The Janus Architecture for a Robot Brain

F.J. Śmieja smieja@gmdzi.uucp

H. Mühlenbein muehlen@gmdzi.uucp

German National Research Centre for Computer Science (GMD),
Schloß Birlinghoven, 5205 St. Augustin 1,
Germany.

December 13, 1990

Abstract

In the evolution of the mammalian brain the two-sided nature of the neural motor control and sensory input has been preserved in the architecture of the cerebral matter. We believe this fundamental two-sided nature of the mammalian brain to hold a very important clue to the problem-solving capability of the higher mammals. The idea of a two-hemisphere architecture on the macro-level and modules of neural networks on the micro-level, with conflicts between the two, as observed in Nature, provides a basis for the next generation of artificial problem solvers, exemplified by our Janus architecture. This architecture consists of two halves that can independently process the same data and generate decisions about a change of the robot's state. The halves are connected by simple processing channels through which they can exchange information at various levels. The brain consists of neural networks at its lowest level, and so is adaptive, but the two sides process and learn about the environment in subtly different ways. The Janus architecture is described here in a top-down modular fashion, leading to the presentation of the model for a simple prototype system, Janus I, which is to be developed through simulation. Janus I is a first step into "reflective" architectures, whereby the system internally reflects about the capabilities and reliability of its modules.

1 Motivation

One of the higher-level aims of neural network research is to develop systems that will be able to perform a wide range of tasks, and survive in a changing and perhaps hostile environment. Most neural network research to date has been involved in developing low-level models of network architectures and dynamics, as well as learning algorithms for the same. Each of the individual networks alone cannot however satisfy all the requirements of an adaptive and at least partially autonomous system. It has been shown on a number of occasions that the fundamental scaling properties of such networks make the quest for a scalable general-function net rather naïve [SM90, MP88a, Müh90, TJ88]. What is required to solve large, complicated tasks is not large, complicated networks but a large number of small network modules, each of which may be specialized in the performance of a particular aspect of the complicated task, with perhaps suggestions made by these modules to be combined further up the hierarchy of the system, in order to produce a measured response.

The actual details of the mechanism of such a system, and its architecture, is a matter for investigation, and not all hierarchical structures will necessarily produce desired intelligent behaviour—indeed this depends very much on the environmental constraints and the network constraints in the form of the types of functionality available from the modules. The call for a step in the direction of the development of modular systems has frequently been made [Sel59, Min85, MP88b, Ede87, Mel88], but up till now most of the emphasis has been on developing small modular systems specialized as a whole for a particular task domain [Wai89, NKK90, SNA89, Ebe89] rather than systems capable, merely through a higher-level alteration in the specification of the architecture or environment interaction, of being used in a wide range of applications. This call is echoed in this research, with the introduction of a fundamentally different architecture for the brain of a robot.

2 Introduction to the ideas of Janus

This paper is concerned primarily with a description of the fundamental properties of the architecture and its prototype, rather than with a justification, be it technical, philosophical or biological, of its existence or desirability. Indeed, one of the primary goals of this work is to determine the properties and advantages of such a brain architecture, rather than derive it from a desired functionality.

In this paper we describe the Janus architecture in a top-down modular fashion, starting with a general heuristic for a problem-solving brain, based on simple observations of the two sides of the human brain, leading to a presentation of the specific Janus I model for a simple prototype system. The choice of the architecture constraints was directly influenced by simple observations of the top-level structure and behaviour of the human brain [Ecc79, SD85]. However, the similarities end at that level and a great deal of freedom is permitted in lower-level development, within the framework of the above constraints.

The emphasis at the initial stages should be not so much on exploiting the new architecture so as to attack more difficult tasks, but rather in developing it in such a way that simple but non-trivial tasks may be performed using an architecture that can be built upon and be applied to the harder and more interesting tasks. With the harder tasks the contribution from the use of two sides of different processing nature may be better assessed.

A Janus brain controls a robot that exists in a changing environment, and that can be affected by the environment and likewise can affect the environment itself. The robot can possess various sensory attributes in order to perceive the environment in which it exists. It can be affected by the environment either directly, through physical contact with stationary barriers or collisions with moving objects, or indirectly in the thought and learning processes within the brain. The robot itself can affect (its perception of) the environment either non-physically through controlling the sensory inputs or directly through effecting physical alterations in the environment.

The highest-level description of the Janus architecture is illustrated in figure 1. The sensory representation of the environment provides the input to the system, and the output from the system is in the form of motor connection to parts of the robot which can change its or the environment's state. At this level we find it expedient too to introduce an input representing "reinforcement"—an immediate low-level signal which does not require analysis by the brain. The brain is divided into two halves—left and right. All sensory and reinforcement signals

