dr inż. Grażyna Brzykcy, dr inż. Jacek Martinek, dr inż. Adam Meissner, dr inż. Piotr Skrzypczyński
Katedra Automatyki, Robotyki i Informatyki Politechniki Poznańskiej

Blackboard Agent Architecture for Perception and World Modeling[*]

In the paper, the architecture of a mobile robot cooperating with other robots and some stationary devices in a task of collective perception and world modeling is considered. We analyze data-driven processing of information performed by an individual robot treated as an agent and we propose to organize it as a set of experts (also treated as agents) exchanging data by means of a blackboard. We analyze functions performed by the blackboard agents.

1. INTRODUCTION

In the contemporary research projects concerning robotic systems, problems of the use of many cooperating robots are very often considered. It is obvious that a team of robots can solve more complex tasks than a single robot. But even in the case of simpler tasks, which can be solved by one robot, the introduction of a system with many cooperating robots may be reasonable. If every component of the system represents a different perspective of the problem to be solved, uses different representations of data and all components work in cooperation, then the solution can be found faster and the system is more reliable and more resistant to unexpected disturbances.

Regardless of types of particular tasks solved by robotic systems (e.g., production or transportation tasks), their operation strongly depends on the available model of surroundings (the world, the environment). In the paper, the task of collective perception and collective world modeling by a team of different robots is considered. Our research aim is to design architecture of such a system for perception and world modeling.

We assume that the system prototype will be realized by means of physical devices and software modules. In the system, mobile robots with sets of sensors and some stationary devices like a scene monitoring overhead (ceiling-mounted) camera are separate components. Moreover a knowledge base containing *a priori* given data about the environment is available and may be updated by the system operator. The system structure is not fixed. The number and the characteristics of components may vary, so the system is open to modifications. The particular system components have different perception abilities and their individual world knowledge is limited, potentially incomplete, uncertain or out-of-date.

The collective perception and world modeling task is realized by mobile robots during their movement along an indoor route, which is described by a sequence of points. Information gathered during this movement from own sensors of the robot, from other robots (on demand) and from stationary devices is used to construct or to update the internal robot world model.

In the prototype set-up, two mobile robots of Labmate type, and one monitoring camera will be used. Both robots have on-board PC computers and are equipped with laser scanners and

---

[*] This work is sponsored by the Polish National Committee for Scientific Research under grant 8T11A003 19

ultrasonic rangefinders (sonars). One of these robots will be also equipped with a simple vision subsystem with one CCD camera. Labmate robots are differential-drive vehicles. Each of the two driving wheels is controlled independently so that robot can translate, and rotate with zero turning radius, what significantly simplifies the path planning.

The preliminary analysis of the perception and world-modeling issues in the mobile robotics domain facilitates to distinguish following properties of the system under study:

- components of the system act in an independent autonomous manner,
- all the components have the common area of operation (e.g. an industrial plant),
- the robots and the environment influence each other: the actions of a robot (e.g. a displacement) depend on the state of the environment, and from the other hand the robot effects changes in its environment by executing actions,
- system components can communicate each other.

To model a cooperative system with the above-mentioned properties the agent concept can be used [5]. In the robotics domain agents are usually defined as autonomous or semi-autonomous hardware or software systems, which perform their tasks in a complex, dynamic environment [8]. Autonomy is understood here as the ability to make decisions based on an internal agent world representation, without being controlled by any central station. An agent has a perception and communication ability, and its functionality is expressed through the actions it takes, including the communication actions.

In the system under study the agent concept is used to model the mobile robot, monitoring subsystem with ceiling-mounted camera and the human operator. The problem of the collective perception and world modeling in such a multi-agent system can be reduced to the organization problem of an effective co-operation among agents. The robot-agents play a particular role in this system, because they are owners of the world models. These models are stored in a form of vector- and raster-based maps.

In the next part of this work the single robot-agent architecture is outlined.

2. MOBILE ROBOT ARCHITECTURE

2.1. Characterization of the mobile robot

2.1.1. Mobile robot tasks

In the framework of the perception and world modeling task the mobile robot moves among given sub-goal points in a partially unknown indoor environment. The robot perceives this environment by means of available sensors and builds a model of it.

