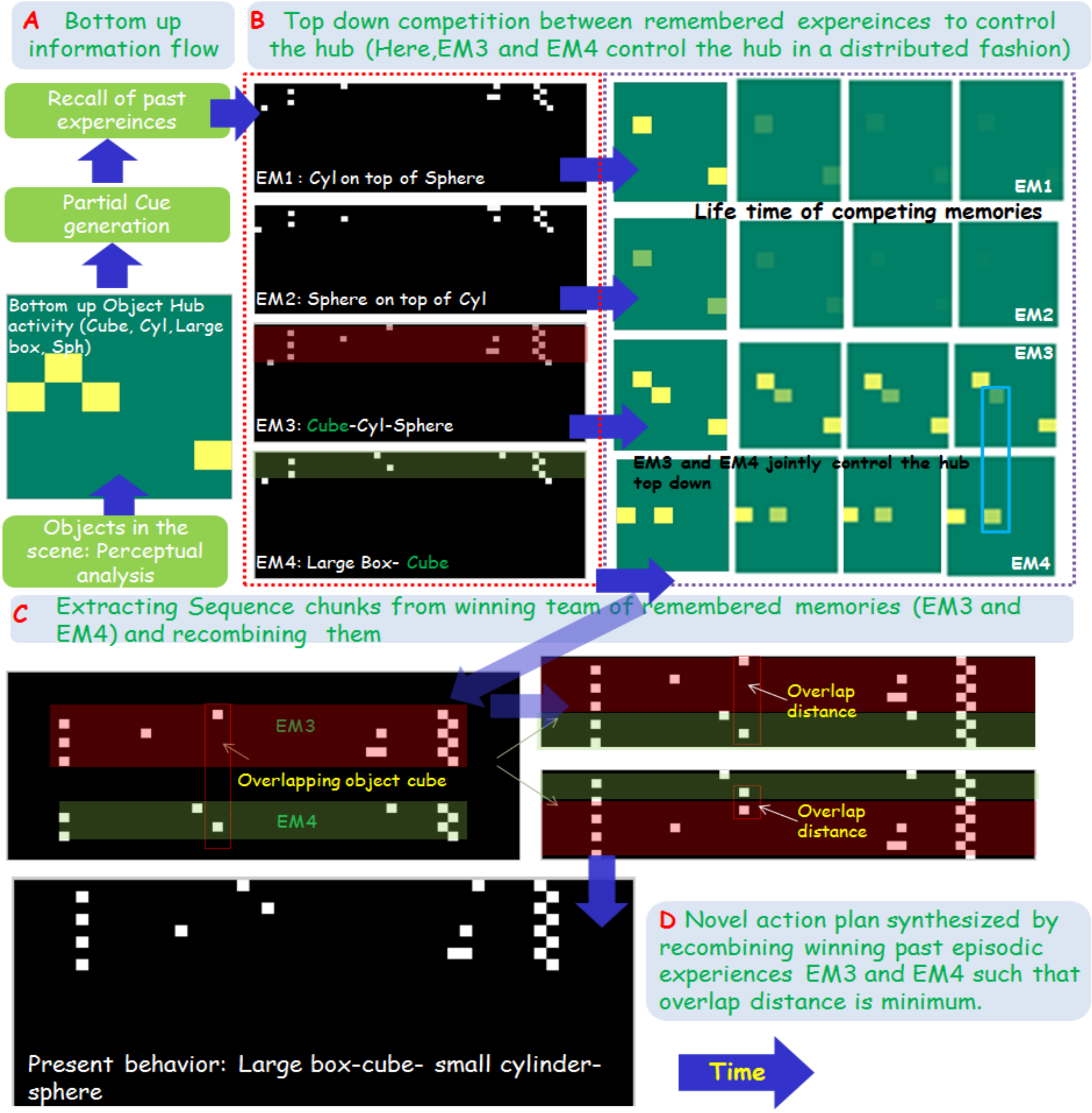


Figure 11: Cube, cylinder, large box and sphere are presented. The robot has had isolated experiences with all of these objects, but none of its past experiences encode the complete solution to build the tallest stack in the present situation. The figure shows how the robot achieves this by recombining its multiple past experiences; the blue arrow shows the temporal evolution of this process. **11A**: Bottom up perception leads to generation of partial cues and the retrieval of relevant past experiences (note all past episodes EM1-EM4 are remembered as they all contain some information relevant to the present context). Remembered episodic memories compete to control the hub top down (temporal evolution of the top down influence of these memories on the hub shown in **11B**). EM1 and EM2 are eliminated, EM3 and EM4 both jointly control the hub partly competing for the common element experiences related to which both of them encode i.e. the cube, shown in a blue box (EM4 exclusively encodes experience related to large objects). Note that the net top down activity of the hub is identical to the bottom up activity, which indirectly implies that the complete solution is available in the isolated past experiences (without the need for exploration). **11C** Action sequence chunks of EM3 and EM4 enter the construction system, with two ways to bind these sequences, the preferred solution is the one in which

overlapping elements of knowledge encoded by different experiences are brought as close as possible (overlaps in this sense playing the role of a sub goal). **11D** shows the final solution i.e. large box-cube-cylinder-sphere with an anticipation of full reward that is given.

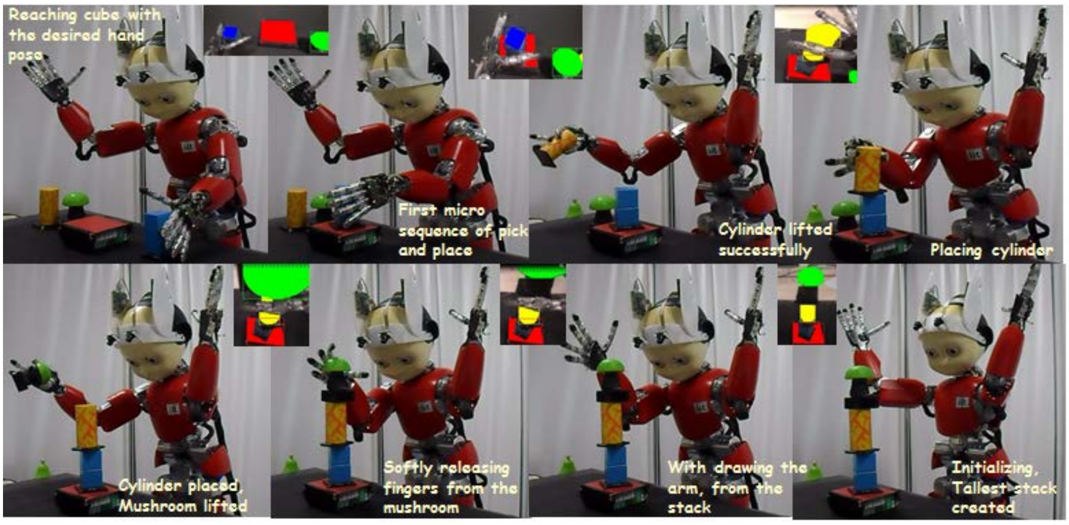


Figure 11E. Novel sequence of actions generated to assemble the tallest stack using four available objects in the scene

