AIAA-94-1257-CP

A ROBOT CONTROL FORMALISM BASED ON AN INFORMATION QUALITY CONCEPT

A. Ekman, A. Törne, D. Strömberg
Dept. of Computer and Information Science, Linköping University
FOA (National Defence Research Establishment)
Linköping, Sweden

Abstract

The efficiency of an autonomous robot navigating in indoor environments depends crucially on the ability of the robot to exploit spatial relationships extracted from perceptions of its environment. Smith, Self, and Cheeseman[23] describes a formalism based on Kalman filter theory, where perceptions from different locations can be combined to improve the accuracy of the robot pose estimate. We argue that while accuracy is an important property of perceptions of the robot state, a more important property of perceptions of the environmental state are their temporal and spatial range of applicability, which will be referred to as perceptual relevance. This paper introduces a relevance measure based on Jaynes maximum entropy principle, measuring the relevance of a spatial description of the robot environment. The conjunction of accuracy and relevance is denoted information quality. A formalism based on the information quality concept is developed for the class of one-agent applications, for which the formalization of the dependency between perceptions and actions of a robot is straightforward.

1 Introduction

A robot can be viewed as a controller, the purpose of which is to transform the current system state into a goal state. After having executed the action sequence, the system state should be closer to a goal state. If we by "world" denote the conjunction of robot and system, this paper is based upon that essentially three issues determine the performance of the state transformation process: 1) The accuracy and relevance of the robot's perception of the world state, 2) The robot's capability to find an action sequence that forces the current model state into a desired goal state, and 3) The precision of the transformation from abstract to physical actions.

The behavior of the robot corresponds to what actions the robot selects to execute in a particular situation. For sensorless robots, behavioral information in the form of action sequences are given a priori, and may not change due to external events during operation. Although this is a straightforward way to implement robot behavior, the robot requires a well-defined working environment where the properties of each object must be accurately specified. In effect, all information is given to the robot a priori, and a major problem is to maintain a configuration of the working environment that is consistent with the specification.

More flexibility is achieved if the robot is capable of acquiring information about the true configuration of the working environment during operation. Robots capable of acquiring information during operation may be classified as being either reactive[1,2] or deliberate. While reactive behavior commonly is hard-wired into the robot, deliberate behavior is exhibited by robots maintaining an explicit world model.

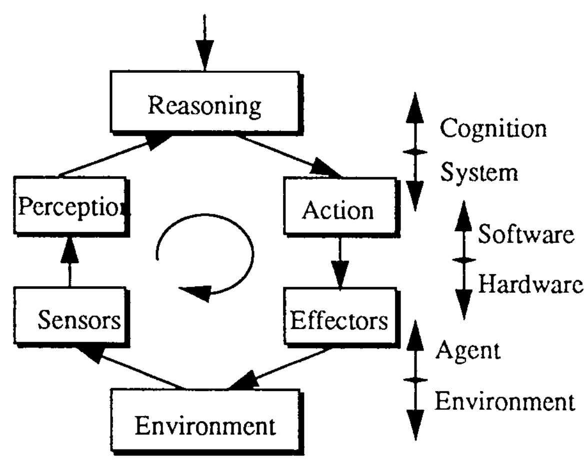

Fig. 1. Structure of a deliberate robot

Obviously, reactive robots are inherently autonomous, while deliberately behaving robots may be anything from autonomous to tele-operated. In the autonomous case the explicit model is implemented in the robot software, while in the tele-operated case the model exists in the mind of the expert controlling the robot. Issues affecting the performance of a deliberatively behaving robot will be addressed in this paper. Throughout the paper, except where explicitly stated, the only assumption being made concerning the deliberately behaving robot is that it possesses an explicit world model.

In *Fig.1*, a useful way to describe the structure of a deliberate robot is presented. By splitting the graph horizontally at increasing heights, one gets the following sequence of dichotomizations: environment/agent, hardware/software, system/cognition. The figure also illustrates the cyclic processing of perceive-reason-act which starts when a task has been given.

A distinguishing feature among deliberate robots is the degree of human interaction, spanning from autonomy to tele-operation. For an autonomous robot the "reasoning" module (see *Fig.1*) corresponds to a computer program while for a tele-operated robot the it corresponds to the human operator. Albeit having this difference, all deliberate robots need high quality information in order to do proper inference. By keeping the information quality above some pre-defined level, the likelihood of erroneous inference is kept sufficiently low. In this paper we will develop means for preserving the information quality, which are applicable to a large class of deliberate robots.

The formalism developed in this paper is based on an assumption of a one-agent application. The formalism might however be extended to cover many-agent applications as well. One distinguishing feature between one- and many-agent applications is that while failure to execute a plan in a one-agent application is caused either by poor information or by poor control, for many-agent applications an additional cause of plan execution failure are actions executed by other agents. Without doing any further elaboration on the class of many-agent applications, it suffices to notice that more powerful models must be developed and that real-time constraints become crucial[16].

