ARCHITECTURE OF AN AUTONOMOUS SYSTEM:

APPLICATION TO MOBILE ROBOT NAVIGATION

Heinz Hügli*, Jean-Pierre Müller+, Yoel Gat+, Miguel Rodriguez+, Claudio Facchinetti*, François Tièche*

+ Institut de d'Informatique et Intelligence Artificielle

* Institut de Microtechnique

Université de Neuchâtel

ABSTRACT

We investigate the behavioural approach for building autonomous systems like mobile robots. In this paper we describe the mobile robot MANO together with its experimental behavioural architecture and present several advanced topics related to the interplay of the physical, behavioural and cognitive levels of the architecture like integration of vision, navigation with selfpositioning, methodology of task implementation, and two original behavioural based knowledge representation approaches.

1. INTRODUCTION

1.1 Goals of the project

The main goal of this research project is to validate an autonomous system architecture in the context of mobile robotics. It is motivated by the central role autonomous intelligent systems will play in advanced robotics. Two previous stages were devoted to

- the selection of the behavioural approach and design of the related system architecture
- the implementation and validation of the architecture with two generations of mobile robots: MARS and then MANO

The third stage, which is the object of this paper, is concerned with topics related to the interplay of the physical, behavioural and cognitive levels of the architecture.

1.2 Topics

This paper starts with a short review of the fundamentals of the selected behavioural architecture.

A section dedicated to vision shows why the behavioural architecture is best suited to integrate vision in an autonomous system and presents the adopted design methodology and the developed solutions.

Selfpositioning is presented and developed as a fundamental behavioural concept that provides a robust solution to the navigation problem without the use of a complex geometric modeling of the environment.

Another section describes MANO, the implemented behavioural architecture, that provides a versatile experimentation and development platform for autonomous mobile robotics.

Vision-based tasks are then presented to demonstrate the performance of MANO and also to illustrate the interplay of the various levels of the architecture.

2. BEHAVIOURAL ARCHITECTURE

The behavioural approach to design autonomous systems is based on the existence of individual behaviours and on the coordination of these behaviours. It states: autonomy emerges from the co-operative work of various behaviours [12].

We define a behaviour as an independent stereotyped action that is maintained by a specific perceived stimulus. Examples of robot navigation behaviours are *Wander around* or *Go along* . Each behaviour is activated by a stimulus. The overall agent behaviour emerges from the coordination of the various active behaviours.

The behavioural approach is of special interest for

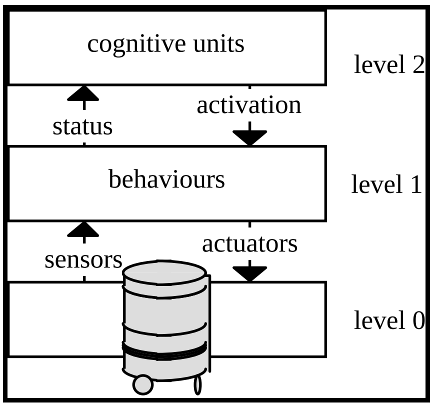

Figure 1: Behavioural approach

building agents which interact strongly with their environment by sensing and acting.

Figure 1 illustrates the basics of a the behavioural architecture which derives from this approach [4]. We recognise a structure organised in three levels of abstraction:

- Level 0: physical
- Level 1: behavioural
- Level 2: cognitive

Level 0 includes all devices acting on the robot and sensing the environment.

Level 1 is the heart of the architecture and implements the behaviours as independent modules.

Level 2 includes cognitive units which implement the tasks to be performed by the agent. These units know the behaviours by their status and proceed by activating/deactivating them. Units can be built according to very different principles.

In the frame of this paper, we present three very different approaches and implementations of the cognitive level.

- The units are controllers in connection with the sensory graph representation
- The units are state automatons in connection with a state machine representation of the robot tasks

Above all, the behavioural architecture is favourable because it is a physically grounded system [12]. Furthermore, it provides advantages by its intrinsic features like concurrency of several behaviours, simplicity and modularity of design, both abstraction and time response hierarchy.

3. INTEGRATION OF VISION IN AUTONOMOUS MOBILE ROBOTICS

We need vision for improving robots working in unstructured and changing environments. While it is easy to add multiple vision devices onto the robot, it is far more difficult to take full advantage of their individual and collective sensing capabilities. Adequate integration of multiple vision devices into the robot architecture is needed.