hub activation. In our architecture, *this implies that "the complete action sequence to solve the problem is already available in the isolated past experiences that won the competition" and this applies always independent of how many past experiences claim their control over the hub. Either the most valuable action sequence is directly available (in a single episodic memory) or multiple past experiences may have to be combined in a novel fashion to generate a new behavior. In any case, if the net top down hub activity is equivalent to the bottom up hub activity (or equivalently VSSP) then even if the environment is "novel" (like in the present case), the robot can infer that its past experiences contain enough information to realize the goal (by optimally combining these past memories into a novel sequence).* So action sequence chunks encoded by EM3 and EM4 are extracted **11C**, the overlapping object cube highlighted in red box. Overlaps in knowledge between different remembered experiences are advantageous, because it helps to connect them together. The reasoning system just employs one simple rule namely ''construction'' to achieve this:

If there are overlaps in knowledge encoded by different "winning" past experiences, bring them as close as possible. In this sense the overlapping element is similar to an intermediate sub-goal (a point of intersection between two different past experiences)".

As seen in **11C** (right panels), binding the sequence encoded by EM4 before EM3, the overlaps are closest (highlighted by the red box), this is the one enforced by the construction rule. The other alternative is also shown but does not make cognitive sense because we believe that overlaps in knowledge related to past experiences in general play the function of "sub goals" (or points where one chunk of knowledge/memory connect to another). *Combining isolated memories of past experiences, a novel sequence emerges (9D)*: *Stack the large box at the bottom, then the cube, the small cylinder on top of the cube and the sphere on top of the small cylinder and anticipate full reward for this! Indeed full reward is given!*

Finally the construction rule to combine multiple experiences based on minimizing overlaps is inspired from navigation (Ex: consider Munich is an overlapping location to connect Genova to Boston). Interestingly, the Default mode network of the brain involved in recalling past experiences and simulating future consequences is also activated during navigation. The connecting link we believe is that travelling in space and travelling in time (i.e. memories)

might share computational basis. This intriguing issue has been a topic of recent debate (Eichenbaum, 2014).

3.2 Pushing a non-existent object: From recall of past to simulation of future

This subsection presents another interesting scenario where episodic recall of past experience directly facilitates simulation into the future and enables the robot to realize and otherwise unachievable user goal. Panels in figure 4 show the temporal evolution of the behavior of the robot (indicated by the red arrow). In this case, the user issues a goal to **Push the "Fuse Box."** As seen, a fuse and fuse stand is present in the scene and perceived by the robot. *However, the requested object on which the primitive behavior "Push" needs to be executed is not present in the scene thus rendering the user goal not directly realizable.* To realize the goal of Pushing a Non Existent object, iCub must be able to recall a context related past experience, anticipate the consequence of it and initiate real actions to transform the present situation in a way that becomes more conducive towards realization of its goals. As seen in figure 4, though the user goal of pushing the fuse box is not directly realizable, the objects present in the scene are perceived through vision and activate "bottom up" the object hub (two objects fuse and the fuse stand are present). Object hub activations directly generate partial cues for the episodic memory network and trigger recall of past experiences in relation to the context. As seen, from the partial cue, the robot recalls its past experience of picking up the fuse and placing it into the stand, the consequence being the creation of the fuse box. Note that this is directly inferred through the reconstruction of past experience from the partial cue. In other words, recall of the past experience also directly allows simulation into the future allowing the robot to infer that the actions of picking and placing the fuse into the fuse stand will result in creation of the 'fuse box'. In such a modified world, the goal requested by the user is indeed realizable. Thus even though the user goal was to Push the fuse box, the robot picks up the fuse and places it in the stand, thus creating the Fuse box and then executes the Pushing. Thus, exploiting its past experience, the available objects in the present and the anticipation of the resulting consequence of its actions, the robot thought its actions modifies the world appropriately to create the fuse box and realize an otherwise unachievable user goal.

