KNOWLEDGE ARCHITECTURE FOR ENVIRONMENT REPRESENTATION IN AUTONOMOUS AGENTS

Dario Maio, Stefano Rizzi

DEIS - Facoltà di Ingegneria, Università di Bologna, Italy

dmaio@deis.unibo.it, srizzi@deis.unibo.it

Abstract - Representing knowledge of the environment is a primary research issue in designing intelligent autonomous agents. In order to reach a satisfactory level of autonomy in executing repetitive or hazardous tasks, agents should be provided with a compact though effective method for modelling the environment at different abstraction levels. The paper proposes a knowledge architecture for the representation of environments where distinctive places can be identified. Knowledge is structured according to a taxonomy of layers, where each layer represents an abstraction of the environment which can be profitably used to carry out specific tasks. Different formalisms may be adopted for representing the different layers, so that the specific properties and advantages of each formalism can be exploited to best advantage; in particular, we adopt an analogic representation to achieve motion between distinctive places, and a symbolic representation to support high-level planning of paths in terms of sequences of distinctive places. The paper briefly discusses some solutions to problems of knowledge representation and task decomposition, and presents some experimental results.

1. INTRODUCTION

An *intelligent autonomous agent* (IAA) is a versatile machine including sensors, actuators and computing devices capable of interacting coherently with an environment and executing a variety of tasks in unpredictable conditions [7]. Agents move in the environment and improve their navigational skill through on-line sensor-based learning of robust spatial descriptions of the surroundings. They can be profitably employed for exploration in hazardous and hostile environments, as well as for performance of repetitive tasks in offices, factories, hospitals, etc. Significant research projects in this field have led to prototyping of the HERMIES series of autonomous mobile robots, as described in [8], and of the MIT Mobots as in [1].

Autonomy implies movement, which in turn requires reactive motion control on the one hand and capability of planning rough paths at high levels of abstraction on the other [4]. Both reactive behaviour and high-level path planning are based on a description of the environment, but each has radically different demands. The first asks for a well-rooted correspondence between entities in the real world and their internal representation, and requires local sensor-based information. On the other hand, too much detail in the description may become overwhelming for path planning which is more easily carried out by leaning on a symbolic representation of the whole environment.

Since each knowledge representation formalism is characterized by different individual properties, which could favour certain tasks but penalize some others, the choice of a formalism for IAAs is still an open problem. An approach to path planning based on symbolic representation of the environment can be found in [5], where connectivity graphs are generated by abstracting the original map information. In [9] reinforcement learning is used to learn to perform a sequence of elemental navigation tasks. An integrated symbolic-analogic approach is introduced in [10], where the navigation problem is generally formulated and a complete algorithm for operating a real robot in a real world is proposed.

Within the Neural Nets Project of the Italian National Council of Research, our sub-project has proposed that knowledge of an environment should be represented within an IAA at two different levels, symbolic and dynamic. The hybrid functional architecture we described in [2] is based on the coupling of a symbolic module managing explicit knowledge on distinctive objects present in the environment, and a dynamic (sub-symbolic) module implicitly encoding knowledge necessary for movement among these objects. Our work has mainly concerned techniques for the correction of sensor measures and algorithms for map clustering on the symbolic side [6], and a neural architecture for goal-oriented navigation on the sub-symbolic side [2].

In this work we outline the approach we are currently pursuing in continuing our research in knowledge representation for IAAs. We suggest that autonomous movement should have the support of a multi-layered representation of the environment, where each layer corresponds to a meaningful abstraction. Multi-layered representation supplies a semantically rich description of the environment; besides, the agent tasks can be decomposed into parallel or sequential sub-tasks on the different layers, so that their complexity is decreased.

Layering is based in the first place on the existence of distinctive places (*landmarks*) in the environment. A landmark is a singularity in the space of available measures. The criterium for recognizing landmarks depends on the environment. In structured environments, some classes of objects may be *a priori* defined as distinctive or significant; in this case, landmarks can be recognized by means of a classification algorithm applied to sensor data. For instance, an urban agent endowed with a sonar might recognize every cross-roads as a landmark by classifying sonar patterns, and an agent in an office environment might analyse the scenes taken from a camera to recognize computers, telephones, photocopiers, etc.

The concept of landmark is used to draw a "boundary line" in the environment knowledge, corresponding to the separation between reactive motion and path planning. Further abstraction levels are introduced in the environment representation through clustering. The semantic role of knowledge clusters depends on the nature of the environment: room for house-keeping robots, operative zone for industrial robots, city district for urban vehicles.