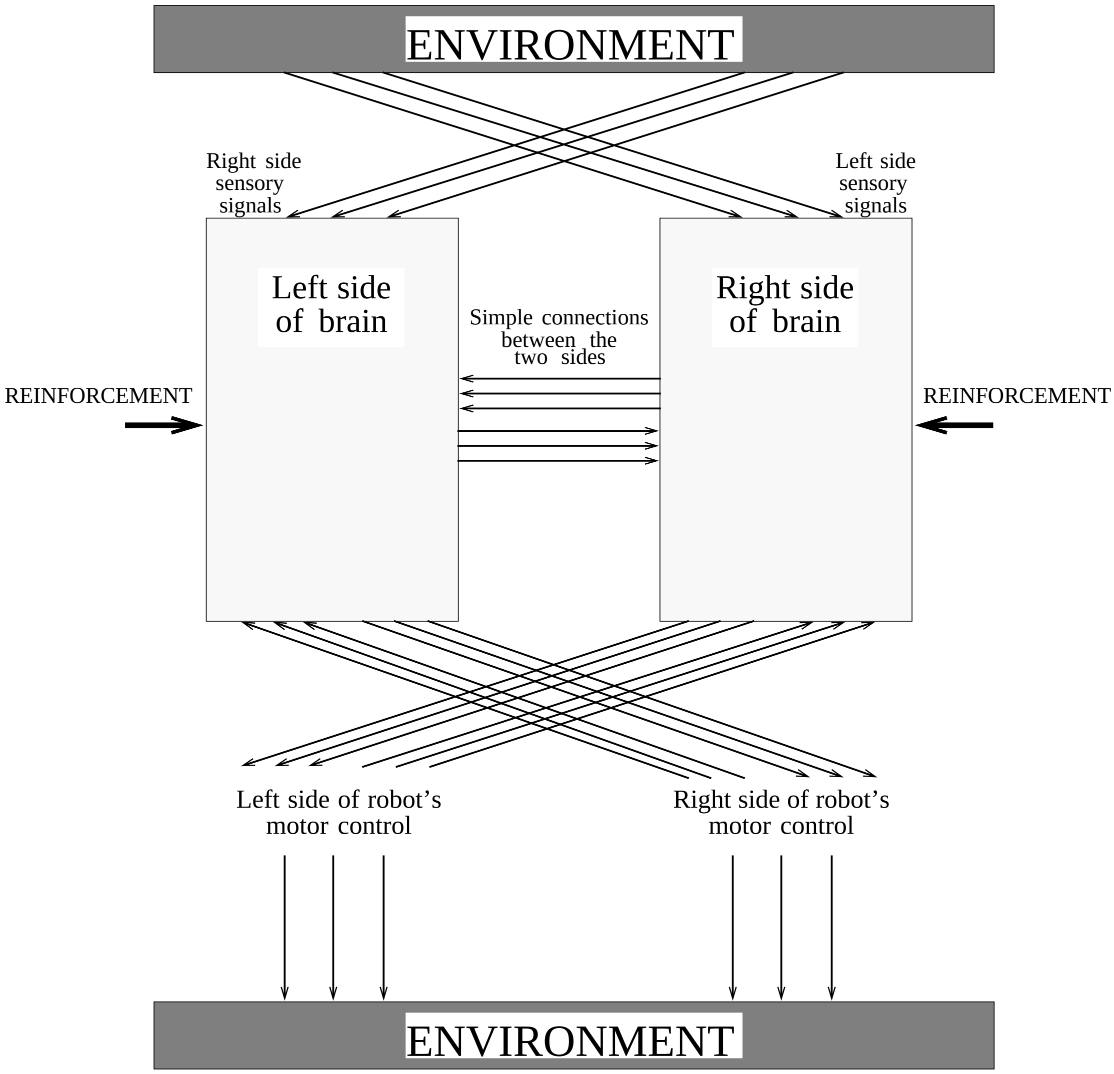

Figure 1: Highest-level representation of the Janus architecture.

from the left side of the robot pass directly to the right half of the brain, while those from the right side of the robot pass directly to the left half of the brain. The left half of the brain controls directly the motor outputs controlling the right side of the robot, and similarly the right half controls the left motor side.

There exist simple connections between the two sides, through which information can be exchanged. Thus it is possible for both sides to receive sensory information about the whole environment and robot's body. Since the sides can both operate independently and process the same information differently, some mechanism must exist for reconciling two opposing motor decisions. This also entails use of the connections between the two sides.

Finally, an approximate symmetry is imposed between the halves, with regard to the modular structure (to be described below) they possess.

In addition to these constraints, an important aspect of the Janus architecture is the difference in the processing and learning of the data itself on the left and right sides.

3 Difference between left and right processing

There exists a difference in the method of processing information on the two sides of the brain, but this difference, the partial existence of which forms a formal constraint of the architecture, may only be visible at an intermediate level of the brain operation.

The same inputs are accessible to both sides of the brain and the data is analysed by modules of neural networks of which there are an abundance of all types on each side. Thus the same *information* is available to each side, but different *aspects* or global/local *representations* are considered by each side. Likewise, in learning about the environment different aspects are considered, and in such a way two different interpretations of the environment may develop. Such differences were inspired through observation of the many split-brain experiments [SD85] and the evidence they provide, in a nutshell, for the greater degree of analytical interpretation of events on the left side, and more of a non-critical passive acceptance and modelling of the environment in a synthetical way on the right side. We hope to include such differences in our model eventually in a general way through imposing greater *self-supervision* on the left side [Ecc79]. In the initial stages such differences will have to be more specific.

Since the sides can both operate independently and process the same information differently, a mechanism exists for reconciling two opposing motor decisions, based on a dominance–recessiveness principle. This also entails use of the connections between the two sides. We view the concept of conflict between sides as an important ingredient that will later highlight the desirability of Janus architectures.