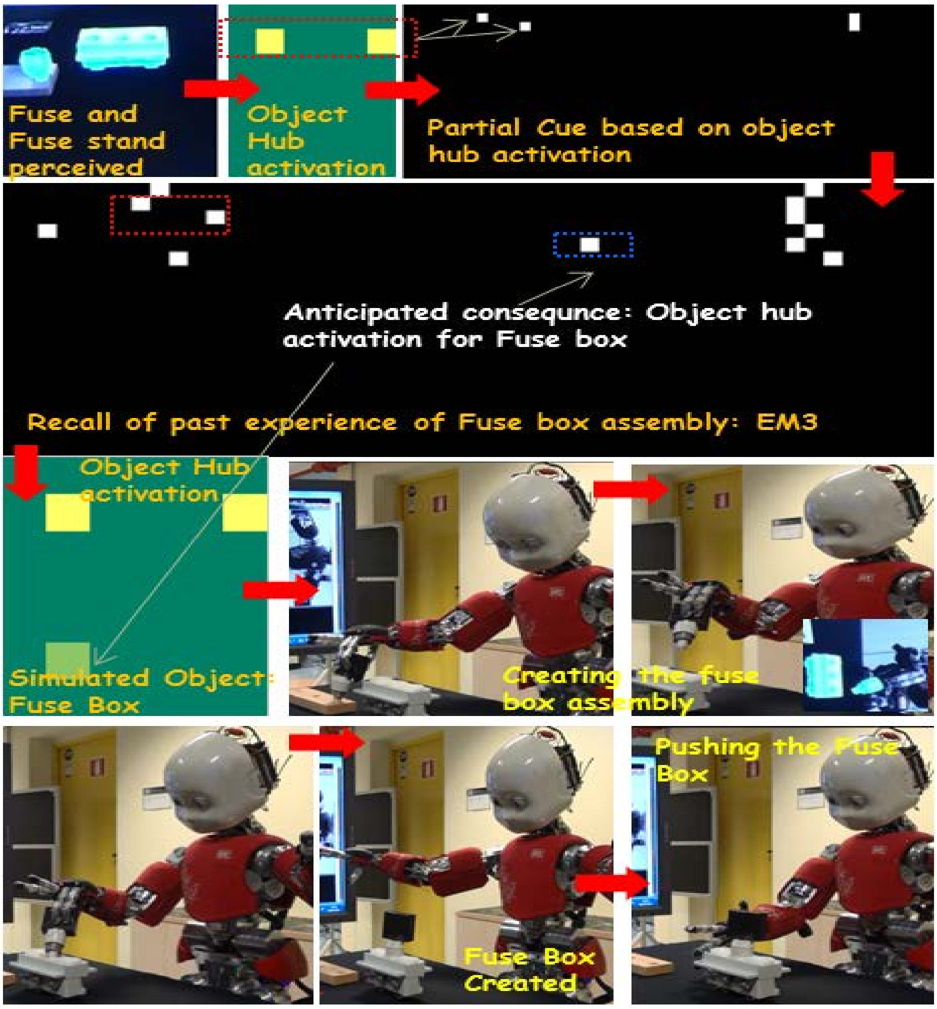


Figure 11. Panels show the temporal evolution of the behavior of the robot (indicated by the red arrow) when the user issues a goal to Push the "Fuse Box": a nonexistent object in the present scene. The panels depict how episodic recall of context relevant past experience also indirectly enables simulation into the future, enabling the robot to initiate actions that "modify the existing world" to make more conducive towards realization of its internal goals. As seen, even though the user goal was to Push the fuse box, the robot picks up the fuse and places it in the stand, thus creating the Fuse box and then executing the Pushing.

3.3 Spatial planning for two Assembler robots operating in a shared workspace

During assembly in a typical unstructured industrial setup, it is inevitable that based on the spatial arrangement of the objects in the scene (i.e. variable), appropriate action sequences must be generated by both robots to operate in parallel (in the shared workspace) and realize as many assemblies as possible (different scenarios are shown in figure 5). In general, to maximize parallelism (while minimizing the need to trigger collision avoidance that increases the assembly time), inevitably requires a flexible spatial reasoning system (on top of the action generation system) that given an environmental scene dynamically allocates sub-goals to the two parallelly operating robots: i.e. planning which object to act on (at different time instances) to facilitate parallel completion of multiple assemblies by both robots. Interestingly, this problem can be connected to foraging and navigation experiments (Toussaint, 2006, Mohan et al 2011) considering the analogy of rats navigating for food with Darwin robots foraging for fuses and fuse boxes. To this effect, as described in section 2.2.3, the peripersonal space for the two robots is learnt through a GNG algorithm that forms the neural substrate on which a range of spatial planning can be achieved. A moving neural field (see, Toussaint, 2006 for formal details of the algorithm), based on anticipated rewards fetched for

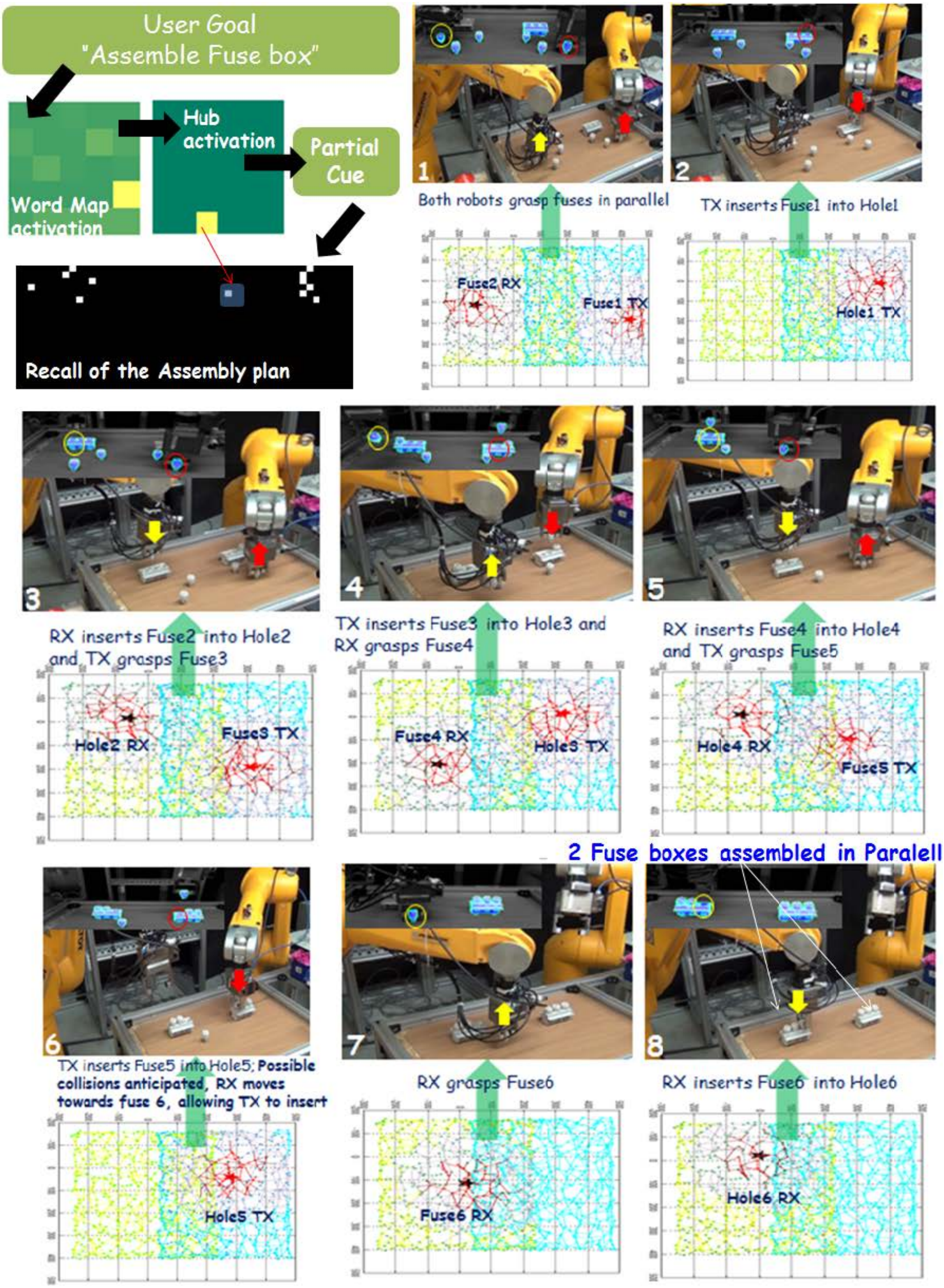


Figure 12. Shows the parallel operation of both industrial robots operating in the shared workspace and performing assembly in a typical unstructured set up. Different panels show how 6 fuses lying at various spatial locations are successfully inserted with both robots operating in parallel in the shared workspace, with basic plan coming from episodic memory, spatial planning taking care of aspects related to allocating sub-goals to the

two robots at different time instances during the evolution of the parallel assembly, and motion planning if necessary taking care of possible anticipated collisions.