A feasible approach has to overcome the semantic gap between signal and symbols which is a fundamental limit of classical methods. Because of its feature of a tight coupling of sensing and acting, a behavioural system does not have this gap: it is physically grounded, i.e. its activity is in strong interaction with the real signals.

The vision architecture we developed [1] includes a large number of possible vision devices and behaviours. It illustrates how vision components integrate it well.

Following a presentation of the general methodology for integrating vision in the behavioural architecture, we describe the role and interplay of vision at the three levels of the architecture [8].

Vision in the behavioural architecture

The integration of vision in the behavioural architecture requires operations at all three levels: defining vision devices at level 0, defining vision-based behaviours at level 1 and, at level 2, where the tasks are defined, making best use of the visual behaviours available. Let us consider the three levels successively.

At level 0, defining vision devices consists merely in selecting a number of vision devices needed or useful for the application. Vision devices that have been considered so far in our investigations belong to active and passive vision, landmark vision, laser, sonar and infrared ranging.

At level 1 and of central concern is the definition of vision-based behaviours. Each behaviour is defined and developed as a widely autonomous unit responsible for a stereotyped action the robot performs under the control of a stimulus. In vision-based behaviours the stimulus is a visual pattern. Upon detection the stimulus initiates a robot action and maintains it as long as it exists, building up a control loop with feedback across the environment.

Sensor capabilities and application requirements dictate the kind of behaviours to be implemented. Typical behaviours considered for vision-based navigation are: *Going towards, Going along, Obstacle avoidance, Obstacle detection, Landmark following, Wander around, Homing, Self-positioning*, etc.

At level 2 finally, tasks the robot has to perform are defined. In the frame of vision-based applications, we select the principle of a task described by a state machine whose transitions are controlled by a status vector and each state gives rise to an activation vector. Status and activation vectors refer to the signals from and to the behaviours respectively (figure 1). This way, each cognitive unit is an automaton.

Vision at the physical level

At this level we describe a number of vision devices that we mounted onto the mobile robot in order to improve its performances.

We selected various vision devices [13] [14] [1] :

- *Laser range sensor*

This measurement device uses the principle of triangulation to measure the distance of objects in the robot environment. In our configuration, it combines a laser stripe projector and a camera to measure the geometry of the stripe profile projected on the surface of objects surrounding the robot.

- *Passive vision*

This is classical vision involving a video camera and a video processor that performs standard image processing and recognition.

- *Landmark vision system*

This active vision system uses a light source coupled to a video camera to enhance the detection of reflecting landmarks distributed in the environment. The bright landmarks are detected, labelled and tracked in a dedicated Transputer system that produces the time sequence of labelled landmarks at an approximate rate of 15 Hz[9].

- *Infrared sensor*

Each infrared or IR sensor basically measures light emitted initially by a neighbouring IR diode and back reflected by the environment. Distance is derived from light attenuation on the path. IR sensors are used for close-range distance measurements (0.3 - 1.2 m)

- *Sonar*

Sonars use the time-of-flight of a back reflected acoustic signal to measure the distance to the environment in front of the sonar. Sonars are used for medium range distance measurement (1-6 m)

Figures 2 and 3 illustrate the Mobile Autonomous Nomad (MANO) composed of the commercially available Nomad200 [15] and selected vision devices mounted on top.

Two laser-range sensors are used. One, located in front of the robot, measures a horizontal profile of the environment. The second laser range sensor is located in the back and measures the geometric profile of the environment near the ground.

In addition, two video cameras are mounted on the robot top, together with active lighting. They can be used for passive vision or, together with the illumination and back reflecting landmarks distributed in the robot environment, for landmark vision.

A belt of 16 sonars surrounds the robot, delivering the sonar range field 360 degrees around the robot. At the bottom, there is a similar belt of 16 IR sensors.

Vision at the behavioural level

This level includes the set of vision-based behaviours. It is the kernel of the system in the sense that behaviours contribute for a large part to the system capabilities.