It is assumed that in the case of an obstacle (a blockade on the preplanned path) encountering, the robot is able to detect it and to make a necessary detour. This task is performed by making use of the simple reflexive navigation [1]. To obtain the necessary information for self-localization task the robot takes advantage of its odometry as well as external sensors. In weakly-structured environments, cluttered with obstacles or in a case of dynamic object presence, this problem can be better and much easier solved in collaboration with the monitoring agent, acting as an external localization system.

The up-to-date world model is built by integration of data supplied by mobile robot sensors and data extracted from the *a priori* model of the environment. This model is provided to the mobile robot in the moment of system initialization. To compensate its limited perceptual capabilities in the process of map building and to resolve possible ambiguities in the model, the robot can exchange data with other agents in the system (other robots).

2.1.2. Necessary data

To perform the reflexive navigation an up-to-date representation of the surrounding environment is needed. This representation should be built fast and it exploits all sensory information available to the robot. An interesting proposal of such a local representation aimed at supporting obstacle avoidance methods is the 'histogram grid', introduced by Borenstein and Koren in [2]. Moreover, the grid-based map can be used as a common ground for fusion of different range sensor data.

The task of determining the current position and orientation of the mobile robot (the self-localization task) is performed when the estimate of robot position obtained from the odometry needs to be reset by an independent source. The common way of providing the robot with the position information is the map-matching technique. It needs a local environment representation that can be matched effectively to the global world model given *a priori*. Such a representation is the local vector-based map. It is not necessary that this map contains all the perceived objects in the environment – its goal is to represent the main geometric structures (e.g. walls) which can efficiently support the localization task. The vector-based map is composed of line segments.

The global world model takes also a form of the vector-based map. Unlike the local vector map it is structured in particular objects (sets of line segments) which resemble obstacles in the real environment. These objects are represented as polygons and poly-lines. They are attributed with additional properties which are important to the model updating process, e.g. possibility to move an object or to modify its shape.

2.1.3. Robot sensors

The data necessary to perform mobile robot tasks are provided by the on-board range and vision sensors. There are optical scanners and ultrasonic range finders on mobile robots.

The optical (laser) scanner can provide precise range measurements to the surrounding obstacles. It has high angular resolution and high measurement credibility. But the scanner has a very small light spot, and due to this limitation it can overlook small obstacles (e.g. pipes, wires). For the scanner used in the presented system an additional disadvantage occurs – the low scanning speed. The scanner can produce 2D map of the environment, vector- or raster-based.

In contrary to laser sensors, ultrasonic range finders suffer from wide-beam problems and spectral reflections, but they are quite fast and have 3D, conical field of view. With respect to these features sonars are used to detect local obstacles which often are small or moving objects. They are also able to generate grid-based map.

The on-board camera of the mobile robot is solely used to detect passive artificial landmarks purposefully attached to objects in the environment. This assumption simplifies the image processing and gives the vision subsystem a chance to contribute to the robot navigation in real-time, by providing an alternative way of self-localization.

An important source of information is also the odometers, which are able to yield the current position and orientation of the mobile robot without recourse to any external sensing.

2.2. Data processing

In the mobile robot, a complicated, multi-stage data processing is undertaken:

- data can be provided by robot subsystems or they can originate from another agents of the system,
- different robot tasks are performed on the strength of data that are expressed in the diverse formats though can have a common origin (e.g. the range information is used both in the raster and in the vector map),
- different sensors produce data with different uncertainty level,
- vast majority of data processed in the system are the local data, i.e. they describe only a part of the environment,
- data have to be acquired and processed continuously while the robot is moving,
- process of the world model building involves the integration of very different types of information (sensor-based and *a priori*, local and global, uncertain and certain etc.).

Two kinds of actions can be distinguished in the information processing in the multi-agent system, namely the transformation of raw sensory data to the form of environment maps, and the exchange of data between different world model representations. Most of these processes has been recognized and described as the operators of so called Perception Network for a group of mobile robots, proposed in [7].