The robot uses the world model for interpreting the present situation. Therefore, it is of great importance that the model discrepancy is small. On the other hand, a too detailed model suffers from high time and space complexity. The difficulty to satisfy these somewhat contradictory constraints is one reason why many deliberate robots are either too slow or too error-prone. At the core of the problem are the issues of uncertainty and complexity. Typically, reducing the complexity of a model increases the uncertainty and vice versa. However, by using application-specific heuristics, it is possible to suppress world properties of minor importance, thereby simplifying the model. In this way a model with both low discrepancy and moderate complexity can be establish. For example, in the mobile robot case, a 2D (global) map suffice for robot navigation, while a 3D (local) map is needed for many object manipulation tasks. By using this heuristics, a 2D/3D composite model is created, with a fair trade-off between complexity and discrepancy. The problems to represent and reason under uncertainty have been addressed in several papers[10, 11, 12, 25]. Although this is an important research area, the work do normally not consider the problem of uncertainty and relevance maintenance, in particular not when the uncertainty and relevance varies in time and space.

Having developed an appropriate model, the next question is why and when the model should be updated with fresh information? In order to answer the first question, we recapitulate that the model should have limited space and time complexity. This inevitably leads to an information loss. Accordingly, situations may occur where ignorance of some world property will result in robot malfunction, although such situations may be very rare. Crucial for the prevention of robot malfunction is to maintain a low model discrepancy, since the next action to execute is determined by the model interpretation. Obviously, the penalty for having a model of low complexity is that it must be updated frequently to keep the discrepancy low.

When to update the model depends on how low one want to keep the likelihood of robot malfunction. While it is obvious that a high model discrepancy increases the risk of robot malfunction, it is very difficult to calculate the probability of robot malfunction as a function of the model discrepancy. The reason for this is that in order to calculate the discrepancy the model state must be compared to the world state, which contains an infinite number of elements. An approach to this problem is to introduce a threshold value for each perception, representing the minimum permitted information quality of the perception. After each executed action during the execution of an action sequence, the robot checks that each perception has a quality exceeding the corresponding information quality threshold. If this is not the case, a sensing action is executed to increase the information quality of the perceptions. Otherwise, the next action in the sequence is executed.

2 Model - world duality

The correspondence between the model and the real-world is established through the following definition.

Definition 2-1 To find a solution to a real-world problem is analogous to the abstract problem of finding a path, p, satisfying the predicate Q(p), from an initial state to a goal state in a model state space, S.

For instance, Q(.) might represent the predicate "shortest(.)", "least expensive(.)", or "most safe(.)".

Although *Def. 2-1* is rather general, a straightforward interpretation within the framework of this paper is possible. The problem space, S, corresponds to the space of possible model states. The path, p, corresponds to a sequence of actions, while Q(t) is interpreted as "Information Quality high enough along the path".

Ideally, the abstract plan is in perfect agreement with the solution to the real-world problem. In practice, though, this is rarely the case and the robot must perform plan validation iteratively during the execution. The efficiency in the detection of plan failures and the subsequent re-planning is closely related to the performance of the robot, and could accordingly be taken as a measure of the same.

Each cycle of processing in *Fig.1* corresponds to the transition between two states in the problem space - ideally towards a goal state.

This duality between the cyclic processing of the physical system and the problem space transitions suggests the possibility to analyze plans in the problem space before executing them on the physical system.

3 Concepts

One-agent applications

A large class of robotic applications are one-agent applications. In a one-agent application, it is assumed that all changes to the system state are caused by the actions of the sole agent. Consequently, for one-agent applications it is straightforward to formalize the dependency between actions and perceptions of an agent.

World

The world is composed of two entities, agent and environment. This is in accordance with the one-agent application assumption. The agent may correspond to either an autonomous or a tele-operated robot.

Sensing

In order to gather information about the world, the agent must use sensors. Sensors measure either the state of the agent or the state of the environment. Sensing, σ, maps the world state to perceptions, which in turn can be mapped to a more abstract perception, or be combined with previous perceptions to give a more accurate perception.

Perception

Perceptions provide descriptions of details of the world. To distinguish between perceptions with different properties, perceptual classes are introduced. Within each perceptual class, perceptions are distinguished by their creation time. Accordingly, a perception from perceptual class *i*, created at time k will be denoted $p_{i,k}$. Perceptions acquired by the agent up to time k, where $k \in N$, is represented by the perception vector P_k

$$P_k = (p_1, p_2, \ldots, p_r) \tag{1}$$

where r is the number of perceptual classes and p_i corresponds to a set of acquired perceptions of the *i*:th perceptual class, that is

$$p_i = \{p_{c,k-k_1}, p_{c,k-k_2}, \ldots, p_{c,k-k_e}\} \tag{2}$$

where $0 \le k_e < \ldots < k_2 < k_1 \le k$.