Section 2 proposes a multi-layered architecture for knowledge representation, and outlines the structures and functions of the different layers. An example for a specific environment is also presented. Section 3 discusses some issues concerning conceptual navigation of layered knowledge for solving path planning tasks.

2. MULTI-LAYERED KNOWLEDGE ARCHITECTURE

A *knowledge layer* is a meaningful abstraction of the environment. Each layer returns a different view of the environment, including only the details which are significant for a specific family of tasks or sub-tasks and using the most suitable representation formalism. The concept of layer thus encompasses knowledge, skill and representation.

If no map of the environment is available *a priori*, the agent is forced to learn a description of the environment by exploring it and interpreting the measures acquired from sensors. The knowledge architecture we propose is based on the assumption that the agent can experience the environment through three sensor channels, conceptually distinct:

- The *metric* channel returns the agent current position; the sensors used to this end are, for instance, a compass, an odometer and an altimeter.
- The *visual* channel returns an image of the nearby surroundings; candidate sensors are a sonar or a camera.
- The *symbolic* channel allows some landmarks in the environment to be tagged with a name, obtained for instance by "reading" a sign or a writing.

The view of the environment supplied from these measures can hardly be directly exploited by the agent, so it should be reorganized and interpreted. This is formally done by "abstracting" from it one or more knowledge layers which are suitable for some of the agent's tasks. Each layer may in turn generate other layers for other tasks, through a procedure of progressive abstraction which creates a taxonomy of layers. The global knowledge architecture is sketched in Figure 1; the features of the single layers are outlined below, in terms of skill (*why*), knowledge (*what*) and representation (*how*).

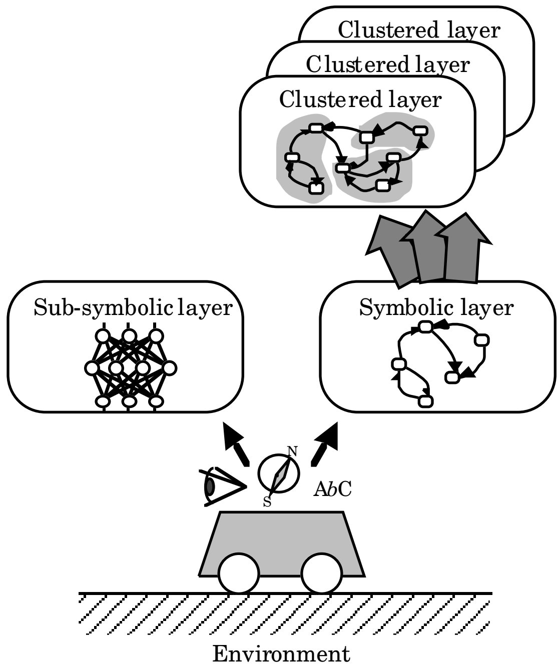

Figure 1. Multi-layered representation for environmental knowledge.

Sub-symbolic layer

Why: The sub-symbolic layer is responsible for inter-landmark movement and obstacle avoidance.

What: Local knowledge is essentially required; any global description of the environment would be an unnecessary burden.

How: Neural models seem to be more appealing than symbolic ones for representing the sub-symbolic layer, because of the properties related to distributed knowledge representation, such as immunity to noise and capability of generalization. We use a neural network which receives in input the measures from the metric and visual channels (the state of the agent in the environment), and produces as output the action to be executed, i.e., the direction to follow. A reinforcement algorithm allows optimal mapping from the set of inputs to the set of possible actions to be learned, using the current goal (the landmark to be reached) as a contextual input. Unlike most deterministic approaches, our reinforcement-based approach proves itself to work independently of the environment typology; for instance, the presence of cul-de-sacs does not affect its performance. Specific issues related to the application of reinforcement learning techniques to inter-landmark navigation are discussed in [3], together with some experimental results.

Symbolic layer

Why: The symbolic layer is the foundation for path planning.

What: Path planning is carried out more easily by hiding the details of the physical paths; hence, the environment is here represented as a map of landmarks and feasible inter-landmark paths (*routes*). Landmark recognition is based on data from the visual channel; each landmark is associated with its position (metric channel) and description (symbolic channel). A route is an abstraction corresponding to the straight-line connection between two landmarks, and is described by a cost expressing, for instance, the length of the corresponding physical path or the average time spent to cover it.

