Proceedings of the 1996 IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems

Decision–Theoretic Multisensor Planning and Integration for Mobile Robot Navigation

Steen Kristensen* Henrik I. Christensen

Laboratory of Image Analysis, Aalborg University
Fredrik Bajers Vej 7D, DK-9220 Aalborg E, Denmark
E-mail: {sk, hic}@vision.auc.dk, URL: http://www.vision.auc.dk/~{sk, hic}

Abstract

In this paper, a decision–theoretic approach to multisensor planning and integration is investigated. The decision–theoretic framework allows for rational decision making under uncertainty and furthermore a highly modular system description that facilitates easy system integration. Experiments with a real robot show that the decision–theoretic sensor planner is capable of making rational real–time decisions about sensor use in an autonomous mobile robot context.

1 Introduction

In this paper, we are concerned with the issues of multisensor planning and integration for mobile robot navigation. Some authors (e.g., [5]) define the sensor planning problem as selecting the optimal sensor parameter values given one or more sensors. This, of course, is primarily of interest with respect to active sensing. Another part of the sensor planning problem is not *how* to use the sensors but determining *what* sensors to use and when (e.g., [7]). This is also called the *sensor selection problem.* It is that part of the problem that is addressed in this work and subsequently referred to as the sensor planning problem. Regarding the sensor integration problem, we will adhere to Bozma's definition [3]: "to provide an effective model of assembling a system's modules together and incorporating the information provided by each module into the operation of the whole system".

The autonomous mobile robot domain provides a realistic and challenging context for empirically testing the obtained results, since this domain requires real–time response to events in an uncertain environment. This means that the developed theory/framework should be fast and capable of dealing with uncertainty.

*Part of this work was done at the GRASP Laboratory, University of Pennsylvania, USA.

1.1 Sensor Planning and Integration

The nature of the sensor planning/integration problem depends on the overall task addressed and the architecture of the system considered. In this work we consider a system designed after the *purposive paradigm*, where a number of specialised functional modules, also called *purposive modules*, are employed to solve the robot navigation task. We will define a purposive module as a self–contained entity capable of doing its own sensing, processing, and actuating—all specialised for a single purpose. In a system with a number of such modules operating in parallel two problems arise; the action selection problem and the sensor planning (selection) problem. The action selection problem is the often addressed problem of selecting an action on basis of the actions (outputs) from each purposive module. The sensor planning problem, which is the problem of managing the inputs to the modules in the system, has been addressed to a much less extent. This indicates that the assumption has been that sensing is a pure read operation and thus not subject to the potential conflicts as are the outputs to the actuators. This is, however, not true for several reasons. First of all, with the advent of active perception [2] it has become necessary to decide what module can use a given sensor/actuator[1] at a given time. This is due to the fact that sensors are potentially un–sharable between modules since each module may configure the sensing system for its own purposes as, e.g., letting the cameras of a camerahead fixate a specific world point.

Therefore, there has to be a sensor planner to schedule the sensing resources and a mechanism for doing action selection. In the purposive paradigm this is the same thing, however, since a purposive module with no sensors allocated can not do any useful work and thus, when doing active perception, the sensor planning problem encompasses the action selection problem.

[1]When dealing with active sensing, the border between a sensor and an actuator is blurred and thus we will from now on use the term "sensor" in the meaning "sensor/actuator". Actually, there is—especially in mobile robotics—a strong dualism between sensing and acting; you sense to move and you move to sense.

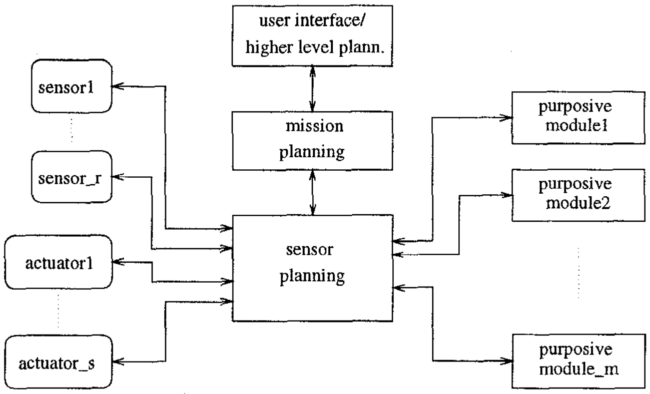

Figure 1: Illustration of a purposive system with handling of sensor/actuator contention. Some sensors might be accessed by several modules at the same time, and some may only be used by a single (or no) module. The sensor planner acts as an intelligent switchboard connecting sensors/actuators to purposive modules.