We developed following vision-based behaviours. [1] [13]

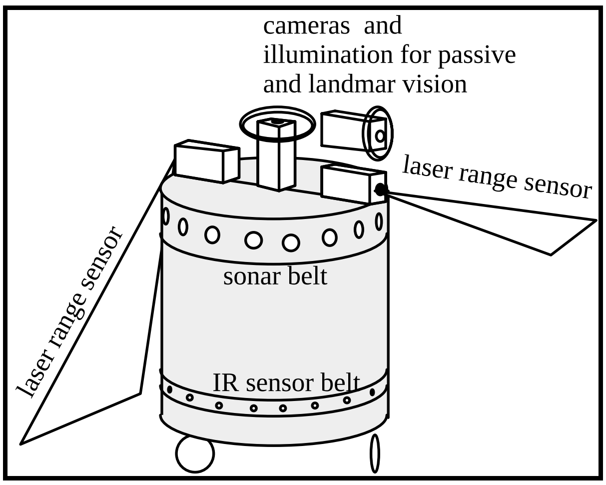

Figure 2: Visual sensors of MANO

Figure 3: MANO during *Homing on corner*

• *Obstacle detection and stop*

The behaviour stops the robot and is activated when obstacles are found in front of the robot. The stimulus is from range sensing: specific shapes of the range profile are interpreted as obstacles.

• *Wander around*

Moves the robot straight ahead and changes the direction when an obstacle is detected. The new direction is such that the new move is away from the obstacle. Here, an obstacle is a configuration of the radial range field detected by the IR sensors.

• *Go towards*

Moves the robot towards a landmark. When several landmarks are visible, the move is towards the landmark just ahead of the robot. The behaviour is no longer stimulated when the landmark is near to the robot. In this behaviour, landmarks are visible spots, detected, labelled and tracked by the landmark vision system.

• *Go along*

Moves the robot along extended obstacles like walls, keeping a constant distance to them. This behaviour comes in several flavours depending on the sensing device used for its implementation. A first one is based on the radial range profile from infrared, the other comes from an interpretation of the laser range profile.

• *Go along left, Go along right*

This behaviours are specialised forms of *Go along*. Whereas any direction of following is possible with *Go along*, these two new forms have forced following directions.

• *Obstacle detection and turn*

This behaviour turns the robot in the direction of the nearest space without obstacles. It is stimulated by obstacles located in front of the robot and detected by given shapes of the laser range profile.

• *Positioning*

The behaviour is stimulated by observed landmarks. It finds the relative position of the robot with respect to a set of three known landmarks.

• *Push box*

This behaviour is stimulated by an object near to the robot. It moves the robot towards this object and upon collision, continues its move by pushing the object straight ahead. The robot's moves are controlled to keep the object on the straight line. Here, object detection for moving toward it and for controlling the pushing is based on radial range field detected by the infrared sensors.

• *Homing*

On activation, it moves the robot to a fixed location and orientation with respect to specific visual patterns. It comes in several flavours depending on the vision device being used. It is stimulated on detection of two appropriate landmarks, on given configurations of the laser ranger and upon detection of specific visual patterns in the passive video image.

• *Free-space mapping*

Keeps track of the geometry of the environment as observed when the robot is moving. This is done by reporting the successive laser range profiles in a map.

Vision at the cognitive level

Because the cognitive level can take different forms, the role of vision will strongly depend on the form chosen for the current system. We explored vision within several contexts.

In a first context, we considered the case where the cognitive level is implemented by an automaton. Because the tasks are designed by the system developer, it is in his hands to make best use of the available behaviours. An example of a task designed in this frame is Tidying up chairs in a room that makes best use of various visual behaviours like *Wander around*, *Homing by landmarks*, *Obstacle avoidance*, *Searching a landmark, Go towards*..

In the context of autonomous navigation based on vision, we described the environment by a graph-map M=(V,E) where V is a set of vertices describing sites and E a set of edges describing the relative geometrical (or odometric) paths binding two sites v_i and v_j. The graph-map M is restricted to cycles of length greater than 2, so that it does not contain redundant paths between two same sites [2] [3] [5] [6]. Robustness is critical in this approach and is obtained by

- keeping an edge path short such that navigation can take advantage of the relative accuracy of odometric sensors on short distances
- selecting and using robust sites defined by selfpositioning

Furthermore, as it is desirable to have the robot build and maintain a map of the environment by itself we considered learning in the context of selfpositioning . The advantage of self-positioning is that it provides means for the robot to learn the structure of the environment by reducing its huge state space to the small amount of homing sites and paths described by M. During this learning phase, robust sites are selected to build the graph; site nodes get as attributes, the visual features of the associated selfpositioning behaviours [10].