How: The formalism adopted is that of a directed graph whose vertices and arcs correspond to landmarks and routes, respectively. The symbolic layer is considered to have abstraction level 0.

Clustered layers

Why: A hierarchy of clustered layers, depending on the nature of the environment, may be abstracted from the symbolic layer; clustering supplies a richer description of the environment and allows for navigation tasks to be carried out more efficiently.

What: At the first abstraction level, the environment is represented as a map of clusters of landmarks and inter-cluster passageways (*bridges*). At the subsequent levels, each cluster includes clusters of the level below. The clustered layer at abstraction level k is called *k-clustered layer*; *k-clusters* and *k-bridges*, respectively, its clusters and bridges. If a significant form of structuring is present in the environment, clustering can be based on semantic criteria derived by classification of the landmarks; for instance, in a hospital, clusters corresponding to the progressive abstractions of rooms, wards and departments can be identified. Within an unstructured environment such as the surface of a planet, or a structured environment whose structure is not known *a priori*, clustering must be performed according to topological and metric criteria.

How: The k-clustered layer is represented as a graph whose vertices and arcs correspond, respectively, to k-clusters and k-bridges. A k-cluster represents a sub-graph of the (k-1)-clustered layer; a k-bridge between two k-clusters represents all the (k-1)-bridges which connect pairs of (k-1)-clusters of the two k-clusters. Figure 2 shows a simple example. The representation of both the symbolic and the clustered layers is object-based; the object schema adopted is shown in Figure 3.

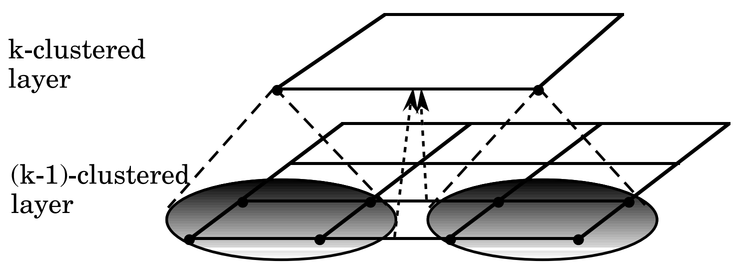

Figure 2. A k-cluster represents a sub-graph of the (k-1)-clustered layer; a k-bridge represents the set of (k-1)-bridges which connect two k-clusters.

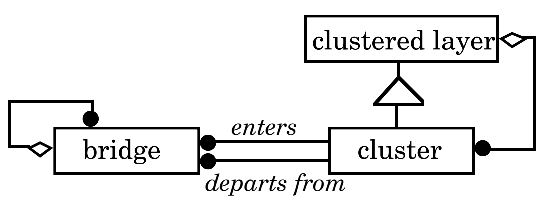

Figure 3. A clustered layer (graph) is a set of clusters (vertices); zero or more bridges (arcs) enter each cluster, and zero or more bridges depart from it. A cluster is, in turn, a graph whose vertices are clusters of the underlying layer (the white triangle represents inheritance); a bridge is a set of bridges of the underlying level. The symbolic layer is a particular case of clustered layer: its vertices are landmarks, and its bridges are routes.

2.1. An example: the hospital environment

Consider an IAA in a hospital, where landmarks are associated with serviceable or conceptually relevant entities such as departments, rooms, medical equipment, etc. The IAA should be able to execute a variety of tasks: visiting the in-patients of a department in order to bring them food and medicines, preparing the surgery for an operation, or delivering material to wards. Generally, these tasks entail path planning, possibly taking into account time, energy or more complex constraints ("before bringing the documents to the administration, photocopy them", "find as soon as possible a recharge station"). Besides, the agent must be ready to re-plan its activity in order to deal with unexpected circumstances: for instance, prepare the surgery on an emergency.

An example of layered architecture for representing knowledge of the hospital is shown in Figure 4. Three clustered layers are defined; they represent, respectively, rooms as clusters of landmarks, departments as clusters of rooms, floors as clusters of rooms (a floor may contain more departments, and a department may take up more floors), buildings as clusters of floors.

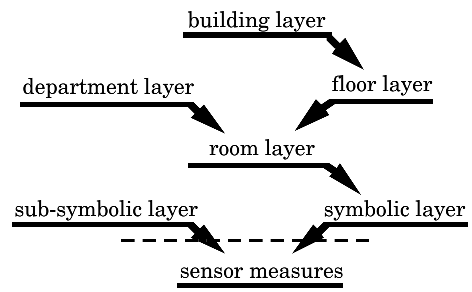

Figure 4. The taxonomy of knowledge layers for the hospital IAA.