choosing an object in the scene organizes the behavior. Figure 12 presents an example of the behavior of the two industrial robots in an unstructured setup (i.e. objects scattered at random locations), where the goal is to jointly assemble the fuse box in the shared workspace (in the present setup two fuse boxes and 6 fuses are present). Note that the planning system itself is agnostic of the number of objects or where they are, but will ensure maximum number of assemblies with both robots working in parallel. As seen, in panel 1, both robots grasp fuses in parallel, then TX inserts and proceeds to grasp the next fuse while RX also inserts the first fuse (Panels 2-3). Note that two fuses are already inserted in parallel based on the spatial arrangement, without the need to trigger the action planning system for collision avoidance. Panels 4-5 show the next sequence of actions with RX grasping another fuse while TX inserting the fuse it grasped in panel 3, and moves to grasp fuse 5. Note that as the neural fields move closer (panel 5), the spatial planning for parallel operation is slowly reaching the limits, hence triggering motion planning to take care of collision avoidance. Hence as seen in panel 6, as TX inserts fuse 5, RX waits and slowly approaches fuse 6. Now no more fuses are remaining, hence TX initializes while RX inserts fuse 6. *In sum, 6 fuses lying at various spatial locations are successfully inserted with both robots operating in parallel in the shared workspace, with basic plan coming from episodic memory, spatial planning taking care of aspects related to allocating sub-goals to the two robots at different time instances during the evolution of the parallel assembly, and motion planning if necessary taking care of possible anticipated collisions.*

3.4 From proto-language to episodic memory: Swift ''switch over" to new Assembly goals

As, illustrated in figure 9, the Darwin architecture is being designed keeping in mind that multiple learning streams (exploration, imitation, proto-language) applied in the context of diverse tasks can be encoded into the neural episodic memory network (and then recalled a based on multiple contexts and exploited to realized present goals). **Particularly, in an industrial setup, the capability to reconfigure basic assembly line operation for new products, with a specified industrial arm in a short duration is a critical desirable feature urgently needing innovative solutions (Reference: US Roadmap for robotics in Manufacturing, 2013).** The Darwin architecture especially the loop between **Observer-Episodic memory-Neural PMP** facilitates this flexibility of swift switchover, with all added advantage of past experiences gained (and already encoded in the episodic memory) seamlessly transferred to the new context. With the help of figure 13, we detail the steps and different kinds of learning needed (detailed procedure described in section 2 and annexes) to achieve this switchover to a new task.

1. **If the robot performing the assembly is new, then the neural PMP forward/inverse model** has to be learnt for the new embodiment. The procedure is outlined in 2.2.2 and section 6 of Annex 5 (user manual). In the present example, the same industrial robots are used; hence this step is not necessary. However the method for training is the same for other any new robot (this is validated by the fact that the framework is operational in both iCub and two industrial robots presently).
2. Assuming that the objects are recognized by vision (trained offline), the next step is to associate linguistic labels to the new objects **to learn associations between word map and shape map (see figure 8) that activate the Object hub** (leading to generation of partial cues, recall of experience if any in the episodic memory). In the preset case, the new objects are called ''big fuse", 'holder' and 'TypeN' (i.e. the new assembled object).

3. Figure 13, Panel A, shows the response when the user asks the robot to **Assemble "typeN".** The word map is activated, leading to activations in the object hub that generates partial cues. ***However, no experience related to the context exists***; hence there is no activations in the neurons of the episodic memory. A null matrix is retuned back to the Observer module. Since this is the Root goal, with no actions executed yet and no experience existing, the system asks user to type "NewAsm" to initiate learning the assembly plan leading to object "typeN".
4. Figure 13, panel B shows the sequence of steps with the user instructing the sequence of actions on the novel objects (learnt in 2) leading to his new assembly request. In this case the sequence is to pick the big fuse and insert it into the holder. This is a simple case, but the present system facilitates also recursive assembly with the same steps (for example, using typeN object and some other new object that leads to a new assembly). ***Every word inputted by the user activates the hubs and the sequence of activations in the hubs is recorded as the present trace. Note that, as described in 2.3, the content of the episodic memory is the sequence of activations in the hubs when experience is gained (taking into account multimodality in the ways experience can be gained).***
5. *To encode this new trace of experience, a weight update in the connectivity matrix of the episodic memory has to be done.* ***Note that all past experiences are retained, while the new one is also encoded (keeping into account cumulative learning of multiple experiences related to multiple tasks).*** *Equation 3, described the learning rule to update (*10^6 synaptic weights in the present network*), while* ***section 7.5: Annex 5****describes the software steps to update the weight files. The process takes less than a minute. Panel C shows the old neural connectivity, the new one after cumulatively learning this new experience and the difference between them (i.e new connections formed in the same network)*
6. Panel D shows the behavior when, after learning, the user repeats step 3 (step 1-2 apply for a new robot or a new object only): i.e. issues the goal to assemble typeN object. The Darwin robots get in action to realize the new goal.

Figure 13. Panel A illustrates step 3, when the user issues a new assembly goal, no experience exists hence a null message is returned to the observer. Panel B panel B shows the sequence of steps with the user instructing the sequence of actions on the novel objects (learnt in 2) leading to the new assembly request. Every word inputted by the user activates the hubs and the sequence of activations in the hubs is recorded as the present trace. Note that, as described in 2.3, the content of the episodic memory is the sequence of activations in the hubs when experience is gained (taking into account multimodality in the ways experience can be gained). Panel C shows the old neural connectivity, the new one after cumulatively learning this new experience and the difference between them (i.e new connections formed in the same network). Note that all past experiences are retained, while the new one is also encoded (keeping into account cumulative learning of multiple experiences related to multiple tasks).