Thus, one could in a generalizing way say that the right side models the environment and its effects through unconscious associations, without trying to rationalize them, while the left side models the environment through classification and categorization of the properties of objects and events in a more analytical manner. The opposing processing characteristics on the two sides create conflicts that will allow a higher level of system behaviour.

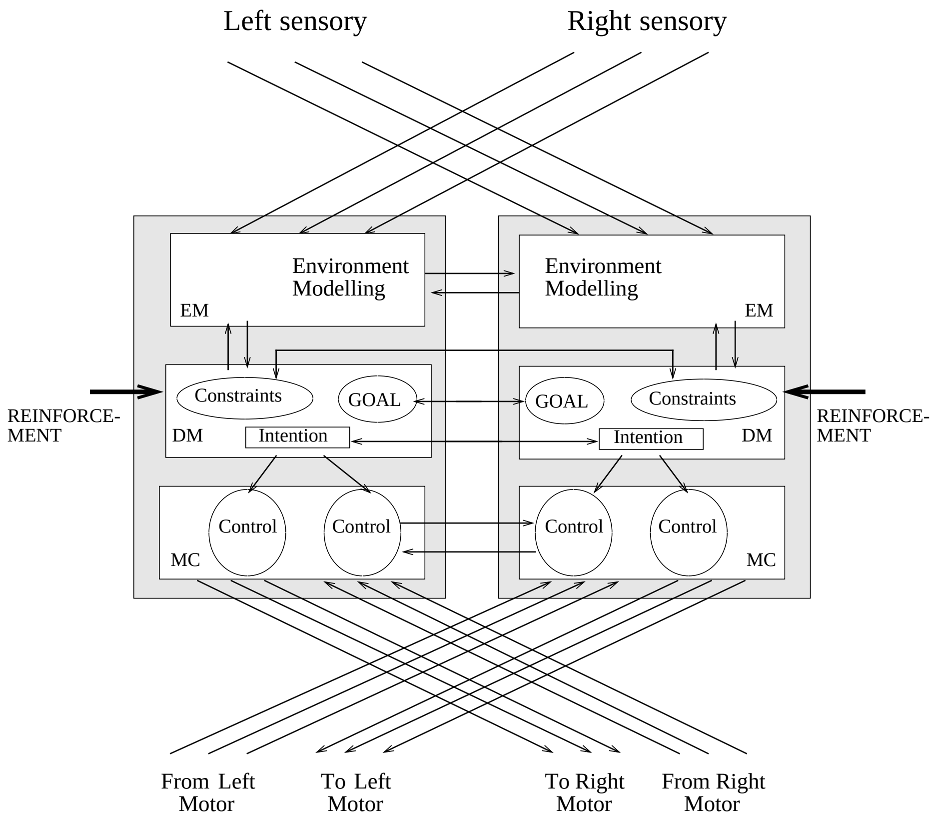

Figure 2: Block-level description of the Janus architecture, showing the three main areas of the brain, and their interactions.

4 Detailed description of the Janus architecture

The next level down in the description of a Janus system concerns the three processing blocks on each side. The first is the *environment modelling* block (EM). Here all the sensory inputs are received and processed, objects and events are classified, and future states of the environment can be predicted using past states. Such skills are learnt from observation of the environment and causes and effects. This block represents the understanding acquired by each side about the environment.

The next block is concerned with *decision making* (DM). Input comprises an internally generated objective or goal (constrained to be the same on both sides), the current state of the environment and the system (which may be channelled from the last block or transmitted independently via a preprocessing mechanism directly from the sensory inputs), and a specification of the constraints on the system's capabilities and the hindrances within the environment, or externally defined rules the system must obey. Output consists of an *intention* to be fulfilled by the next block.

The third block is the *motor-control* centre (MC). It translates intentions from DM into signals intended to achieve this, that pass out of the brain to the attributes of the system producing environment interaction via the use of various appendages. Further inputs to this block come from the attributes themselves, informing the brain of their current state. The overall block-level description is illustrated in figure 2.

Within each block are a set of *modules* (of variable complexity—this is a generic name) concerned with different aspects of the functionality of the block. EM might receive data from visual and tactile senses. The areas in the block concerned with low-level processing of infor-