SELF-POSITIONING

We present in this section a new class of vision-based behaviours we proposed and developed to provide navigation with self-positioning. Self-positioning is performed by servoed behaviours that use as homing sites simple visual features of the environment, such as landmarks, wall corners or ceiling structures [10].

Selfpositioning approach

In the selfpositioning approach, behaviours may control the robot by servoing its moves to low-level visual primitives, such as the points and segments extracted from image sequences that relate to features and structures of the environment. We understand here self-positioning (or homing) as the action of finding a stable state of the robot relatively to the environment in terms of visual primitives in an image or a set of images.

Selfpositioning is an alternative to the positioning approach which consists in reconstructing the environment from the robot sensors point of view and proceeds by finding the robot position from a match between a map and observations derived from vision, i.e. for example stereoscopy or dynamic vision, a procedure known to be rather complex.

Self-positioning proceeds differently. It provides means for the robot to learn the spatial structure of unknown environments by building a map with the sole homing sites and paths between them. Unlike other localisation methods, a geometric reconstruction of the environment from the sensors is not needed.

Applying self-positioning we developed three servoed behaviours that home the robot relatively to reflective landmarks, ceiling structures or wall corners

Homing on corner

The sensor used by the *Homing on corner* behaviour is a range-finder based on a laser-line stripping vision system. The behaviour is stimulated when a corner is recognised in the scene as shown in figure 4.

The site centre C_C (c stands for corner) lies on the corner bisecting line at a distance that is inversely proportional to the sensor depth error distribution. This parameter is fixed in the behaviour and is hence identical between corner homing sites.