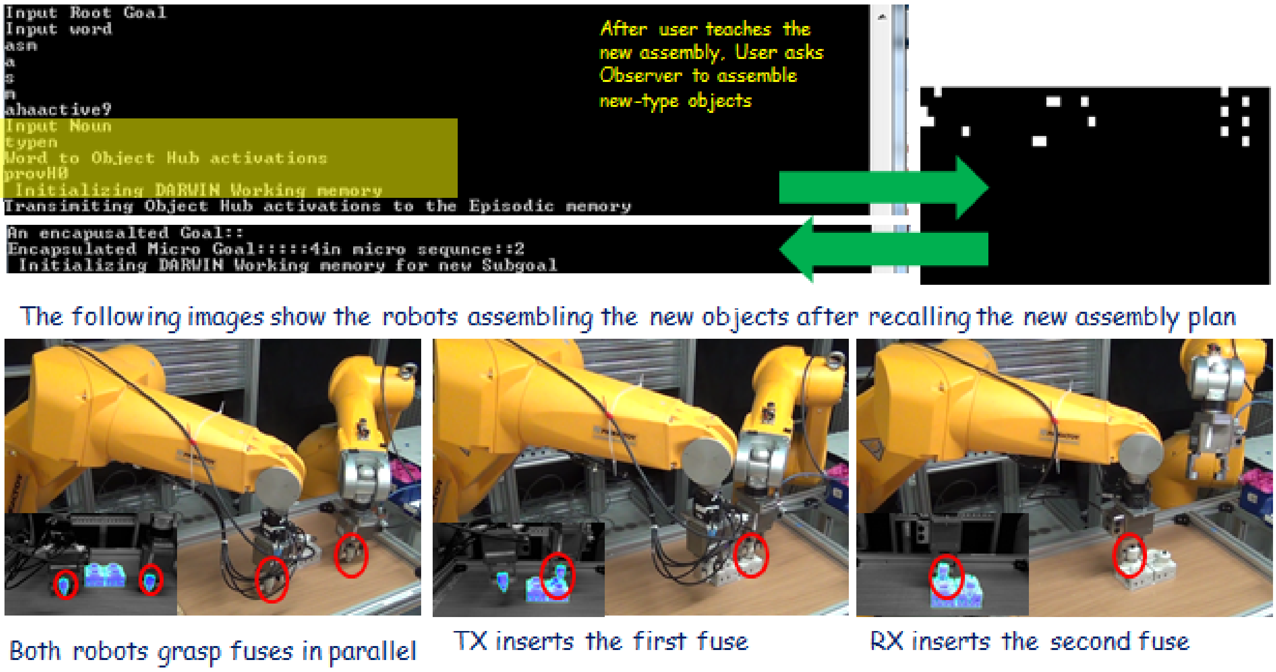


Figure 13. Panel D shows the resulting behavior as a result of the Observer-Episodic memory-PMP loop when the user repeats step 3 and requests the Darwin robots to perform the assembly of TypeN, with the new objects.

3.5. Mimicking the Betty task on iCub: ongoing Y4 scenario

This ongoing task on iCub attempts to further connect different streams of learning existing in the present architecture and in different modules (outlined in section 2) and various inferential capabilities of the present architecture (outlined in section 3) in an interesting novel scenario: that both draws inspiration from studies in animal and infant cognition and putting it in the context of a typical assembly situation.

- The first inspiration from animal behavior is the fascinating tinge of creativity demonstrated by the Caledonian crow Betty (Kacelnick et al, 2002) when faced a nontrivial problem of fetching her dinner basket trapped in a transparent vertical tube (figure 14 right panel shows a pictorial depiction of the situation). A long wire was available nearby. After a short hesitation, she picked up the wire, fashioned a hook out of it using her beak and used the newly created tool to pull out her dinner basket. The behavior demonstrated by Betty involved ***nontrivial "perception-action"*** *and* ***inferring*** *that the goal is directly unrealizable (given the present environment)*, ***recall*** **of "a specific" learnt past experience about a year back,** that was evaluated as ***valuable in the context of the "present"*** (making hooks), some basic knowledge and ***anticipation of physical causality*** (i.e. the consequence of pulling the basket up using the newly created "hook tool" connected to the beak) and ***"flexible" integration*** of all this knowledge in the context of the ***goal***.
- The second inspiration comes from the studies of Andrew Whiten investigating the understanding of physical causality and inferential capabilities in monkeys, namely the "turn disc" task. In this scenario the animals have to learn the consequences of using the new tool i.e. in the simplest case turning the ratchet handle such that the food reward becomes available/reachable (many other variations to this interesting tasks have been conducted in different species).

Note that while these studies present intriguing aspects related to learning, inference and reasoning capabilities (some going into the domain of creativity), the underlying computational basis is still blurred and needs explanation. Thus, we recreate such a scenario in iCub, also keeping in mind the context of Darwin assembly tasks, to validate further the computational framework of the proposed reasoning architecture, at the same time explain how simple but fascinating behaviors captured by studies from animal cognition can be replicated by the underlying computational framework.