Operations described below are important with respect to the vital tasks of the mobile robot.

1. Estimation of the position and orientation (with uncertainty measure) from the robot odometry.
2. Local grid map update from the laser scanner data.
3. Local grid map update from the sonar data.
4. Local vector map building from the laser scanner data.
5. Analysis of the local grid map and the generation of the next move for the mobile robot controller (here slightly modified Virtual Field Histogram algorithm from [2] is used).
6. Conversion of the local grid-based map to the vector form.
7. Integration of the vector map extracted from the grid map with the local vector map generated directly from the scanner measurements.
8. Estimation of the current robot position and orientation by means of map matching.
9. Extraction of visual landmarks from the environment by the on-board camera and the estimation of robot position from these data.
10. Optimal integration of all position estimates available to the robot at the given moment (including estimation from monitoring subsystem) by means of Kalman filtering.
11. Global vector map update by using the current local vector map, current position estimate, and (possibly) pieces of vector maps from other mobile robots.

All these operations are well determined independent subtasks that robot has to perform to achieve its goal (solve the problem). Each of the subtasks can be separately defined as a "black box" with some input and output. The "black boxes" are loosely coupled by data they exchange – the input of one "box" is the output of another. This kind of subtasks interaction can be realized via shared global database of data needed by or produced by the "boxes".

An order of subtasks execution is however not known in advance – the decision to perform particular subtask is made dynamically using current data values (e.g. inconsistency of data triggers on-board vision system). Control is data-driven and opportunistic in nature. Data processing system with many independent operations – "experts" collectively working on commonly accessible data can be seen as a blackboard system [4].

2.3. Blackboard architecture of mobile robot

2.3.1. Blackboard system

Some proposals of the blackboard architecture applications to multi-sensor system for an autonomous single robot were discussed in literature (e.g. [6]). With relation to the task of collective perception and world modeling an idea to impose blackboard architecture on the robot-agent was sketched for the first time in [3].

The blackboard system consists of three basic components: the data structure (blackboard) that is appropriate for the problem solving domain and is mostly organized as one or more application-specific hierarchies, the set of processing modules (knowledge sources, experts) that transform data from the blackboard and control machinery realized by a special expert. The modules are kept separate and independent and each of them is able to perform an action. They join in solving the problem according to the following cycle:

- triggering the module in view of new information on the blackboard,
- recognition of information context (satisfaction of action preconditions),
- execution of an action,
- storing data in the blackboard.

The control mechanism is responsible for execution of each problem solving cycle, particularly for allocation of processing resources to the most promising expert. Explicit representation of the control facilitates the system users to define the complex control strategies.

The blackboard system can be easily modeled as an agent system [9]. In the system under study both processing modules and information providers – sensors – placed around the blackboard are good candidates for the agents. Their architecture is potentially very simple because:

- each module has precisely defined tasks (e.g. feature transformation, data fusion) which are performed by itself (no calls of other agents),
- the blackboard (often the part of the blackboard) is an environment for each module; it is the source and destination of the module data,
- module needs no knowledge about other modules.

Blackboard organization of the agent system introduces additional advantages such as parallel data access (here blackboard data access) and concurrent execution of many agents. It is easy to introduce changes to the blackboard system as putting an agent in and deleting it from the system is very simple (this property is material in case of sensor set modification). Finally, the system organization does not depend on agent implementation.