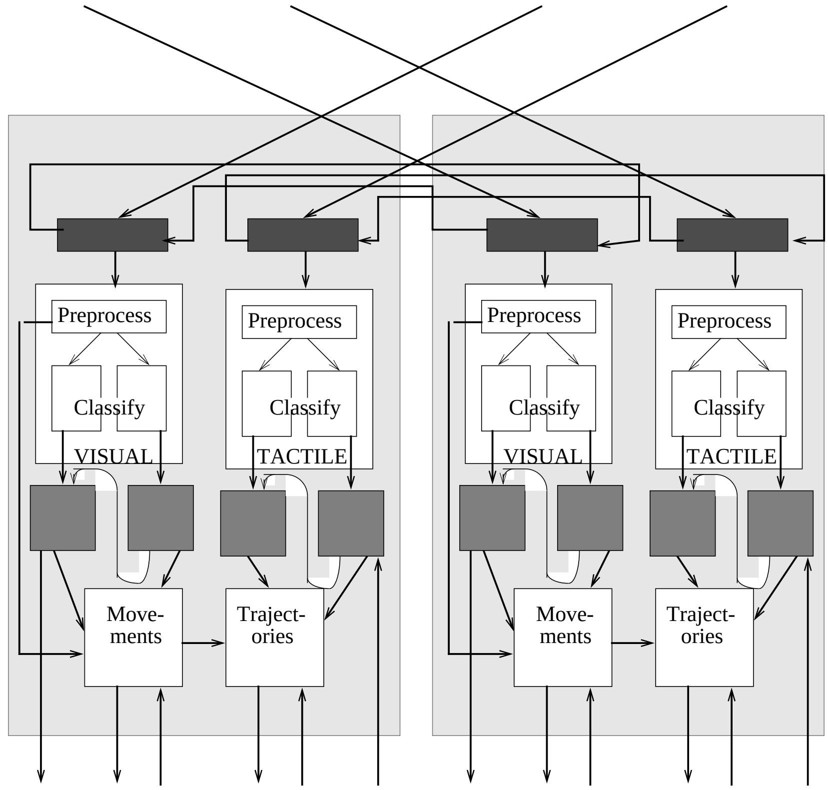

Figure 3: Environment modelling (EM) block of the Janus architecture, showing modules and information transfer between the two sides.

mation of each sensory type each have preprocessing modules to build up the image into sets of fundamental shapes. From here inputs could be provided to object classification/recognition modules, which could further supply inputs for the classification of more complex objects composed of sets of the basic objects, as in Pandemonium I [Sel59]. The inputs and outputs of such modules are not however constrained to follow this specification nor is a module prevented *a priori* from receiving input from any other module (although one would in general desire a downward flow of information). In addition to classifying static objects, this block is equipped with an appreciation of temporal events. Thus one might also find modules concerned with detection of motion and then prediction of trajectories followed by objects. In short then, the modules work together to model the environment using sensory information. At this block too, the connection between the two sides of the brain allow unprocessed sensory information to be exchanged, so that both sides might act on the same data. The EM part of the brain is shown in figure 3.

There are various mechanisms one might envisage for the DM block. The constraints are that it employ a set of learnable strategies in the analysis of current system/environment/goal states, and that it evaluate the appropriateness of a suggested decision using knowledge of the environment in EM, so that it might "look ahead" a certain number of steps, and predict consequences of its intended actions. DM must also receive information as to the success or failure of a strategy/series of moves. This can take the form of progress monitoring in attaining a goal or time spent searching etc. DM should have the capability of changing a strategy (defined perhaps as a series of sub-goals) before the goal has been attained, if it no longer is the optimal one (due to changes in the environment etc.). A simple connection must exist between left and right halves in order that the intentional output is the same on both sides. To represent such a mechanism we postulate a "Master" that receives the suggestions from each side and, depending on the current state of dominance of either side (or some other

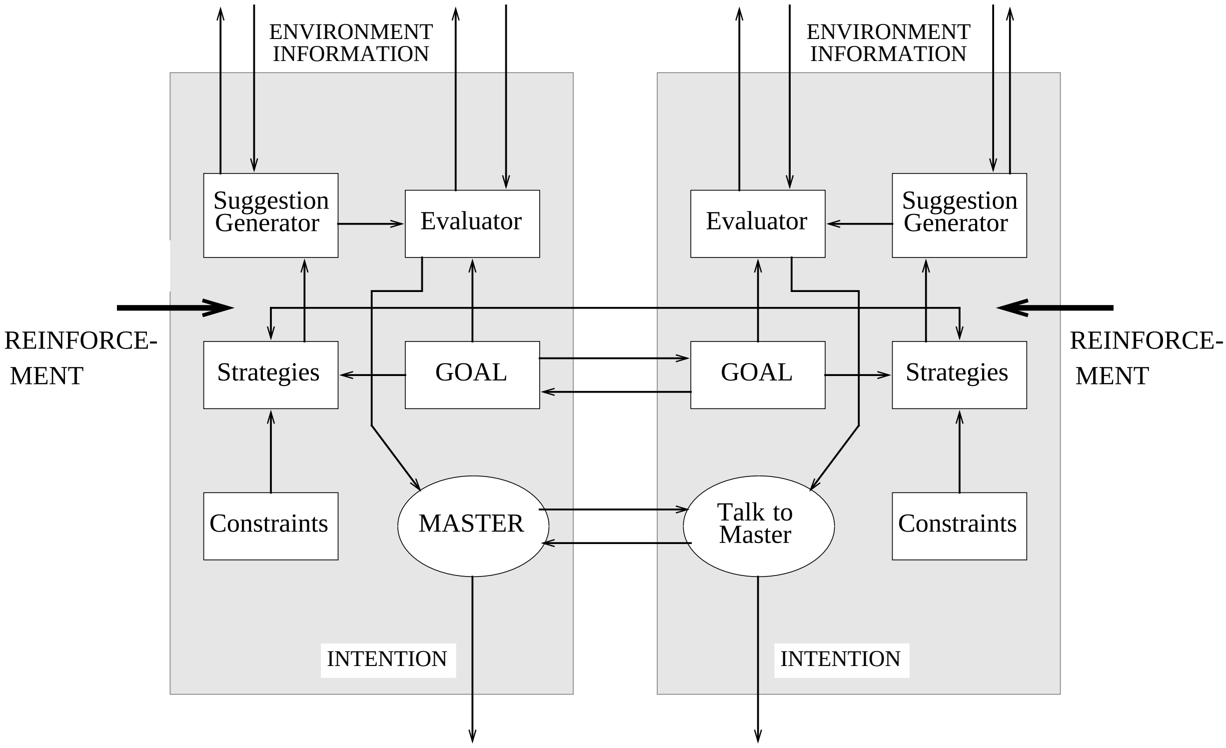

Figure 4: Decision making (DM) block of the brain, showing modules and possible interconnections with EM.