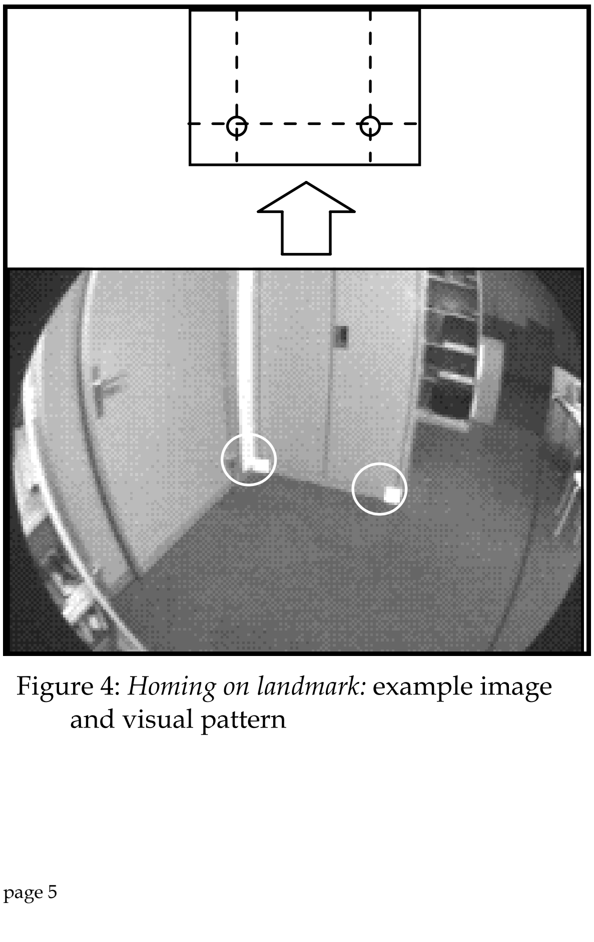

Figure 4: *Homing on landmark:* example image and visual pattern

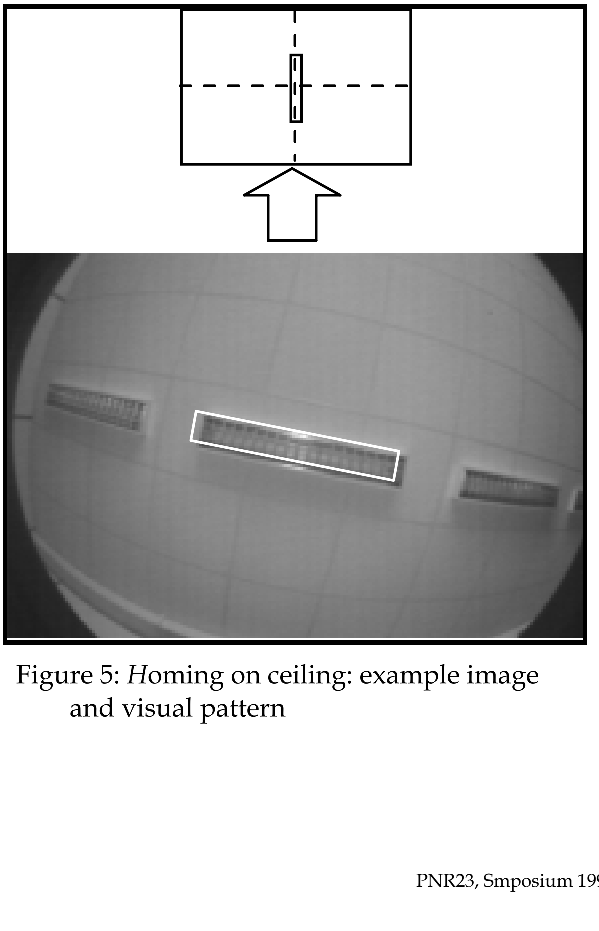

Figure 5: *H*oming on ceiling: example image and visual pattern

For autonomous navigation, it may be desirable to have the robot build and maintain a map of the environment by itself. An advantage of self-positioning is that it provides means for the robot to learn the structure of the environment by reducing its huge state space to a small amount of homing sites and paths. An internal representation is needed, so that the sequence of moves and homing sites may be inverted and retraced at any time. Several approaches have been proposed in the literature for map representation: geometric, probabilistic (which retain some properties of the geometric representation), occupancy grid, graph-based or topological. Among them, the graph-based representation usually mixes properties of some or all the other representations.

Homing on landmarks

The *Homing on landmarks* behaviour uses the landmark vision system together with a wide angle camera. Self-positioning can be performed on any two distinct landmarks.

Figure 5 illustrates Homing on landmark. The signal derived from the two landmarks visible in the instantaneous image controls the robot moves until the instantaneous landmarks correspond to the visual pattern shown.

The site centre C_l (l stands for landmark) lies on a line that passes half-way between the landmarks, perpendicular to the line supporting them. The distance between the site centre and the landmarks, as well as the size of Ω_l', is proportional to the distance between the landmarks themselves. Hence, homing sites consisting of landmarks disposed too far away may be rejected in a validation process.

Homing on ceiling structures

Since typical robot workplaces are in most cases constrained to a flat surface, a particularly interesting set up for a passive omnidirectional sensor is to place it vertically so that the optical axis is perpendicular to the ceiling. The simplicity of ceiling structures in typical office-like environments is tempting since the image processing complexity is greatly reduced. Furthermore, the *Homing on ceiling structure* behaviour can take advantage of interesting symmetric properties when servoing the robot.

Performance of homing

We ran as set of experimental tests that showed great stability for the *Homing on corners* and *Homing on landmarks* behaviours. The *Homing on ceiling structures* is currently being evaluated.

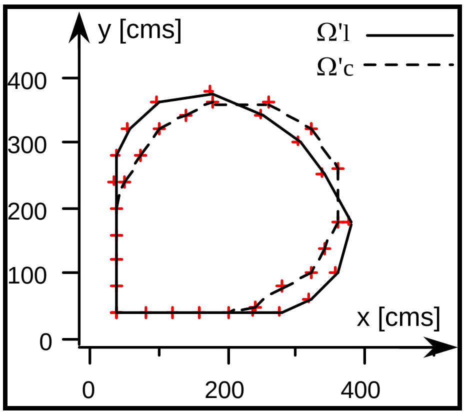

Figure 6: Capture zone for Homing on corners (Ω_c) and Homing on landmarks (Ω_l)

Quality of the homing can be characterised by the capture zone, which is the region of the robot configuration space from which the homing behaviour converges to the homing site.

Figure 6 shows the measured limits of the capture zone in the two-dimensional configuration space x and y of the robot ground. For *Homing on corner*, the wall is located on the axis Ox and Oy. For *Homing on landmarks*, the landmarks are located on a x=-y diagonal. It is interesting to observe that the capture zones of *Homing on landmarks* and *Homing on corners* have a similar shape, although they use very different vision sensors and site features.