What we would like to have is a system like the one depicted in figure 1, where the central issue for the sensor planning is to decide what modules to grant the control of what sensors in what sequence, in order to achieve the best overall performance of the mobile robot system.

Also shown in figure 1 is that the sensor planner is at the bottom level of a hierarchy of planners that comprises the planning capacity of the system. This means that the system architecture, we are considering, is a hierarchical one, where the sensor planner is at the lower level. At the very lowest level is purely reactive planning which resides inside the purposive modules. This allows higher level planners to abstract away from simple things, e.g., obstacle avoidance, that are necessary in order to protect the robot and the environment. Likewise, the sensor planner is there to allow higher level planners to abstract away from how to schedule sensors and actuators and thereby also when to run what purposive modules.

It is important to notice that the sensor planner schedules sensors and actuators but does *not* perform sensor fusion. This takes place in the purposive modules once they have been granted access to the sensors. So while the sensor planner decides *what* sensory actions to carry out, the purposive modules decide *how* to do this.

1.2 Motivation

The motivation for the work presented in this paper is that there to date seems to be a lack of work addressing the problem of sensor planning in a framework where multiple purposive modules are competing for sensor resources—especially in the context of active perception. Active perception means that sensors are potentially un–sharable and that sensing and acting cannot be separated in a meaningful way. Much previous research does not seem to have taken the consequence of this fact.

Another strong motivation for this work is that current integration frameworks are not very flexible since they do not explicitly cope with sensor contention and since they are not formal. This means that much of the integration consists of creating integration schemes more or less by hand for each specific task. While good methodologies and tools for this have appeared, e.g., Simmons' *Task Control Architecture* [11] and Gat's *Conditional Sequencing* [4] it is still believed that on–line decision making will result in more dynamic systems and better resource utilisation. Moreover, there seems to be only very few purposive modules in current robotic systems. This is also believed to be a consequence of the tedious integration work associated with building systems by hand. We believe that this can be alleviated by using a formal method with a well–defined semantics.

Therefore it has been wanted to build a decision–theoretic sensor planning framework that will allow for efficient, on–line sensor utilisation and system integration, thus resulting in efficient, modular, and dynamic autonomous robot systems. Or, to quote Russell and Wefald [10, pg. 177] "We believe, however, that a real–time decision–theoretic approach to planning promises to satisfy the goals of all sides of the 'planing debate' .".

In the following section, the basic theory behind the proposed decision–theoretic sensor planner is presented. Then, in section 3, experimental results are presented, and finally, in section 4, conclusions are drawn and future research directions are given.

2 Sensor Planning with Bayesian Decision Analysis

It has been chosen to use Bayesian Decision Analysis (BDA) for the sensor planning since it is a framework that allows for reasoning under uncertainty which is considered crucial in real world applications. Being a formal framework, this approach enables the use of a large body of theoretical work and, moreover, it allows the direct use of statistical data. BDA has traditionally been used in economics and in that terminology, the purposive modules are competing for the resources in a cost/benefit manner according to probabilistic models.

Decision–theoretic planning is defined as [12]:

> *Decision–theoretic planning* is a special case of the general AI planning task where (1) effects of actions are described by probability distributions over outcomes, (2) objectives are described by utility functions, and (3) the criterion for effective plan synthesis is expected–utility maximization.

In the following, we will outline the theory behind using BDA for sensor planning.

An important thing to consider when planning in general is how far and how detailed to plan. The level of detail is fixed in this problem since we want to make decisions at the level of sensor action, i.e., what sensors to grant what modules. How far to plan is however an open question which should also be addressed.