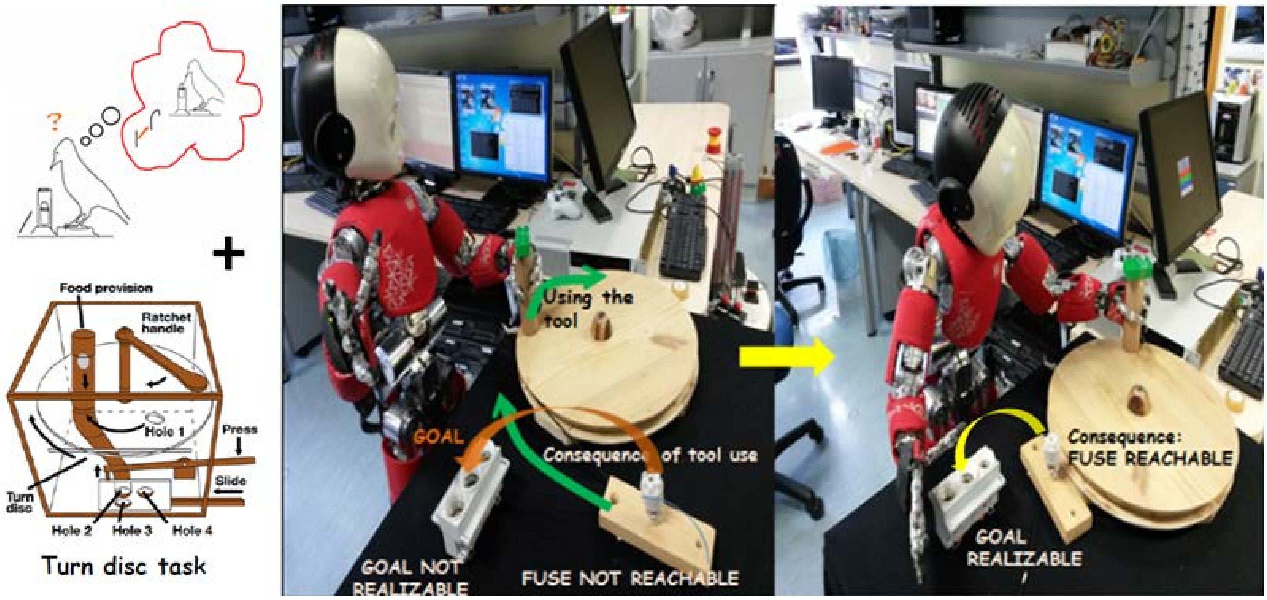


Figure 14. Shows the snapshot of the ongoing Y4 demonstrator on iCub and the underlying inspirations from well-known studies form Animal cognition related to understanding of physical causality and goal directed reasoning.

Figure 14, right panels show the scenario, the challenges we outline in points below.

1. **Existing Past experience**: The robot already has the past experience of assembling fuse boxes and is already encoded in the episodic memory (this includes the type new objects learnt in the previous section).
2. **Imitation:** As another isolated learning experience the teacher demonstrates the use of the turn disc i.e. the tool object (see figure 3). Firstly, the robot imitates the teacher and learns the procedural aspects of coordinating the tool coupled to the end effector (this information supplements the neural PMP, with the new object acting as an extension to the body schema during coordination).
3. **Exploration/Physical Interaction**: Learning the causal consequences of such tool use (an object previously unreachable, becomes reachable).
4. **Meta learning:** learning that the causal consequence of using the turn-disc applies to any object placed on the turn disc tool (of similar size/weight, similar to food provisions: example fuse to assemble the fuse box, Mushrooms, Cylinders to build the tallest stack, irrespective of the goal issued by the user).
5. **Analogy with the Betty Scenario:** As seen in figure 14, middle panel, if the user issues the goal to Assemble the Fuse box, iCub is faced with a similar situation like Betty: Goal not directly realizable (like dinner trapped inside the tube), turn disc available (like the long wire) that affords a potential to transform the present environment in a way that is conducive towards realization of the goal, if past experiences are recalled, consequences anticipated, creatively connected and executed in the context of the present.

To successfully realize the goal, iCub must:

A. **Infer inconsistency** in the execution of the user goal given the present situation (fuse is unreachable), hence assembly not possible: **Neural PMP-Observer functionality**; (Analogy: Betty infers dinner basket cannot be reached).
B. **Recall** the context related learnt experience of the past: i.e. using the turn disc and the ensuing consequences (Observer-Episodic memory link: Observer generates partial cues, episodic memory reconstructs the context related past experience).
C. **Anticipate the causal consequences of the use of the turn disc** that modifies the present environment in a way that is concussive towards realization of the User goal.
D. **Connect Assembly experience with the Turn Disc experience**, to generate novel sequence of actions (Observer-Episodic memory-PMP loop). Analogy: (Inferring that creating a hook and using it allows the dinner basket to be reachable).
E. **Execute the novel sequence of actions** to realize the assembly goal.

In this sense, the scenario captures the essence of the architecture we have developed so far for both Darwin platforms: a) Connecting multiple streams of learning (that are functionally incorporated into different software modules); b) Validated through different tasks that use the same computational model and software architecture (hence not restricting to isolated examples of demonstrations); c) Correlating the developed architecture with emerging trends from neurosciences d) Inversely providing a computational framework that at the same time takes concrete steps to provide a novel architecture enabling different robotic embodiments to use their experience to go beyond experience, in the unstructured world we create an inhabit. Finally, comprehensive discussion on the novel features in the Darwin Reasoning architecture, connecting to emerging trends in neurosciences and insights gained form the computational framework is described in **section 7 of Annex 1.**

Appendix

A.1 Preamble to the Annexures

This deliverable includes 5 annexures, contents of which are linked to the text in the draft, to furnish the reader with further details related to the architecture described. This includes four publications (attached, **called as Annex 1-4**), a user manual (**Annex 5**) describing software aspects of installation of the integrated architecture, with details of steps for learning in the Neural PMP, Observer and Episodic memory modules, link between computational model and implementation of the architecture. Other related publications, already published are referred to as **DP** are not included in the annex, but are available in the Darwin publications page.

The Annexures to Deliverable D4.4 can be downloaded from:
https://www.dropbox.com/s/4xpcvphaobcoxjs/Annexures_D4.4.pdf?dl=0

We summarize the content of each annex below:

Annex 1
Mohan V, Sandini G, Morasso P. (2014). A neural framework for organization and flexible utilization of episodic memory in "cumulatively" learning baby humanoids, **Neural Computation** 26(12), 2692-2734, MIT Press. Doi: 10.1162/NECO_a_00664.

This article provides an in-depth analysis of the Episodic memory framework that forms the heart of the reasoning architecture presented in this deliverable. Multiple aspects related to the computational model like cumulative learning, recall of multiple past experiences based on context, simulation of future consequences and formation of goal directed plans, effects of various parameters on the performance of the system, novelty in the proposed architecture is discussed in detail. ***The results from this article feature the cover of the December 2014 issue of Neural Computation.***

Associated software modules in the Darwin Architecture: *Observer-Episodic memory (see user manual for details)*

Annex 2

Bhat A.A., Mohan V., Rea F., Sandini G., Morasso P. (2014) "Connecting Experiences": Towards a biologically inspired episodic memory for developmental robots". ***IEEE International Conf. on Development and Learning (ICDL-EPIROB 2014)***, October 13-16, Genoa, Italy, 2014.

This conference proceeding reports further advancements and experimental results with the reasoning architecture in the iCub platform.

Associated software modules in the Darwin Architecture: *Observer-Episodic memory*

Annex 3

Mohan V, Morasso, P, Sandini G. (2014). Muscleless motor synergies and actions without movements. In preparation for Trends in Cognitive sciences.

The article presents the rationale behind the Neural PMP based forward/inverse models of action, in relation to other leading approaches in the field. The article was originally written as a letter in response to a recent article in TICS: *Pickering, M,J., Clark, A. (2014). Getting ahead: forward models and their role in cognitive architecture, Trends in cognitive sciences, 18(9)* and is presently being expanded for the longer format at the recommendation of the Editor in Chief.