comparison procedure) replies with the intention that must be used and passed on on both sides. There are also connections between the sides to equalize the goal understood on both sides, and a connection may also be used to transmit information about the strategies used and learnt (see below). The decision making blocks are shown in figure 4.

The third block also consists of a series of modules. They are each responsible for effecting the intention from the DM by moving limbs, firing motors, moving eyes, etc. A typical module might receive input from DM specifying decision of motion, and the muscle being controlled informing it of the current situation. There exists a coupling between respective modules concerned with corresponding attributes on either side of the robot, so that a coordinated movement may be learnt. Thus a module may be informed of its neighbour's state and corresponding modules may learn to work together to achieve a coordinated movement. In order to enable learning, general signals denoting the success or failure of a movement are available from a type of reinforcement mechanism, which may in fact indicate the robot has crashed or fallen due to incorrect coordination (figure 5).

The modularity of a Janus architecture permits almost independent development of its various blocks, and modules within the blocks. Modules may be added gradually, to allow more complex behaviour to arise from each of the blocks, and the system may grow within the framework and constraints of figure 2.

5 The Janus I brain

Janus I is a first simple exploratory attempt at building a system with a Janus architecture. Janus I exists in an environment occupied by at least one other object, and which is a two

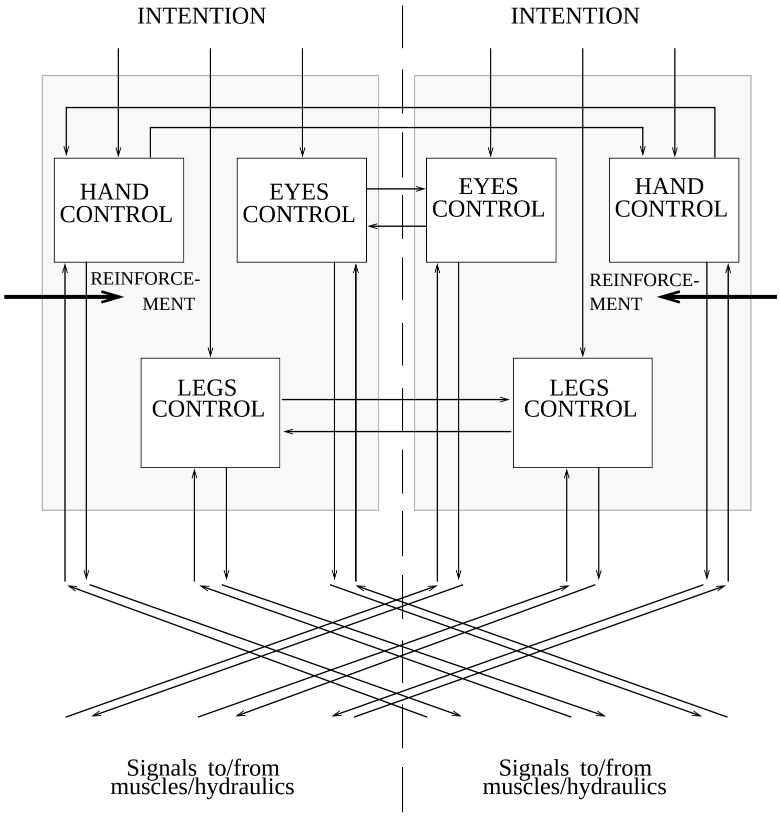

Figure 5: Motor control (MC) block of the brain. The modules control different physical appendages which may affect the environment, and must coordinate on the two sides.

dimensional space. The environment modelling block comprises only visual information, from which is derived the recognition of simple one dimensional objects and movement. At the next level a module is involved with learning the motion of such objects. In the DM block the goal consists of catching a particular object in the least number of steps. There exists a strategy learning module and a module that evaluates suggestions using look-ahead from the motion-learning modules above and by generating new imaginary environment states, as was done in Pandemonium II [Śmi88, MK89]. The decided intention is passed to the MC block which contains a module (or set of modules [Śmi90]) involved with coordinating leg movements to produce a walking action. Figure 6 shows all these parts together, and their mutual connections. The blocks are described in more detail below. The task of the robot is to catch an object in the environment. This object may move and the robot is required to learn this movement. A further robot capacity is to *assess* the strategy, working on the reinforcement it receives about the length of time taken to reach the object. It does this by storing it together with the *classification of the movement* of the object in a neural module. Thus the robot makes decisions and chooses strategies based on past experience, although it still has the capacity to choose new strategies when the familiar ones do not work. In the first implementation the strategies are simply the number of look-ahead steps considered by the robot, in choosing the best next step.