The reason for planning far ahead is that elaborate sensing strategies can be found and thus sub-optimal solutions avoided. There are however many problems related to planning far ahead when dealing with uncertain, dynamic real world environments, the major ones being the computational complexity and the *ramification problem*, i.e, the problem of predicting the consequences of actions. In general, planning far ahead has shown to lead to behaviours that do not respond very well to dynamic events [1], [8] and therefore it has here been chosen to do the opposite thing, to only plan one step, i.e., one sensing action, ahead. This is called *myopic* decision making and can be compared to a local optimization strategy as, e.g., gradient descent. A standard decision tree for the myopic decision problem is shown in Figure 2. This tree illustrates the decision problem for one sensor only.

The root decision node is called *sensory action* since this is the basic decision task of the sensor planner, namely what sensors to allocate to what modules. If there are m modules that request a sensor, the planner can choose between $m+1$ actions, namely to grant the sensor to one of the m modules (the *informed* case) or not to grant the sensor to any module (the *un-informed* case). The latter would be the case if the cost of sensing would not make up for the expected value of the information.

The *report* chance node represents the (discrete) random outcome, x_j, of the sensing action, A_i. It is noted that this chance node is absent in the un-informed case since this implies that no sensing is performed.

The second decision node, called *actuator action* is the node representing the final consequence type of action that will lead to the task completion. There could be cases where this action is not an actuator action in the traditional sense, but the term has been chosen to distinguish between (1) sensory actions that represent information gathering (and thus expenses) and (2) final, somehow productive actions marking the completion of a task and thus a utility "income".

The *state* chance node represents the world state, Z, that the world assumes "after" the actuator action has been chosen. It is of course only in the representation that the world state is set after the actuator action has been chosen. The reason for representing it this way is that it is only at this time that the world state matters and that it somehow is finally "probed". For example, if the chosen actuator action, a_k, is to drive through a door then the state of that door (open or closed) is going to be asserted.

The utilities, $U(a_k, z_l)$, in Figure 2 denote the payoff for completing a certain task. The utility is dependent on which action is chosen to complete the task, *and* the state of the world. For example, if it is chosen to drive through a door and the door is open, this should result in a high utility. However, if the door turns out to be closed, this should result in a low utility.

Seen as a whole, the myopic decision scheme can be interpreted as "given that at most one sensory action can be performed before task completion, which one should this be in order to maximize the expected utility?". It is however clear that more than one sensory action *can* be performed and that the actuator action should only be performed when the A_0 action is found to be optimal, i.e., when continued sensing can not be justified by the increase in expected utility. This means that as soon as a sensory action (other than A_0) has been chosen, the system "loops back" to the root decision node and re-evaluates the situation to see what is now the optimal choice.

The findings about the state of the world made by the purposive modules are integrated using Bayesian conditioning, and it is on these updated world state probabilities that the sensor planner bases its decisions.

Above it has been explained that the "optimal action" is chosen each time the planner has to make a decision. By "optimal action" is meant the action that will optimize the expected utility of performing the task. Choosing this action is called Bayes decision making after Bayes' Decision Rule that can be formulated as:

$$A_{opt} = \arg \max_{A_i = A_0}^{A_m} EU(A_i) \tag{1}$$

where $EU(A_i)$ is the expected utility of performing action A_i and A_{opt} is the optimal action. $EU(A_i)$ is derived in the following.

The expected utility, $EU(a_k|x_j)$, of a state chance node is:

$$EU(a_k|x_j) = \sum_{l=1}^{q} P(z_l|x_j) U(a_k, z_l) \tag{2}$$

The state probability, $P(z_l|x_j)$, is unconditioned on the sensor action A_i which is a result of the fact that we assume the sensory actions have no effect on the state of the world (the sensory actions are *non-intervening*).

From equation 2 and Bayes' Decision Rule (equation 1) the expected utility of receiving report x_j, $EU(x_j)$, can be found as:

$$EU(x_j) = \max_{a_k} EU(a_k|x_j) = \max_{a_k} \sum_{l=1}^{q} P(z_l|x_j) U(a_k, z_l) \tag{3}$$

It is assumed that the cost of performing actuator action a_k is included in (i.e., has been subtracted from)