Perceptions are created either by using data from one or more sensors or by combining previous perceptions into a more abstract or more accurate perception. To create more abstract perceptions, feature extraction algorithms are applied, while to create more accurate perceptions spatio-temporal dependencies among the previous perceptions are exploited.

Action

The set of actions that the agent can execute is denoted A. Actions, $a_i \in A$, are used for changing the world state. Actions can be identified as being of either manipulatory, navigational, or sensing type. Accordingly, three action classes are introduced. Actions from the manipulatory action class change the state of the manipulator, which in some cases also changes the state of the environment. Actions from the navigational action class correspond to the movement of the agent to a new location. Sensing actions correspond to the acquisition of information about the world state.

Reasoning

Given perceptions of the world state, the agent performs reasoning, ρ, to determine what action sequence to execute.

Effectuating

The robot is using effectors to translate actions into changes of the world state. Effectuating, ε, therefore maps actions to world state changes.

Functional description of an agent

In a one-agent application, it is possible to describe perceptions as functions of actions or vice versa. This is illustrated in *Fig.2*.

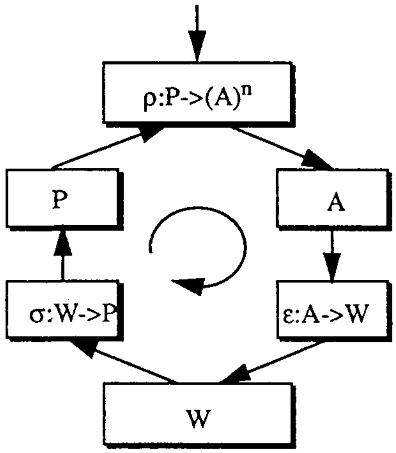

Fig. 2. Functional description of an agent

Following is the interpretation of *Fig.2*: Perceptions, P, are mapped by reasoning, ρ, to an appropriate action sequence $(A)^n$. The effectuating, ε, maps each action to a new world state, W. Sensing, σ, maps the world state to perceptions, P, .

Let C_k denote k:th reasoning-effectuating-sensing cycle, that is

$$C_k \equiv \sigma_k\left(\varepsilon_{k,n_k}\cdots\varepsilon_{k,2}\varepsilon_{k,1}\right)\rho_k = \sigma_k\varepsilon^{n_k}\rho_k \qquad (3)$$

(3) has the following interpretation:

By reasoning, ρ_k, the robot decides to execute an action sequence consisting of n_k actions, which are send to the effectuating, ε. Thereafter the robot uses sensing, σ_k, to acquire information about the resulting world state. Accordingly, the k:th perception can be written as:

$$P_k = C_k(P_{k-1}) = C_k C_{k-1}\cdots C_1(P_0) \qquad (4)$$

In applications where the agent corresponds to an autonomous robot it is possible to describe the mapping (ρ) explicitly by a mathematical function. Accordingly, it is possible to, given a perception vector P_i and the mapping ρ, determine what action sequence will be executed consequently. This is possible since machine reasoning consists of well-defined deterministic operations.

For applications where the agent corresponds to a teleoperated robot (or where man/machine cooperative decision making[15] is used), the value of the mapping, ρ, corresponds to the action sequence (partly) decided upon by the human expert. Since it is hardly possible to establish a deterministic model of the reasoning process involved, the true mapping function, ρ, is (partly) unknown for teleoperated robots.

However, in either case, the impact of an action on the world state will be the same. This implies that the perceptions dependency on actions is invariant under different reasoning approaches. Consequently, this suggests that the same principles for maintaining the information quality can be utilized for both agent types.

4 Information quality

Having introduced a set of perceptual classes, a measure of the information quality of each perception is needed. The measure should reflect both the accuracy and relevance of a perception. External perceptions are commonly maintained at three abstraction levels to reduce complexity and enable inference. The lowest level contains numerical, the second geometric, and the third symbolic information respectively[3]. Consequently, the information quality measures at two distinct abstraction levels will differ. In this paper, we will consider only the two lowest abstraction levels. Internal perceptions, describing the state of the robot, are often of limited complexity. Hence, only a numerical model is needed. For example, if ignoring dynamical properties, the state of a mobile robot equipped with a manipulator arm can be described by a 9-dim. vector (3 dim. for describing the robot pose and 6 dim. for describing the position and orientation of the end effector).