Figure 5 shows how a fragment of the hospital would be represented by clustered layers. In Figure 5.a the map of buildings B1 and B2 is shown; B1 and B2 include two departments each. Figures 5.b, 5.c and 5.d show the room layer, the department layer and the building layer, respectively.

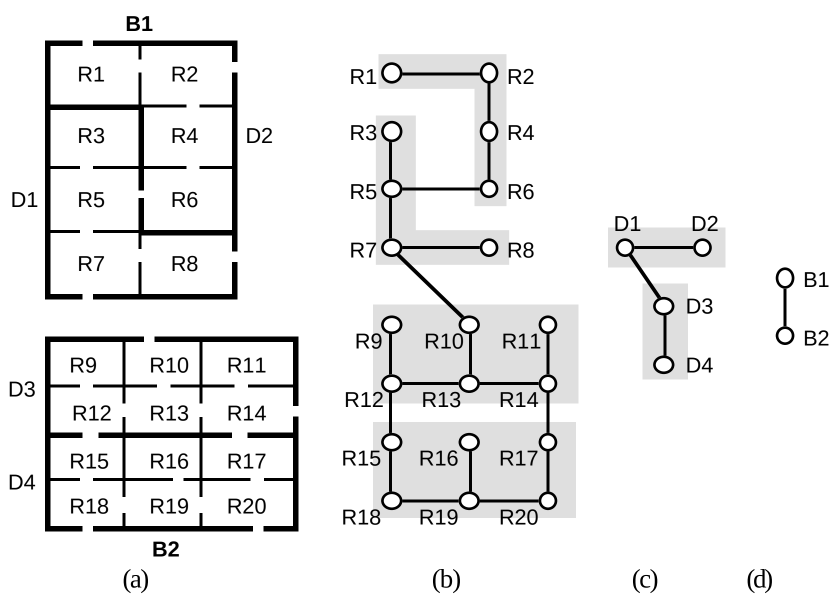

Figure 5. A fragment of the hospital. (a) Topological map. (b) Room layer. (c) Department layer. (d) Building layer.

3. PATH PLANNING AND KNOWLEDGE NAVIGATION

Most activities for an IAA involve planning a path for moving in the environment satisfying some given constraint. A simple instance of path planning problem is that of finding a path connecting two distinctive places, possibly minimizing some global cost function. More complex instances may involve multiple goals to be achieved, possibly inter-related, as well as contrasting cost measures to be taken into account.

Notable features of path planning on layered knowledge are:

- The semantics of the tasks which can be required of the agent is extended by allowing the new abstraction levels introduced by layering to be addressed.
- Complex path planning tasks can be decomposed into sequential or parallel sub-tasks on the different layers, following a *divide-et-impera* policy.

In general, the solution to a path planning problem requires the integration and co-operation of different layers. Hence, effective mechanisms for conceptual navigation of the layer taxonomy must be provided. The environmental knowledge may be navigated horizontally, i.e. intra-layer, or vertically, i.e. inter-layer.

Intra-layer navigation occurs when a problem concerning a specific layer must be solved; for example, in the hospital environment, the agent might be required to plan a path for visiting all the beds in a specific room (symbolic layer). The elemental operator for intra-layer navigation, *neighbours*, when applied to a k-cluster c in the k-clustered layer returns the k-clusters adjacent to c.

Inter-layer navigation occurs when the agent activity concerns two or more abstraction levels; an example is the planning of a path for reaching the ambulatory (room layer), after visiting a photocopier (symbolic layer). The elemental operators for inter-layer navigation, *father* and *sons*, when applied to a k-cluster c in the k-clustered layer return, respectively, the (k+1)-cluster which includes c and the (k-1)-clusters included in c.

We implemented a heuristic *divide-et-impera* algorithm for solving, on the hierarchy of graphs representing the clustered layers, a generalized shortest path problem formulated as follows: find, on the k-clustered layer, the shortest path connecting the i-cluster $\mathcal{C}_{start}^{(i)}$ to the j-cluster $\mathcal{C}_{dest}^{(j)}$ (*clustered shortest path problem*). Let n be the highest clustering level; coarsely, the algorithm works as follows:

1. find, on the n-clustered layer, the shortest path connecting the ancestors of $\mathcal{C}_{start}^{(i)}$ and $\mathcal{C}_{dest}^{(j)}$, say $\mathcal{P}^{(n)}=(\mathcal{C}_1^{(n)},...\mathcal{C}_p^{(n)})$;
2. for h=n downto k+1 do
   - 2.1 for each h-bridge $[\mathcal{C}_w^{(h)} \rightarrow \mathcal{C}_{w+1}^{(h)}]$ included in $\mathcal{P}^{(h)}$, determine which of its component (h-1)-bridges is most convenient for moving from $\mathcal{C}_w^{(h)}$ to $\mathcal{C}_{w+1}^{(h)}$, say $[\mathcal{C}''_w{}^{(h-1)} \rightarrow \mathcal{C}'_{w+1}{}^{(h-1)}]$;
   - 2.2 within $\mathcal{C}_1^{(h)}$, find the shortest path connecting the ancestor of $\mathcal{C}_{start}^{(i)}$ to $\mathcal{C}''_1{}^{(h-1)}$;
   - 2.3 within each h-cluster $\mathcal{C}_w^{(h)}$ included in $\mathcal{P}^{(h)}$, find the shortest path connecting $\mathcal{C}'_w{}^{(h-1)}$ to $\mathcal{C}''_w{}^{(h-1)}$;
   - 2.4 within $\mathcal{C}_p^{(h)}$, find the shortest path connecting $\mathcal{C}'_p{}^{(h-1)}$ to the ancestor of $\mathcal{C}_{dest}^{(j)}$;
   - 2.5 chain the paths obtained into a path $\mathcal{P}^{(h-1)}$;

Our algorithm can be evaluated in the case i=j=k, when the clustered shortest path problem reduces to the classical problem of finding the shortest path on a graph. The optimal solution to this problem is yielded by Dijkstra's algorithm in $O(\lambda^2)$, where λ is the number of vertices in the graph. It can be proven that, when i=j=k=0, our algorithm has time complexity $c = O(\lambda^{(n+3)/2n})$, where n is the highest clustering level. Obviously, in some cases our algorithm yields a sub-optimal solution; experimental tests conducted on a sample of random maps showed that the shift from optimality is contained within 1% in 55% of cases, and within 20% in 90% of cases.

4. DISCUSSION

In this paper a multi-layered architecture for representing knowledge of the environment to be used by IAAs for autonomous movement has been presented. Each knowledge layer defines an abstraction of the environment which efficiently supports the execution of specific sub-tasks. The adoption of different formalisms for representing layers allows the specific properties of each formalism to be exploited at their best. Some issues concerning path planning on layered knowledge have been discussed, and a divide-et-impera algorithm for solving the clustered shortest path problem has been outlined.

We are currently working on algorithms for path planning in presence of complex constraints and on the definition of *ad hoc* strategies for multi-agent exploration of unknown environments.

REFERENCES

1. R.A. Brooks and A.M. Flynn, "A robot being", *NATO Workshop on robotics and biological systems*, Tuscany, Italy, 1989.
2. P. Ciaccia, D. Maio, and S. Rizzi, "Integrating knowledge-based systems and neural networks for navigational tasks", *Proc. IEEE COMPEURO*, Bologna, Italy, 1991, 652-656.
3. P. Ciaccia and B. Montanari, "Reinforcement-based systems for solving navigational tasks in large domains", *Proc. MCPA '93*, Halmstad, Sweden, 1993, 367-374.
4. E. Gat, "On the role of stored internal state in the control of autonomous mobile robots", *AI Magazine*, 14(1), 1993, 64-73.
5. P.D. Holmes and E.R.A. Jungert, "Symbolic and geometric connectivity graph methods for route planning in digitized maps", *IEEE Trans. Patt. Anal. Machine Intell.*, 14(5), 1992, 549-565.
6. D. Maio and S. Rizzi, "Map Learning and Clustering in Autonomous Systems", to appear in *IEEE Trans. Patt. Anal. Machine Intell.*, 1993.
7. D. Nitzan, "Development of intelligent robots: achievements and issues", in *IEEE Journal of Robotics and Automation*, RA-1, 1, 1985, 3-13.
8. N.S.V. Rao, "Algorithmic framework for learned robot navigation in unknown terrains", *IEEE Computer*, 1989, 37-43.
9. S.P. Singh, "Transfer of learning by composing solutions of elemental sequential tasks", *Machine Learning*, 8, 1992, 323-339.
10. G. Vercelli, R. Zaccaria and P. Morasso, "A theory of sensor-based robot navigation using local information", *Proc. AIIA 91*, 1991, 342-351.