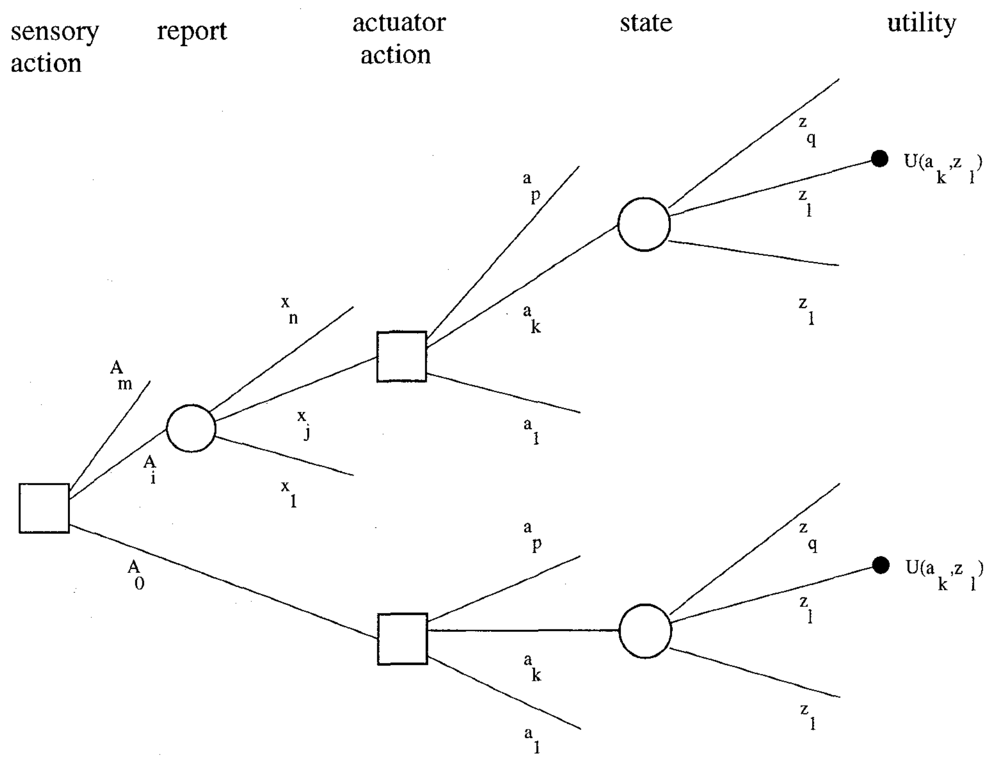

Figure 2: The standard structure of the myopic decision problem, adapted from [9]. Boxes are *decision nodes* since the path taken here is decided by the system. Circles denote *chance nodes* where Chance decides which branch is taken. There is an implicit time, or causality, axis from left to right.

$U(a_k, z_l)$. The expected utility of sensory action $A_i, i \in [1, m]$ is thus:

$$EU(A_i) = \sum_{j=1}^{n} P(x_j) EU(x_j) - C(A_i) =$$
$$\sum_{j=1}^{n} P(x_j) \left[\max_{a_k} \sum_{l=1}^{q} P_i(z_l|x_j) U(a_k, z_l) \right] - C(A_i) \quad (4)$$

where $C(A_i)$ is the cost of performing sensory action A_i. The subscript, i, on $P_i(z_l|x_j)$ indicates that when evaluating action A_i the evaluation should be made according to the conditional probabilities for the purposive module, M_i. The expected utility of action A_0 is:

$$EU(A_0) = \max_{a_k} \sum_{l=1}^{q} P(z_l) U(a_k, z_l) \quad (5)$$

What we are interested in here is, however, not the absolute value of the expected utility but rather the increase in expected utility per invested cost unit (often time) compared to not sensing. Therefore, we define the *expected interest from sample information* $(EISI)$ as:

$$EISI(A_i) = \frac{\sum_{j=1}^{n} P(x_j) EU(x_j) - EU(A_0)}{C(A_i)} =$$
$$\frac{\sum_{j=1}^{n} P(x_j) \left[\max_{a_k} \sum_{l=1}^{q} P_i(z_l|x_j) U(a_k, z_l)\right]}{C(A_i)}$$
$$- \frac{\max_{a_k} \sum_{l=1}^{q} P(z_l) U(a_k, z_l)}{C(A_i)} \quad (6)$$

The quantity in the numerator is called the *expected value of sample information (EVSI)* and can be shown always to be non-negative [9]. The expression for $EISI$ leaves the un-informed case $(EISI(A_0))$ un-defined, since $C(A_0) = 0$. To remedy this, we define $EISI(A_0) \equiv \Delta$ where $\Delta \geq 0$ can be thought of as the minimum acceptable interest rate.