MANO

The autonomous mobile robot architecture consists

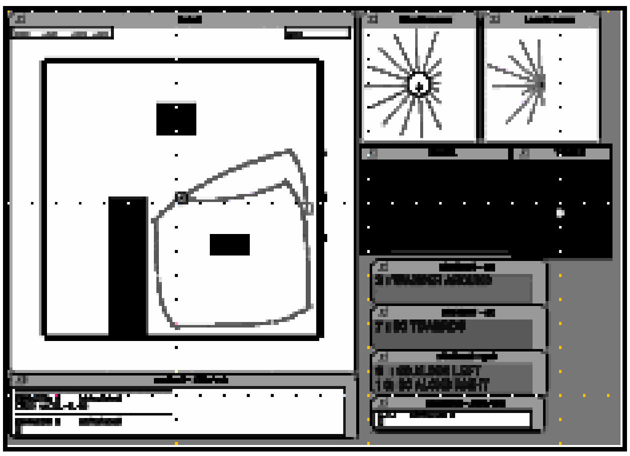

Figure 7: User interface to MANO for monitoring and control

of an environment of UNIX workstations and mobile robots equipped with various vision devices [7]. Main features of the architecture are a pool of highly independent behavioural processes, a communication schema centred on a blackboard, a virtual robot interface and a set of cognitive units (figure 8).

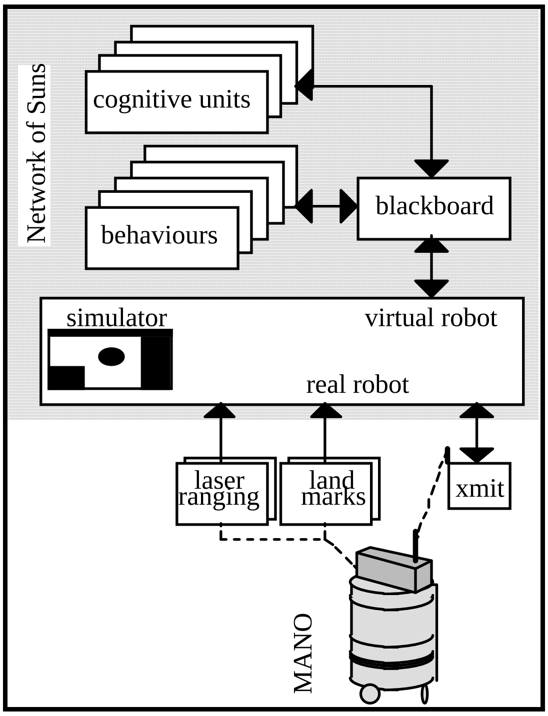

Figure 8: Experimental behavioural architecture MANO

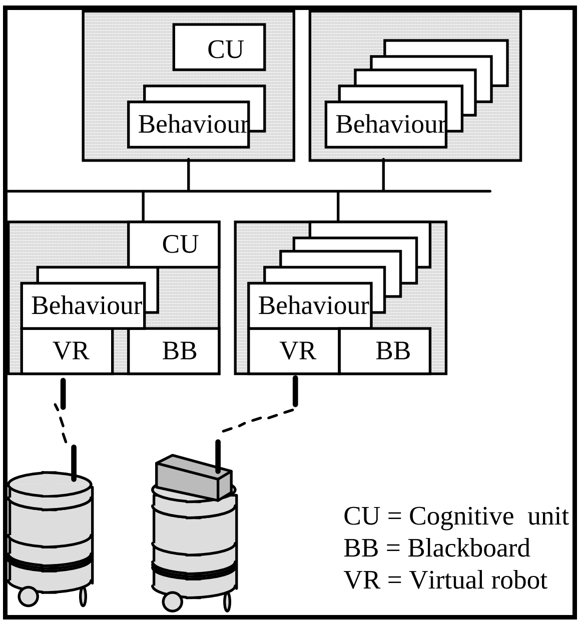

Figure 9: MANO as modular and distributed architecture

Special attention is given to low-level vision which requires intensive computing, and the implementation of instincts which require very short response time (Instincts concern the preservation of the robot integrity). Low-level vision is implemented by dedicated hardware and the instincts take place on the robot on-board computer. Both are part of the physical level.

The virtual robot provides interesting features, like equal interface to the real robot and a simulator, offering versatile experimentation conditions. Additionally, it provides also a large set of monitoring and control features through an ergonomic user-interface (figure 7).

Beyond the intrinsic features of the strict behavioural approach, the architecture offers also network-wide development and execution capabilities, allowing multiple users to work together on various real or simulated robots, and on multiple workstations. A typical configuration of the distributed architecture in use is shown in figure 9.