2.3.2. Agent separation criteria

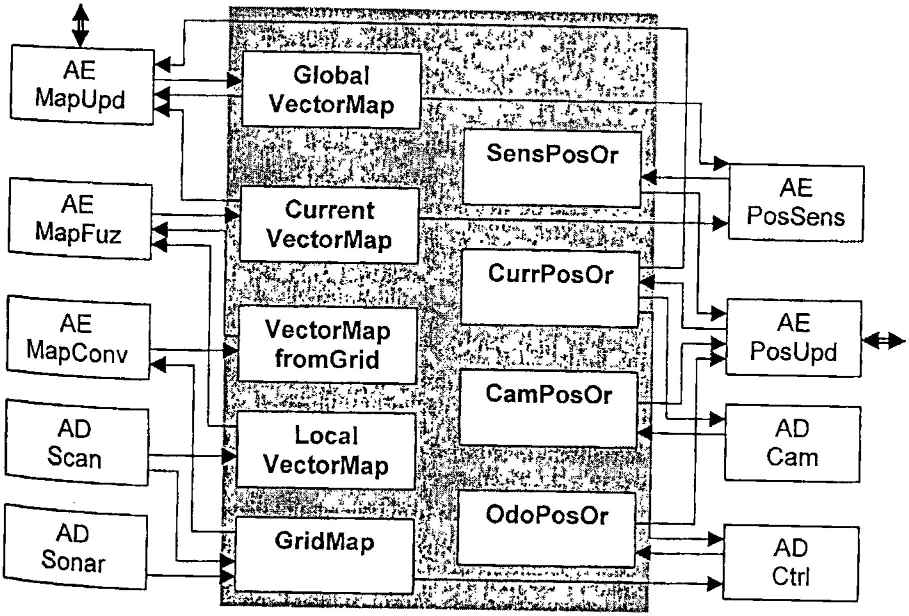
In the blackboard system a strong dependency holds between blackboard data structure and repertoire of agents gathered around it. For the system designer the most desirable and effective way of acting is to obtain the blackboard structure that is a good model of problem under study. To define the appropriate blackboard structure one has to settle what constitutes the wanted solution. The designer ought to define also the necessary data and the knowledge how to process them. The abstraction levels of data representation on the blackboard during the process of problem solving determine the granularity and the way of knowledge division into agents that perform the specialized subtasks. The higher number of intermediate levels, the more specialized processing agents. However how the problem is partitioned into sub-problems makes a great deal of difference to the clarity of the design, the efficiency of problem solving and ability to solve the problem at all.

The set of agents – executors of subtasks proposed in 2.2 – is derived from the list of Perception Network operations [7] and seems to match well the problem of translating multi-sensor data, domain knowledge and *a priori* knowledge into aggregated world model. In 2.3.3 the blackboard structure and actions of particular agents are depicted.

With agents proposed in 2.3.3, the list of potential actions performed by mobile robot in the course of collective perception and world modeling is not exhausted. Especially the subtasks originated in the problem of robot moves planning are not described. But the blackboard architecture does not restrict the possibility of enlargement of an action set; both the blackboard structure and collection of agents are easily modified.

2.3.3. Blackboard and agents

The blackboard contains data describing robot world model on various levels of abstraction. The lowest level consists of the local vector map (*LocalVectorMap*) and the local grid map (*GridMap*). The first map is a 2-dimensional table, the other map is represented by a list of segments. A segment is described by its length, the coordinates of its center and the angle between the x-axis and the line containing the concerned segment.



Blackboard agent architecture

Also, a matrix of uncertainty is assigned to every segment on the list. Both of maps are used to construct the current vector map (*CurrentVectorMap*) and the global vector map (*GlobalVectorMap*) at last, which is the final world model of the robot. Maps are attended by the current position and orientation of the robot. These data are obtained from various sources, such as odometry (*OdoPosOr*), the on-board vision subsystem (*CamPosOr*), the vector map matching procedure (*SensPosOr*) and the position comparison algorithm (*CurrentPosOr*).

In the following part we describe agents in the sense of their input and output data and also initial conditions of agent actions.

The scanner agent (*ADScan*) preliminarily processes data obtained from the scanner. This procedure is executed every fixed quantum of time or on demand. It corrects input data, updates a local grid map (*GridMap*) and builds a new local vector map (*LocalVectorMap*).

In order to update a local grid map the sonar agent (*ADSonar*) processes the data coming every settled quantum of time from ultrasonic rangefinders.

The vision subsystem agent (*ADCam*) is activated when the current position and orientation of the mobile robot is needed (*CamPosOr*). To achieve this goal the agent processes the acquired image, detecting landmarks and comparing them to the *a priori* given list of landmarks. The agent also may obtain the current robot position (*CurrPosOr*) from the blackboard.