Bayes' Decision Rule (equation 1) can then be re-defined as:

$$A_{opt} = \arg \max_{A_i = A_0}^{A_m} EISI(A_i) \quad (7)$$

We can now list the components that go into the decision framework.

Utilities, $U(a_k, z_l)$ These utilities reflect what it is worth taking action a_k when the world is in state z_l. The cost of performing actuator action a_k must be included. The utilities depend only on the current task.

Conditional probabilities, $P_i(z_l|x_j)$ These conditional probabilities model how likely the world is to be in state z_l given that report x_j is received from module M_i. *$P_i(z_l|x_j)$ is therefore a model of a* *purposive module*, M_i. When modeling modules is it however more natural to create the inverse relation, $P_i(x_j|z_l)$, so this will be the actual input to the framework. Bayes' Rule[2] is then employed to create $P_i(z_l|x_j)$. It is important to notice that $P_i(z_l|x_j)$ in general is a function of various state variables, i.e., the performance of a module and thus its model will in general depend on "external" variables such as distances to objects, positional uncertainties etc.

A priori probabilities, $P(x_j)$ and $P(z_l)$ The a priori probabilities express a priori information about the reports and the state of the world, respectively. One can be found from the other, however, using $P_i(z_l|x_j)$, marginalisation, and Bayes' Rule. Thus, we need only specify a priori knowledge about the world, $P(z_l)$.

Costs, $C(A_i)$ This is the cost of performing sensory action A_i.

One of the traditional problems with model based approaches as BDA is how to determine the models, i.e., the components listed above. Therefore, a few remarks concerning this are appropriate. A priori probabilities about the world, $P(z_l)$, are naturally determined from observations about the environment the robot has to navigate in. For autonomous robots cost is typically related to time, power consumption, or the computational demand of an action. In this work, the cost, $C(A_i)$, of performing a sensory action has been set equal to the time spent thereon.

Utilities reflect what is considered good and bad by the decision–maker, i.e., utilities are generally subjective. In this system, the utilities are therefore determined by the system designer and possibly modified according to experimental results until the system performance seems in accordance with the goals of the designer.

Finally, there are the conditional probabilities, $P_i(x_j|z_l)$, i.e., the models of the purposive modules, to be determined. These models are central to the decision process and should thus be developed with some care. In general, we do not believe that these models can be derived analytically and thus we have derived them empirically through experimentation with the purposive modules, which, however, is convenient in a framework where statistical data can be applied directly. It should be noted that the sensors are *not* modeled directly as is customary in, e.g., sensor fusion. Rather, it is the purposive modules that are modeled, i.e., sensors and algorithms together, which results in more empirical but also simpler models.

It is important to notice that the four model components are independent of each other, i.e., the a priori probabilities depend on the environment, the costs on the scarce resources, the utilities on the system designer's preferences (and the current task), and the conditional probabilities on the purposive modules. This eases the model derivation and means that the framework, and thus the system, becomes modular and flexible thereby facilitating easy system integration. For example, if a new sensing module should be added to the system, all that has to be done is to create a model of it and add it to the current set of model descriptions. *Nothing* else has to be changed. Similar holds for adding sensors and tasks.

The discussion here only covers the case for deciding about one sensor. The formalism can, however, be extended to cover planning for multiple sensors and arbitrary subsets of the available sensors, i.e., a purposive module can request, e.g., sonars *and* a camera and this can be handled in a coherent manner by the planner. A more detailed discussion of this is given in [6].

When a purposive module wants to request a set of sensors all it has to specify in the request is the set of sensors, Ω, how long time each sensor will be used, $\bar{t}$ (where $|\Omega| = |\bar{t}|$), and the name of the module model $P_i(z_l|x_j)$, $\mathcal{M}_i$. Thus, a sensor request, $\mathcal{R}$, is a 3–tuple, $(\Omega, \bar{t}, \mathcal{M}_i)$. From this information it is then possible for the sensor planner to calculate the expected utilities and thereby decide which of a number of requesting modules (if any) should have their requests granted.