5.1 Object recognition/Motion prediction

The recognition of a one dimensional object may simply be represented as the classification of its frequency spectrum—which may also be simple, but may vary in amplitude according to its distance and in general through noise. Thus this module may also be used as a proximity of goal indicator. Such learning as this can be initiated by any pattern classification type of

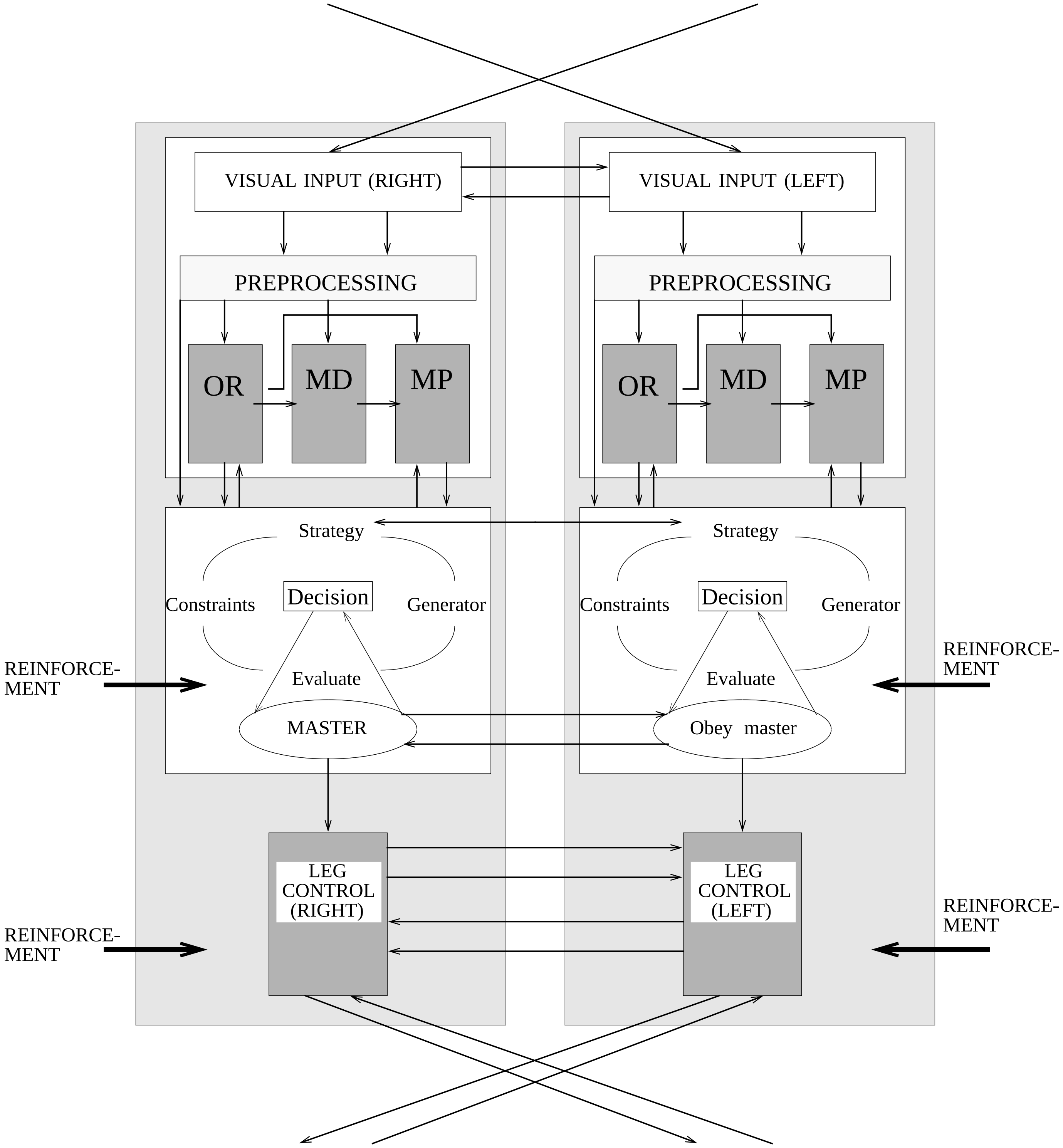

Figure 6: The specific structure of the blocks for Janus I, which possesses a Janus architecture. The details of the modules, and their specific functions, can be developed top-down.

neural network (OR). At this stage the object motion detector may be assumed and is just indicated in figure 6 by a black-box module MD. Motion predictors (MP) are different on the two sides of the brain. On the left the motion of an object is *analysed*, in the same vein as in Pandemonium II [Śmi88, MK89]. That is, the last p displacements and changes of angle of a moving object are associated with the next, $(p+1)^{\text{th}}$, displacement and change of angle, using a neural pattern associator. On the right the trajectories of objects are learnt through mimicking, using recurrent net learning [Ghe89, Pea88, WZ88]. Thus motion is not predicted through analysis of preceeding steps, but rather through continuation of trajectories using contextual information stored temporally. It may be noted that the right side may well be better at predicting smooth motion than the left side, which will itself be good at suggesting alternatives when a motion is irregular. Furthermore, the right side may have more success in distinguishing motions which may be similar in some parts but not in others.