The data gained from odometry, i.e. position and orientation of the robot (*OdoPosOr*) are cyclically stored on the blackboard by the agent of the robot controller (*ADCtrl*). This agent also detects obstacles on the path and planes future moves of the robot with regard to its current position and the information gained from local grid map.

The transition from the grid-based world model (*GridMap*) to vector-based model is performed by the map conversion agent (*AEMapConv*). In this case, the geometrical interpretation takes place at the much later stage than for the vector map obtained in direct line-based interpretation, and thus much more evidence can be accumulated [10]. This process is necessary to establish the current position and orientation of the robot.

The local map fusion agent (*AEMapFuz*) compares a local vector map (*LocalVectorMap*) to a vector-based map acquired from the grid-based map (*VectorMapfromGrid*) in order to integrate the data of the same representation. The unification of maps comprises individual segment matching. This procedure is performed to build a new current vector map (*CurrVectorMap*) every time a grid-based map is transformed to a vector-based form.

In order to determine the position and orientation of the robot (*SensPosOr*), the self-localization agent (*AEPosSens*) compares the current vector map (*CurrVectorMap*) to the global vector map just after its creation.

The current position and orientation of the robot (*CurrPosOr*) is established by the position update agent (*AEPosUpd*) with regard to data coming from such sources as odometry (*OdoPosOr*), the vision subsystem (*CamPosOr*), the vector map matching (*SensPosOr*) procedure or from other agents (e.g. ceiling-mounted camera).

The creation of a new vector map (*CurrVectorMap*) and determination of the current robot position (*CurrPosOr*) are necessary conditions to initiate the process of updating the global world model (*GlobalVectorMap*). Maps are integrated by the global map update agent (*AEMapUpd*) which tries to unify both maps additionally using the domain knowledge. The agent evaluates segment parameters and segment sets attributes. It also detects situations when a robot has to acquire a part of the world model from another agent.

3. FINAL REMARKS

In the paper the blackboard architecture is described which is aimed at the data-driven sensor information processing resulting in the symbolic representation of data. The structure of the blackboard, the set of agents and their tasks are presented. The current research concerns the details of data representation, the agent activation events and the synchronization aspects of control.

REFERENCES

[1] Arkin R., *Behavior-Based Robotics*, The MIT Press, Cambridge, 1998.

[2] Borenstein J., Koren Y., *Histogramic In-Motion Mapping for Mobile Robot Obstacle Avoidance*, IEEE Journal of Robotics and Automation, Vol. 7, No. 4, 1991, pp. 535-539.

[3] Brzykcy G., *Koncepcja wieloagentowego systemu percepcji i modelowania otoczenia,* Inżynieria wiedzy i systemy ekspertowe (Bubnicki Z., Grzech A., red.), Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2000, tom 2., str. 111-118.

[4] Engelmore R., Morgan T. (eds.), *Blackboard Systems*, Addison-Wesley, 1988.

[5] Hunhns M., Singh M. (eds.), *Readings in agents*, Morgan Kaufmann Publishers, 1998.

[6] Kappey D., Pokrandt P., Schloen J., *A distributed multisensor blackboard system for autonomous robot*, SPIE Conference Sensor FusionVII, Boston, 1994.

[7] Kasiński A., Skrzypczyński P., *Cooperative perception and world-model maintenance in mobil navigation tasks*, Distributed autonomous robotic systems 3 (Lueth T. et al., eds.), Springer, Berlin, 1998, pp. 173-182.

[8] Müller J., *The design of intelligent agents: a layered approach*, Lectures Notes in Artificial Intelligence, Vol. 1177, Springer, 1996.

[9] Schwartz D., *Cooperating heterogeneous systems*, Kluwer Academic Publishers,1995.

[10] Skrzypczynski P., Drapikowski P., *Environment Modelling for a Multi-Agent Mobile System*, Proc. EUROBOT'99, Zurich, 1999, pp. 41-48.