3 Experiments

To verify that the ideas presented in this work can be applied to real world autonomous robot navigation, a series of experiments were conducted as proof of concept and to show what type of behaviour a sensor planner controlled navigation system can exhibit.

For our experiments, we use an in-door office setting, more specifically our laboratory. The task in this setting is for the robot to move from one point to another. The start and goal positions may be in different rooms. This task has been chosen since it has enough challenges to show the weaknesses and strengths of the system and since it is a standard task that allows for comparison with other systems in the literature.

The feasibility of the sensor planning method will be considered proven if the planner is capable of controlling the robot in real time. By "controlling" is meant scheduling sensors and thereby determining the actions of the robot.

3.1 The Experimental Platform, ARVID

The experimental platform, ARVID (the Autonomous Robot VIsion Demonstrator), is a Robuter–20 base equipped with the following sensor modalities: a laser

[2] Bayes' Rule should not be confused with Bayes' Decision Rule. Bayes' Rule is the well known inversion formula: $P(B|A) = \frac{P(A|B)\cdot P(B)}{P(A)}$.

Figure 3: ARVID, the experimental platform used for the experiments. The 12 DOF camerahead and the associated control computers are seen on the top of the Robuter base. The acoustic Polaroid sensors can be recognized as shiny, circular disks on the vehicle base.

line striper (i.e., a laser generating a light plane), a 12 DOF robot stereo camerahead, and a ring of 24 acoustic Polaroid sensors. ARVID is shown in figure 3.

The 12 DOF stereo camerahead has two black and white CCD cameras. Each camera is equipped with an Ernitec motorized lens with motorized control of zoom, focus, and aperture. The cameras have common neck tilt and pan but separate vergence.

The 24 acoustic Polaroid sensors each have a range of 0.1 to 6.0 meters with an accuracy of about 0.01 meters. The opening angle of the sonar beam is ± 15 degrees.

The laser line striper and the left camera of the camerahead together constitute a structured light sensor system that will just be denoted "the line striper".

On-board odometry keeps track of the approximate vehicle position. A low level vehicle driver extends this information with position uncertainty information, which is calculated as an upper bound on the vehicle position error.

3.2 The Tasks

The navigation task is broken down into subtasks by the mission planner (see figure 1). Each subtask constitutes a task for the sensor planner. Three such tasks were defined for the experiments. These were *door traversal* for traversing doors, *room navigation* used for navigation inside rooms, and *hallway navigation* which is similar to room navigation but uses a slightly different set of purposive modules.

See table 1 for an overview of which tasks use what purposive modules (described below).

3.3 The Modules

A number of purposive modules were built to solve the task. There has been made no attempt to find the "optimal" set of modules for carrying out the task. Rather, emphasis has been on creating a diverse set of modules that use different sensors and sensing strategies.

A thorough description of the modules is outside the space limitations of this paper (see [6] for details). Therefore, only a very brief description is given here. The "technical specifications" of the modules are given in table 1.

β estimator This module uses the line striper to estimate the angle, β, of the door with respect to the doorway. This is done to see if the door can be traversed by the robot. Since the laser is fixed with respect to the vehicle, the vehicle has to use an active sensing strategy, although not very elaborate, to provide a good viewpoint for the line striper. While moving, the β estimator uses the sonars to check the path.

α estimator The α estimator does the same as the β estimator, plus it also estimates the position of the robot relative to the doorway.

Door pinger This module uses a very simple, passive sonar sensing strategy for detecting if a door is closed. The module simply tries to fire a sonar through the door opening and if a return signal within certain bounds is received, the door is assumed to be closed.

Sonar doorfinder The sonar doorfinder uses the sonars and an active sensing strategy to estimate the position of the robot relative to a door.

Visual doorfinder This module uses the camerahead to do pose estimation by finding doors in the environment.

Sonar navigator This module uses sonars to do point–to–point navigation and obstacle avoidance in a reactive manner.

Visual navigator The visual navigator is the same as the sonar navigator except is also uses visual input from the camerahead to do the obstacle avoidance.