5.2 Decision making

The system has to make at each step a decision as to which direction it wishes to move in. The goal is, as stated above, to reach the prey. The constraints on the motion of the system may be for example that it can only move in smooth curves, rather than in any direction, or, if it has a variable speed, that it must "go through the gears". There may also be environment constraints, such as obstacles or hostile objects. In fact the complexity of this type of problem may be varied very easily.

Given the constraints and the current state of system/prey in the environment, and preceeding environment frames, the DM makes a suggestion as to a suitable next move. The actual details of the decision mechanism is a subject for research, and no specific rules are laid down here, but in order to get a fundamentally different behaviour with respect to decision making on left and right sides, it is suggested that strategy formation and learning should differ. On the left side formation of a strategy involves the piecewise fusion of sub-goals. Each sub-goal is assessed and according to a reinforcement signal of some kind its worth is revised. The same goes for the assessment of combinations of sub-goals. Conversely, on the right side a strategy consists of the consideration just of the goal, and no sub-goals. The worth of the strategy is judged as a whole and learnt as a whole. It might also be suggested that complete strategies initially be generated only on the left, and when constructed be learnt on the right (but as a whole rather than in parts as on the left).

5.3 Learning to walk

Janus I is given two legs and not a set of wheels for a very specific reason. Each side controls a leg and only through cooperation and coordination of the sides will it be possible for the whole to move (note this does not mean the sides' autonomy is diminished). The robot is supported by two legs, and unless the leg movements are coordinated the robot will either loose balance and fall or even perform a random walk. Thus the robot has to learn not only to produce leg movements which move it in a particular direction, but also those that are possible. In walking, one foot is always in contact with the ground, and this leg acts as a pivot. Thus one cannot walk either with both feet on the ground or with both feet off the ground. Nor is it stable for both legs to head in different directions.

As a simple starting point, Janus I may have a very basic and highly constrained walking

Initial state	Stable movements
(S,S)	(F1,S), (L1,S), (R1,S), (S,F1), (S,R1), (S,L1), (S,S)
(F1,S)	(S,F1), (S,F2), (S,R2)
(S,F1)	(F1,S), (F2,S), (L2,S)

Table 1: Possible stable leg movements, given a starting position. A number indicates the length of the stride.

system, so that it might have more chance of getting it right (more complex locomotion systems may of course be built up easily around such a module, or replacing it). The robot can always balance on its feet (or pods). Either leg is constrained to move in one of the three directions left (L) right (R) and forwards (F). Left and right are defined by an angle to the forward direction, and may also be changed adaptively by future modules. The stable end positions of a walking motion are both legs together (SS), and one leg one stride ahead of the other (FS or SF). S indicates standing. After each movement the robot is defined to be facing in the direction it has just stepped, thus, for example, the end points of a motion involving turning left with the left leg while the right leg stands still will be (FS). A leg can move a distance of one or two strides, where a stride is defined as the maximum distance between two legs (Note: in order to effect speed changes, one may alter the length of the stride using future modules, for example). A two stride motion allows a leg to change from being behind the other leg to being in front.

Table 1 shows the only stable leg movements, given the three possible starting positions (numbers indicate length of movement). It may be useful to allow the robot to turn in any direction once it is in the state (SS), but we do not consider this here. One can imagine various levels of complexity/sophistication of such a leg control system. We emphasize here the desirability of starting off with a simple system that is easily defined and results in tight constraints on robot motion. Later on one can add sophistication, once the basic principles of a coordinated motion have been established.

The subtlety of such coordination may be more evident when one considers that not only is an intention and the present state of both legs necessary for deciding of the motion of a particular leg, but most important is the decision for the other leg. In figure 7 a simple adaptive associative net is suggested as a means to learning the above simple coordination. As inputs each side receives the current state of its own leg and the other leg via the inter-side channels, and the intention. The outputs consist of one of four possible suggested motions for both legs. At these outputs a winner-takes-all procedure is initiated for each leg movement [Mal73, Gro76, RZ85], and at the same time excitatory connections channel the outputs of the left leg of the left side to the outputs of the left leg of the right side, and similarly for the right leg outputs. The competition will cease when both sides agree on leg movements. The correctness of these movements at all of course are learnt by each net separately, not employing any winner-takes-all process at the outputs during learning.

6 Conclusion

Janus is an attempt at modelling biological brains not at the lowest neuronal level, but at the module level. The assumption is that complex problem-solving strategies may better arise