Associated software modules in the Darwin Architecture: *Neural PMP*

Another Darwin publication provides a comprehensive review of the basic PMP model:

Mohan V and Morasso P (2011) Passive motion paradigm: an alternative to optimal control. Front. Neurorobot. 5:4.

Annex 4

Mohan V., Bhat A.A., Sandini G., Morasso P. (2014) From Object-Action to Property-Action: Learning causally dominant properties through cumulative explorative interactions. *Biologically Inspired Cognitive Architectures*, Vol 10, 42-50. http://dx.doi.org/10.1016/j.bica.2014.11.006.

This article reports further advances related to cumulative learning of causal relations, especially in relation to pushing and anticipating how objects move when forces are exerted on them. The article was also presented at 2014 Intl Conference on Biologically Inspired Cognitive Architectures, MIT, Cambridge, USA, November 7-9, 2014.

Associated software modules: Observer-Neural PMP

Annex 5: User manual for Darwin architecture

The manual describes both the general installation/use integrated Darwin software architecture and provides details related to the core modules in Darwin cognition i.e. the Neural PMP, Observer and Episodic memory (developed by IIT and in relation to this deliverable). Several issues related to learning, link between computational model and implementation in software are described with examples, expected console outputs etc.

A set of other Darwin Publications, referred to as DP in this deliverable and directly related to the developed reasoning architecture, is listed below.

DP1. Mohan V and Morasso P **(2011)** Passive motion paradigm: an alternative to optimal control. Front. Neurorobot. 5:4.

DP2: Mohan V, Morasso P, Sandini G, Kasderidis S **(2013)** Inference through embodied simulation in cognitive robots. Cognitive Computation, *5(3),355-382,* DOI 10.1007/s12559-013-9205-4.

DP3: Mohan V, Morasso P **(2012)** How past experience, imitation and practice can be combined to swiftly learn to use novel "tools": Insights from skill learning experiments with baby humanoids. Intl. Conf on Biomimetic and Biohybrid systems: Living Machines 2012, July 9-12 2012, Barcelona, Spain.

DP4: Mohan V., Morasso P., Sandini G. **(2014)** Towards a Brain-Guided Cognitive Architecture, in "R. Cingolani (ed.), Bioinspired Approaches for Human-Centric Technologies", Chapter 7, Springer International Publishing Switzerland, 2014. DOI 10.1007/978-3-319-04924-3_7

References

1. Addis, D.R. et al. (2007) Remembering the past and imagining the future: common and distinct neural substrates during event construction and elaboration. Neuropsychologia 45, 1363–1377.
2. Addis, D.R., Pan, L., Vu, M.A., Laiser, N., & Schacter, D.L. (2009). Constructive episodic simulation of the future and the past: Distinct subsystems of a core brain network mediate imagining and remembering. Neuropsychologia, 47, 2222-2238. [***PDF***]
3. Addis, D.R. & Schacter, D.L. (2012). The hippocampus and imagining the future: Where do we stand? Frontiers in Human Neuroscience, 5, 173. doi: 10.3389/fnhum.2011.00173.
4. Barabasi, A,L. (2003). Linked: The new science of networks. Perseus Books. ISBN-10: 0738206679.
5. Barabási, A.-L. , Albert, R. (1999) ***Emergence of scaling in random networks***. Science 286, 509–512.
6. Barabási, A.-L. (2012) The network takeover. Nature Physics 8, 14-16 (2012).
7. Buckner, R.L., Andrews-Hanna, J.R., and Schacter, D.L. (2008) The brain's default network: anatomy, function, and relevance to disease. Ann N Y Acad Sci; 1124:1-38. ***[medline abstract]***
8. Buckner, R.L and Carroll, D.C. (2007) Self-projection and the brain. Trends in Cognitive Science; 2:49-57. ***[medline abstract]***.
9. Bressler, S,L., Menon, V. (2010). Large-scale brain networks in cognition: emerging methods and principles. Trends in cognitive sciences ***Volume 14, Issue 6***, June 2010, Pages 277–290
10. Burgess, N. et al. (2002) The human hippocampus and spatial and episodic memory. Neuron 35, 625–641
11. Clark, A. (1997). Being There: Putting Brain, Body and World Together Again, Cambridge, Massachusetts: MIT Press.
12. Dayan, P., Abbott, L. (2001). Theoretical neuroscience. Cambridge, MA:MIT Press.
13. Damasio, A (2010). ***Self Comes to Mind***: Constructing the Conscious Brain, Pantheon.
14. Dudai, Y. (2006) Reconsolidation: the advantage of being refocused. Curr. Opin. Neurobiol. 16, 174–178
15. Eichenbaum, H. (2004) Hippocampus: cognitive processes and neural representations that underlie declarative memory. Neuron 44, 109–120.
16. Eichenbaum H, Cohen NJ. Can we reconcile the declarative memory and spatial navigation views on hippocampal function? Neuron. 2014 Aug 20;83(4):764-70. doi: 10.1016/j.neuron.2014.07.032.
17. Emery, N.J. & Clayton, N.S. (2004). The Mentality of Crows: Convergent Evolution of Intelligence in Corvids and Apes. Science 306: 1903-1907.
18. Fritzke, B. (1995). A growing neural gas network learns topologies. In G. Tesauro, D. Touretzky, & T. Leen (Eds.), Advances in neural information processing systems, 7(pp. 625–632). Cambridge, MA: MIT Press.