VISION-BASED TASKS

Each task is packaged in a cognitive unit. Typical tasks considered so far are: exploration, navigation, tidying up chairs in a room. Here are two examples of tasks implemented at the cognitive level and making extended use of vision.

In the context of cognitive units implemented as automaton the tasks are designed by the system developer and expressed as state machines. Best use of available visual behaviours is aimed at.

Tidying up chairs

The purpose of the task is to tidying up chairs in a room and consists for the robot in finding chairs and bringing them in a parking zone. The scene is illustrated in figure 10. Notice [11] that this task raises several interesting problems concerning

- full autonomy of the robot
- fast interaction with the environment
- accuracy of navigation for fetching and parking the chairs
- dynamic environment on which the robot acts itself

Figure 10: The *Tidying up chairs* task

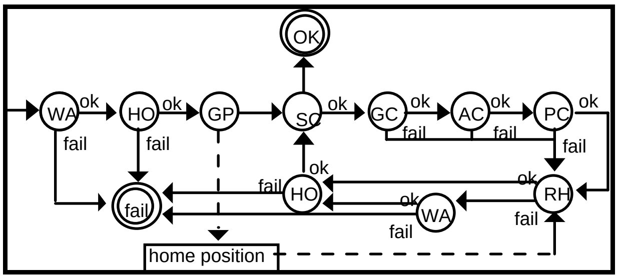

Figure 11: State machine for the cognitive unit *Tidying up chairs*

Visual behaviours involved in this task include *Wander around, Homing by landmarks,, Obstacle avoidance.,, Searching a landmark, Go towards.* Because it is assumed that chairs carry landmarks, the last behaviours translate into *Searching a chair , Go towards a chair* . Beside visual behaviours, a number of odometric-based behaviours are also used.

The state automaton defined for this task is shown in figure 11. The robot repeatedly searches for a chair, detects one, goes towards it and pushes it to a specified parking zone. In addition, obstacle detection is activable and starts escape behaviours in case of activation.

Navigation with visual homing sites

Self-positioning helps in providing robust navigation to the robot in terms most suitable to human perception. Navigating with self-positioning comes down to monitoring the robot moves between homing sites, expressed as nodes in the graph-map M described above.

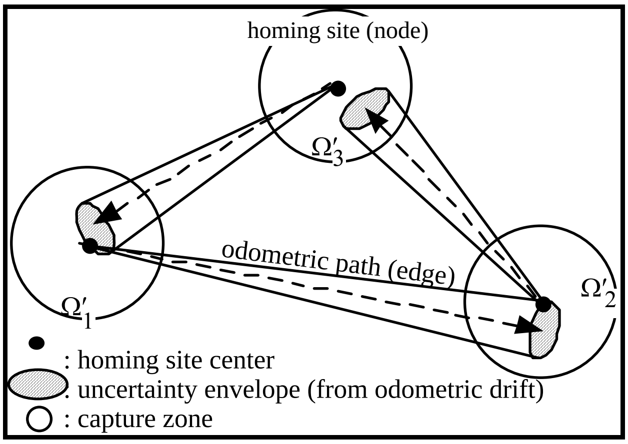

Figure 12: Navigation with visual homing sites

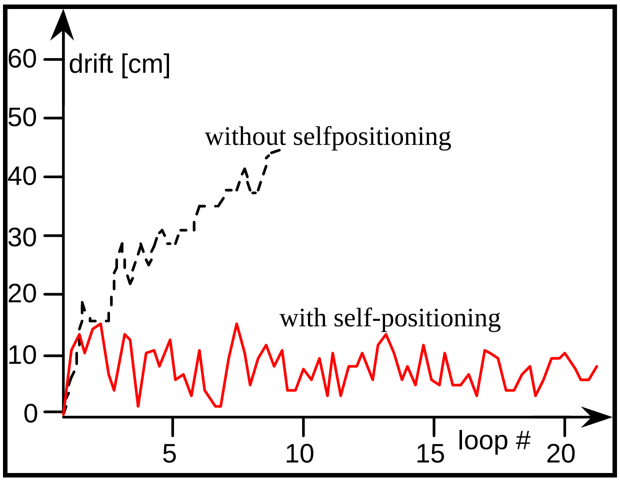

Figure 13: Position error of the robot navigating with and without selfpositioning

To demonstrate the performance of navigation with selfpositioning, we present the results of an evaluation experiment [11] in which the robot navigates in a triangle race between the three fixed homing sites of figure 12. Here we use *Homing on landmarks.* The robot moves on the edges according to the stored odometric path and performs selfpositioning at each vertex. Figure 13 compares two cases where selfpositioning is performed or omitted. The position error reported for the two cases clearly demonstrates the stabilising function of selfpositioning and the robustness of the resulting navigation.