3.4 The Module Models and Utilities

For the experiments, the planner was given probabilistic models of the purposive modules reflecting the characteristics listed in table 1. The cost, $C(A_i)$, of using a module was equal to the expected time consumption in seconds. The models, $P(x_j|z_l)$, were derived empirically from experiments with the respective modules *in*

Table 1: Most important features of the purposive modules. The cost of the navigation modules depends on the distance to the goal. The reason why most modules need the vehicle is that they either use an active sensing strategy or that they depend on the vehicle not moving while doing data acquisition.

module	no	cost	tasks	sensors
β estimator	1	high (4×27 sec)	door	laser, camhead, vehicle, sonars
α estimator	2	high (4×43 sec)	door	laser, camhead, vehicle, sonars
door pinger	3	low (≤ 1 sec)	door	sonars
sonar doorfinder	4	high (2×77 sec)	door, hallway	vehicle, sonars
visual doorfinder	5	med (2×10 sec)	room	vehicle, camhead
sonar navigator	6	var	room, hallway	vehicle, sonars
visual navigator	7	var	room, hallway	vehicle, sonars, camhead

isolation, i.e., independently of the other modules in the system. Similarly were the utilities determined by adjusting them on the basis of a few experiments with the system, as described in section 2. It is important to notice that only one set of utilities per task must be determined. Moreover, the utilities for the room and hallway navigation tasks are identical.

3.5 Experimental Results

We will only outline a couple of experimental results to illustrate the behaviour of the system, since the purpose of the results is not to illustrate the performance of the robot as such, but rather the general behaviour of the system which is a result of the sensor planner distributing sensors to the various modules.

A map of the laboratory environment, where the experiments were performed, is shown in figure 4. The map also shows the result of an experimental run where the robot was told to go from room D1-103 to room D1-101. The robot was given a priori information about the walls and the doorways in the environment corresponding to the solid lines in figure 4. This explains why the robot does not first try the (too) narrow door between room D1-103 and the hallway.

The robot knows its (user specified) start position rather accurately and thus immediately starts out with navigating to the first setpoint provided by the mission planner. When the robot reaches the (large sliding) door between room D1-103 and D1-105, the pinger gets the sonars to see if it is open. When this has been found, the sensor planner decides that no further sensing is necessary before the door can be traversed, which then happens. After this, the visual doorfinder gets the sensors to locate the robot relative to the next door that must be traversed. The robot thus knowing its position, the visual navigator gets control to navigate the robot to the setpoint in front of that door. Here, the α estimator gets the sensors to estimate the exact door position and the state of the door (open/closed). Also, the pinger, which is a very cheap module to run, gets the sonars to add extra certainty that the door is open. Having established this, the door is traversed. From here, the picture should be quite clear, except maybe from the fact that the sonar doorfinder gets the sensors while the robot is underway in the hallway. This is due to the fact that the vehicle's position uncertainty rises while it is navigating down the hallway, causing the *EISI* of doing position estimation to rise and point–to–point navigation to fall. Thus, at some instant, the sensor planner decides to (temporarily) preempt the visual navigator and give the sonar doorfinder the sensors in order to improve the robot's position estimate.

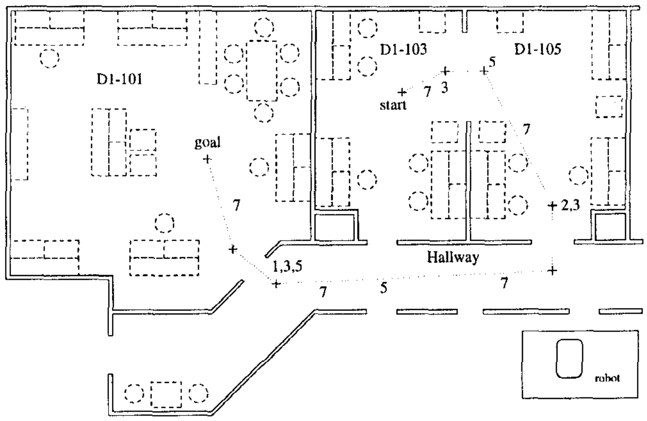

Figure 4: The part of the laboratory where the experiments were performed. Solid lines are walls, dashed lines obstacles, and dotted lines show the route (not the trace) taken by the robot. The crosses are setpoints generated by the mission planner, except from the start and goal points that were user specified. Numbers correspond to the modules getting the sensors from the sensor planner (see table 1 for corresponding modules). The insert in the lower right corner shows the robot to scale. See text for further explanations.