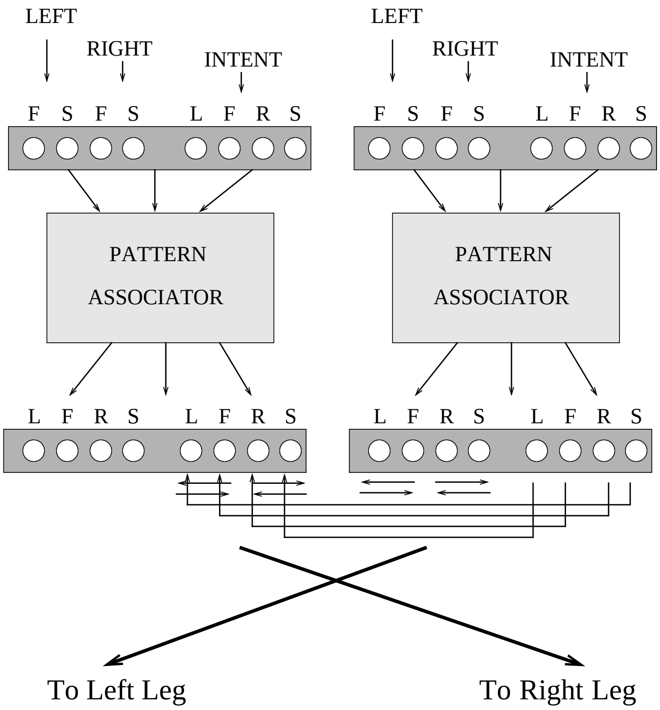

Figure 7: A suggestion for a simple adaptive pattern classification pair of nets to control the walking motion. A winner-takes-all process proceeds in the feed-forward mode at the outputs for the leg not directly controlled, in order to reach an agreement between the sides (only one set of couplings shown).

through harmonizing many conflicting actions and sub-strategies. In Janus the system uses actively the interaction between more analytical problem solving and more synthetical problem solving. The system tries to improve continuously using its "reflective" architecture. It is intended that each important module has associated a monitor module that estimates and alters the capabilities of the important module.

The Janus implementation and simulation is made possible by a very powerful tool: the object-oriented modular simulator NN3 [LT90]. Our initial experiments with such systems of neural modules have been very promising.

References

[Ebe89] S. Eberlein. Developing a decision-making network for a rover. *Neural Computation*, 1(2), 1989.

[Ecc79] J.C. Eccles, *The Human Mystery.* Springer Press, 1979.

[Ede87] G.M. Edelman. *Neural Darwinism: The theory of neuronal group selection.* Basic Books, New York, 1987.

[Ghe89] M. Gherrity. A learning algorithm for analog, fully recurrent neural networks. In *Proceedings of the first International Joint Conference on Neural Networks*, Washington DC, 1989.

[Gro76] S. Grossberg. Adaptive pattern classification and universal recoding II: Feedback, expectation, olfaction and illusions. *Biological Cybernetics*, 23:187–202, 1976.

[LT90] A. Linden and C. Tietz. A research and development environment for modular adaptive systems. Internal report, GMD, 1990.

[Mal73] C. von der Malsburg. Self-organization of orientation sensitive cells in the striate cortex. *Kybernetik*, 14:85–100, 1973.

[Mel88] B. W. Mel. MURPHY: A robot that learns by doing. In D. Anderson, editor, *Neural Information and Processing Systems*, 1988.

[Min85] M. Minsky. *The Society of Mind.* Simon and Schuster, New York, 1985.

[MP88a] M. Minsky and S. Papert. *Perceptrons.* MIT Press, 1988.

[MP88b] M. Minsky and S. Papert. *Perceptrons.* MIT Press, 1988. Attention is drawn to the Epilogue, in which the authors make a critique of "The New Connectionism", and reintroduce Minsky's "Society of mind" approach.

[MK89] H. Mühlenbein and J. Kindermann. The Dynamics of Evolution and Learning – Towards Genetic Neural Networks. In *Connectionism in Perspective*, Elsevier (Amsterdam), 1989.

[Müh90] H. Mühlenbein. Limitations of multi-layer perceptrons—steps towards genetic neural networks. *Parallel Computing*, 1990.

[NKK90] Y. Nishikawa, H. Kita and A. Kawamura. NN/I: a neural network which divides and learns environments. In *Proceedings of the IJCNN*, 1990, Washington DC.

[Pea88] B. Pearlmutter. Learning state space trajectories in recurrent neural networks. In *Proceedings of the 1988 Connectionist Models Summer School*, Morgen Kaufmann, 1988.

[RZ85] D. E. Rumelhart and D. Zipser. Feature discovery by competitive learning. *Cognitive Science*, 9:75–112, 1985.

[SNA89] M. Sekiguchi, S. Nagata and K. Asakawa. Behaviour control for a mobile robot by multi-hierarchical neural network. In *Proceedings of the IEEE international conference on robotics and automation*, 1989.

[Sel59] O. Selfridge. Pandemonium: a paradigm for learning. In *The mechanisation of thought processes*, H.M.S.O, 511–527, 1959.

[Śmi88] F.J. Śmieja. Evolution of intelligent systems in a changing environment. GMD technical report, August 1988.

[SM90] F.J. Śmieja and H. Mühlenbein. The geometry of multi-layer perceptron solutions. *Parallel Computing*, 14:261–275, 1990.

[Śmi90] F.J. Śmieja. Multiple network systems (Minos) modules: Task division and module discrimination. To appear in *Proceedings of the AISB91 conference*, Springer-Verlag, London, 1991.

[SD85] S. Springer and G. Deutsch. *Left brain, right brain.* Freeman (New York), 1985.

[TJ88] G. Tesauro and B. Janssens. Scaling relationships in backpropagation learning. *Complex Systems*, 2, 1988.

[Wai89] A. Waibel. Modular construction of time-delay neural networks for speech recognition. *Neural Computation*, 1, 1989.

[WZ88] R. Williams and D. Zipser. A learning algorithm for continually running fully recurrent neural networks. Technical report ICS–8805, University of California Cognitive Science Dept, 1988.