CONCLUSIONS

We proposed and developed a three-level behavioural architecture and demonstrated its validity and usefulness to realise systems with a certain degree of autonomy like mobile robots. New aspects presented in this paper give insight in the role and interplay of the physical, behavioural and cognitive level of the architecture.

The methodology applied to integrate vision into autonomous robotics constitutes an example for extending a system with new capabilities and serves as a guide for generic system enhancements.

Self-positioning emerges as one solution to build a vision-based robot navigation system that uses universal and versatile behaviours together with simple and intelligible cognitive units.

The cognitive level accepts units of different brand to accommodate the nature of the task and the kind of knowledge representation: homing site graph-map, sensorimotor map, behaviours map.

The architecture modularity and the communication facilities are among the key elements to the design and development of effective systems. In the presented architecture, the modularity provided among behaviours and cognitive units allows to set up a system at will by selection among the available behavioural and cognitive units.

REFERENCES

Publications of IMT during this stage

[1] H. Hügli, F. Tièche, F. Chantemargue & G. Maître, " Architecture of an experimental vision-based robot navigation system", Proceedings of Swiss Vision '93, Zurich, Sept. 93

[2] C. Facchinetti, "Motion Planning and Control With Uncertainty While Sensing the Environment", Proceedings of Swiss Vision '93, Zürich, September 1993.

[3] C. Facchinetti. "Motion Planning and Control With Uncertainty While Sensing the Environment". Proc. of the International Conference on Signal Processing Applications & Technology '93, Santa Clara (USA), September 1993.

[4] J.-P. Müller & H. Hügli, "Architecture of an autonomous system: application to mobile robot navigation", Proceedings of the NRP 23 - Symposium on Artificial Intelligence and Robotics, October 22, 1993, Zurich, Switzerland, pp. 101-120

[5] H. Takeda, C. Facchinetti and J.-C. Latombe, "Planning the Motions of a Mobile Robot in a Sensory Uncertainty Field", IEEE Transaction on Pattern Analysis and Machine Intelligence, to be published in 1994

[6] C. Facchinetti. "Planification et Contrôle d'un Robot Mobile Autonome avec Incertitude sur le Mouvement". Proc. du 9ème Congrès Reconnaissance des Formes et Intelligence Artificelle, Paris, Janvier 1994.

[7] F. Tièche, C. Facchinetti & H. Hügli, "Multi-Layered Hybrid Architecture to Solve Complex Tasks of Autonomous Mobile Robots", GWIC on Intelligent systems, June 6-8, 1994, Dallas

[8] H. Hügli, F. Tièche & C. Facchinetti, "Integration of vision in autonomous mobile robotics", 1994 Int.' Conference on Systems, Man and Cybernetics, session on Real-Time Image Processing, San Antonio, Texas, Oct. 2-5, 1994

[9] F. Chantemargue et H. Hügli, "Parallélisation du suivi de cible pour la robotique mobile", La Lettre du Transputer, submitted

[10] C. Facchinetti & H. Hügli, "Using and Learning Vision-based Self-Positioning for Autonomous Robot Navigation", International Conference on Machine Learning, Robot Learning Workshop, Rutgers, New Jersey, July 1994

[11] F. Tièche, C. Facchinetti & H. Hügli, "An autonomous robot architecture for tidying up chairs in a room", PerAc'94, From Perception to Action, Lausanne, September 1994

Other references

[12] R.A. Brooks, "Elephants don't play chess", Designing autonomous agents, Ed. Pattie Maes, MIT Elsevier, 1990

[13] C. Facchinetti & H. Hügli, "Two vision-based behaviours for autonomous mobile robots", Proceedings of the First Swiss Symposium on Pattern Recognition and Computer Vision, Ed. J. Bigün & J.M.H. du Buf, EPF Lausanne, January 1992

[14] H. Hügli, G. Maître, F. Tièche, C. Facchinetti "Vision-based behaviours for robot navigation", Proceedings of the Fourth Annual SGAICO Meeting, Neuchâtel, Septembre 1992

[15] Nomadic, "Nomad 200", Nomadic Technologies, Palo Alto, USA