In an identical experiment, but with the door from room D1-105 to the hallway closed, the robot shows the same behaviour up until the point where it is found that the door is (probably) closed. At that point, the sensor planner decides that it is not feasible to do more sensing and that the sub–task should be aborted. This is then reported to the mission planner. This shows another feature of decision–theoretic planners, which is that they can actively reason about when to stop a task, i.e., they do not treat a "mission impossible" as an exception, as most traditional navigation systems do.

It is important to notice that there nowhere in the system are any rules or the like that dictate when a given module should have the necessary set of sensors. It is all decided on–line by the sensor planner that continuously evaluates the sensor requests it gets from the purposive modules. In the experiments reported, requests were evaluated every 4 seconds, but this is by no means as fast as the system allows, rather it is a practical frequency when doing experiments.

4 Conclusion

In this work, a decision–theoretic sensor planner for doing multisensor planning and integration was designed, built, and tested.

The decision–theoretic framework allowed the sensor planner to schedule sensors to purposive modules in a rational way, based on a well–understood statistical background and using utility theory. This, in turn, completely decides the behaviour of the robot (at some level), since granting and denying the purposive module the sensors/actuators they need corresponds to enabling and disabling them, respectively. Evidence provided by the purposive modules was fused using a Bayesian probabilistic approach which is theoretically well–founded.

The decision–theoretic framework furthermore provided a highly modular way of describing sensors, modules, and tasks. This greatly facilitates easy system integration, since new sensors, modules, and tasks can be added without making changes to the existing system.

Experiments with a real robot showed that the sensor planner was capable of making rational on–line decisions about what sensors to use for what purposes. This included deciding how much to sense, i.e., when to stop sensing, when to abort a task, and when to preempt an on–going sensing operation in favour of another.

What this implies is that the probabilistic models, created from experiments with the modules, apparently are sufficient for making rational decisions about sensor allocation. Also, it seems confirmed that myopic decision making is sufficient for making decisions at this rather low level.

We think that this altogether shows that decision–theoretic planning is a feasible and promising new technique for doing real–time sensor planning and integration for autonomous robot navigation in partly known, dynamic environments.

Acknowledgments

We would like to thank Dr. M. Mintz and Dr. R. Bajcsy for their valuable discussions and comments.

References

[1] Philip E. Agre and David Chapman. What are plans for? In Pattie Maes, editor, *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, pages 17–34. MIT/Elsevier, 1990.

[2] Ruzena Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):996–1005, August 1988.

[3] H. I. Bozma and J. S. Duncan. A game–theoretic approach to integration of modules. *IEEE PAMI*, 16(11):1074–1086, November 1994.

[4] Erann Gat. Robot navigation by conditional sequencing. In *Proc. of the 1994 IEEE Int. Conf. on Robotics and Automation*, pages 1293–1299, San Diego, California, May 1994.

[5] Gregory D. Hager. *Task–Directed Sensor Fusion and Planning: A Computational Approach.* Kluwer Academic Publishers, 1990.

[6] Steen Kristensen. *Sensor Planning with Bayesian Decision Analysis.* PhD thesis, Laboratory of Image Analysis, Aalborg University, August 1996.

[7] J. Lindner, R. R. Murphy, and E. Nitz. Learning the expected utility of sensors and algorithms. In *Proc. of the MFI'94*, pages 583–590. IEEE, 1994.

[8] David W. Payton. Internalized plans: A representation for action resources. In Pattie Maes, editor, *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, pages 89–103. MIT/Elsevier, 1990.

[9] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* The Morgan Kaufmann Series in Representation and Reasoning. Morgan Kaufmann, San Mateo, California, 1988.

[10] Stuart Russell and Eric Wefald. *Do the Right Thing: Studies in Limited Rationality.* The MIT Press, Cambridge, Massachusetts, 1991.

[11] Reid G. Simmons. Structured control for autonomous robots. *IEEE Transactions on Robotics and Automation*, 10(1):34–43, February 1994.

[12] Michael P. Wellman. Challenges of decision–theoretic planning. In *Working Notes from the AAAI Spring Symposium on Foundations of Automatic Planning*, March 1993.