19. A roadmap for US robotics (2013), White paper: http://www.cra.org/ccc/files/docs/2013-Robotics-Roadmap.
20. Gallese, V., Sinigaglia, C. (2011). What is so special with Embodied Simulation. Trends in Cognitive Sciences.
21. Grafton, S.T. (2009). Embodied cognition and the simulation of action to understand others. Annals of the New York Academy of Sciences, 1156, 97–117.
22. Hesslow, G. (2002). Conscious thought as a simulation of Behavior and Perception, Trends in Cognitive Sciences, 6:242-247.
23. Hesslow, G., Jirenhed, D.A, (2007). The Inner World of a Simple Robot. Journal of Consciousness Studies, 14:85-96
24. Hagmann, P., Cammoun, L., Gigandet, X., Meuli, R., Honey, C.J., Wedeen, V.J., Sporns, O. (2008) Mapping the structural core of human cerebral cortex. PLoS Biology 6, Kohonen, T. (1995). Self-organizing maps. Berlin: Springer.
25. Hassabis, D., Maguire, E. A. (2011). The construction system of the brain. In Bar, M. (Ed.). Predictions in the Brain: Using Our Past to Generate a Future ( ). New York: Oxford University Press.
26. Hopfield, J,J. (2008). Searching for Memories, Sudoku, Implicit Check Bits, and the Iterative Use of Not-Always-Correct Rapid Neural Computation, Neural computation.
27. Georg Stork., H. (2012). Towards a scientific foundation for engineering Cognitive Systems – A European research agenda, its rationale and perspectives, Biologically inspired cognitive architectures (BICA), Elsevier Science publishers, Vol. 1 (July 2012), pp. 82-91, doi:10.1016/j.bica.2012.04.002
28. Kohonen, T. (1995). Self-organizing maps. Berlin: Springer. 13858-13863.
29. Luc Berthouze. An episodic memory model using spiking neurons. IEEE International Conference on Systems, Man, and Cybernetics, 1:86–91, 2000.
30. Martin, V.C., Schacter, D.L., Corballis, M.C., Addis, D.R. (2011). A role for the hippocampus in encoding future simulations. Proceedings of the National Academy of Sciences USA, 108,
31. Martin A. Circuits in mind: The neural foundations for object concepts. The Cognitive Neurosciences, 4th Edition. M. Gazzaniga (Ed.), MIT Press, 1031-1045, 2009.
32. Martin A. The representation of object concepts in the brain. Annual Review of Psychology. Vol. 58, 25-45, 2007.
33. Mohan V and Morasso P (2011a) Passive motion paradigm: an alternative to optimal control. Front. Neurorobot. 5:4. doi: 10.3389/fnbot.2011.00004.
34. Mohan, V., Morasso, P., Zenzeri, J., Metta, G., Chakravarthy, V,S., Sandini, G. (2011b). Teaching a humanoid robot to draw 'Shapes'. Autonomous Robots, 31(1), pp. 21-53.
35. Mohan, V., Morasso, P., Metta, G., Kasderidis, S. (2011c) The distribution of rewards in growing sensorimotor maps acquired by cognitive robots through exploration. Neurocomputing (June 2011) doi:10.1016/j.neucom.2011.06.009
36. Iriki., A., and Sakura, O. (2008). Neuroscience of primate intellectual evolution: natural selection and passive and intentional niche construction. Philos Trans R Soc Lond B Biol Sci. 363: 2229-2241.
37. Umiltà, M.A., Escola, L., Intskirveli, I., Grammont, F., Rochat, M., Caruana, F., Jezzini, A., Gallese, V., Rizzolatti, G. (2008). When pliers become fingers in the monkey motor system. Proc Natl Acad Sci U S A.105(6):2209-13.
38. Mohan, V., Morasso, P., Metta, G., Kasderidis, S. (2011b) The distribution of rewards in growing sensorimotor maps acquired by cognitive robots through exploration. Neurocomputing (June 2011) doi:10.1016/j.neucom.2011.06.009
39. Toussaint, M. (2006). A sensorimotor map: Modulating Lateral connections for Anticipation and Planning, Neural Computation 18, 1132-1155.
40. Mohan V, Morasso P (2012) How past experience, imitation and practice can be combined to swiftly learn to use novel “tools”: Insights from skill learning experiments with baby humanoids. Intl. Conf on Biomimetic and Biohybrid systems: Living Machines 2012, July 9-12 2012, Barcelona,Spain.

41. Mohan V, Morasso P, Sandini G, Kasderidis S (2013a) Inference through embodied simulation in cognitive robots. Cognitive Computation, Springer science publishers New York, DOI 10.1007/s12559-013-9205-4.
42. Mohan V, Morasso P, Sandini G (2013c). On the interplay between learning memory prospection and abstraction in a cumulatively developing system. IJCNN 2013, Dallas Texas, August 2013.
43. Murata, L. Fadiga, L. Fogassi, V. Gallese, V. Raos, and G. Rizzolatti. Object representation in the ventral premotor cortex (area f5) of the monkey. Journal of Neurophysiology, 78:2226–2230, 1997.
44. Meyer K, Damasio A. (2009) Convergence and divergence in a neural architecture for recognition and memory. Trends in Neuroscience. Jul;32(7):376-82.
45. Maguire, E.A. (2001) Neuroimaging studies of autobiographical event memory. Philos. Trans. R. Soc. Lond. B Biol. Sci. 356, 1441–1451.
46. Patterson, K., Nestor, P.J. & Rogers, T.T. (2007) Where do you known what you know? The representation of semantic knowledge in the human brain, Nature Reviews Neuroscience, 8(12), 976-987.
47. Raichle, M.E. et al. (2001) A default mode of brain function. Proc. Natl. Acad. Sci. U. S. A. 98, 676–682.
48. Schacter, D.L.(2012). Constructive memory: Past and future. Dialogues in Clinical Neuroscience. 14, 7-18. [PDF]
49. Schacter,D,L., Addis,D,R., Hassabis,D., Martin,V., Nathan,R.S, Szpunar,K.K (2012). **The Future of Memory: Remembering, Imagining, and the Brain**., Neuron Vol 76, Issue 4, Nov 2012.
50. Sporns, O., Honey, C.J., and Kötter, R. (2007) Identification and Classification of Hubs in Brain Networks. PLoS ONE 2, e1049.
51. Sporns, O., and Kötter, R. (2004) Motifs in brain networks. PLoS Biology 2, 1910-1918.
52. Sporns O (2010) Networks of the Brain. MIT Press.
53. Suddendorf, T., Addis, D. R., & Corballis, M. C. (2009). Mental time travel and the shaping of the human mind.Philosophical Transactions of the Royal Society of London B: Biological Sciences, 364, 1317-1324.http://dx.doi.org/10.1098/rstb.2008.0301
54. Suddendorf, T. (2013). Mental time travel: Continuities and Discontinuities. *Trends in Cognitive Sciences, 17*, 151-152.
55. Tulving, E. (1972) Episodic and semantic memory. In Organisation of Memory (Tulving, E. and Donaldson, W., eds), pp. 381–403, Academic Press
56. Weir, A.A.S., Chappell, J., Kacelnik, A. (2002). Shaping of Hooks in New Caledonian Crows. Science 297:981-3.
57. Whiten, A., McGuigan, N., Marshall-Pescini, S. & Hopper, L. M. (2009). Emulation, imitation, overimitation and the scope of culture for child and chimpanzee. Philosophical Transactions of the Royal Society B: Biological Sciences. 364, p. 2417-2428.