Accuracy

Perceptual accuracy is a static attribute that is determined when the perception is created. It is static because no future event can affect the accuracy of an already made measurement. If perceptions are treated as vectors, one common way to represent accuracy for numerical perceptions are by the corresponding covariance matrices. In this way, Kalman filtering techniques can be utilized to generate accurate state estimates. A classical paper on this is the one by Smith, Self, and Cheeseman[23] which provides a framework for handling robot positional uncertainty, and may be extended to also handle robot arm positional uncertainty.

Relevance

Besides being more or less accurate, a perception will also be more or less relevant. For example, a temperature measurement from one part of a building tells very little about the temperatures in other parts of the building, although the temperature was measured with high accuracy. Furthermore, after some time that particular temperature measurement tells very little about the current temperature even in the position where it was obtained. These two facts correspond to the limited spatial and temporal applicability range of perceptions. Thirdly, assume that the agent that did the previous temperature measurement decides to open a window. Provided it is a temperature difference between the inside and outside of the building, this action will reduce the relevance of the previous measurement. Thus, manipulatory actions executed by the agent is another cause of variation in relevance.

While the first and third example illustrate the dependency of the relevance on the executed navigational or manipulatory actions of the agent respectively, the second example illustrates that the application is not an ideal one-agent application. Since no real-world application is an ideal one-agent application, a perception aging function, monotonously decreasing with time, must be used.

The presented examples describe relevance for a numerical perception. In *Section 7* a relevance measure for geometric perceptions, based on the maximum entropy principle, will be described.

Information quality measure

Our approach assumes a bidirectional dependency between perceptions and actions. This is an extension to the approach by Erdmann[8], who assumes a unidirectional perception/action dependency.

Section 3 introduced a description of the k:th perception as the result after a reasoning-effectuating-sensing sequence has been executed after an initial perception P_0 has been created.

The information quality of a perception vector P_k is denoted $QI(P_k)$. The evaluation of the information quality is as

$$QI(P_k) = (q_1(p_1), q_2(p_2), \dots, q_r(p_r)) \qquad (5)$$

where $q_i(p_i)$ is evaluated as

$$q_i(p_i) = max\ \{q_i(p_{i,k_x})\} \qquad (6)$$

where k_x is iterated over all creation times of the perceptions in the i:th perceptual class.

Using (3) and (4), (5) can be rewritten as

$$QI(P_k) = QI\left(\sigma_k \varepsilon^{n_k} \rho_k P_{k-1}\right) \qquad (7)$$

The index n_k must be selected as to satisfy the condition

$$\left(QI\left(\sigma_k \varepsilon^{n_k} \rho_k P_{k-1}\right) > T\right) \wedge QI\left(\sigma_k \varepsilon^{n_k+1} \rho_k P_{k-1}\right) < T \qquad (8)$$

is satisfied, where T is the information quality threshold value. This condition means that executing n_{j+1} actions in a sequence will result in an information quality value below the threshold value. Since robot malfunction corresponds to the failure to execute a particular action, the threshold value must take into account that for some actions an action failure is harmless while for irreversible hazardous actions a successful action execution is essential. Thus, to permit the execution of an action, the information quality of the perceptions must exceed its corresponding threshold value, T.

Predictions

Knowledge about the statistical properties of the effectors enables the prediction of the environmental state resulting from an executed action. Furthermore, with information about the statistical properties of the sensors, it is possible to predict how the resulting environmental state should be perceived if a new perception where obtained.

Given a perception and an action to execute, it is possible to predict what should be perceived after the action has been executed. The uncertainty in this prediction is determined by the statistical properties of the stochastic mapping. If this uncertainty is considered to be low enough, an additional action may be executed, with a corresponding new prediction. The uncertainty in this new prediction is higher than for the previous prediction. This can be iterated as long as the uncertainty in the prediction is low enough. When, at last, the prediction will be too uncertain, a new perception must be generated.

The agent can predict the true world state resulting after each executed action by using knowledge about the most probable outcome of the action. The resulting prediction of the world state after n_a actions (after time k) have been executed is denoted $\hat{P}_{k,a}$. Thus,

$$\hat{P}_{k,a} = \varepsilon^{n_a} \rho_k P_{k-1} \quad (9)$$

The information quality of the prediction is denoted $QI(\hat{P}_{k,a})$, that is

$$QI(\hat{P}_{k,a}) = QI\left(\varepsilon^{n_a} \rho_k P_{k-1}\right) \quad (10)$$

where n_a must satisfy the condition

$$\left(QI\left(\varepsilon^{n_a} \rho_k P_{k-1}\right) > T\right) \wedge \left(QI\left(\varepsilon^{n_a+1} \rho_k P_{k-1}\right) < T\right) \quad (11)$$

The evaluation of (11) is done in a way similar to (6).

5 Sensor planing

Both autonomous and tele-operated robots require a criterion for when to acquire new information. According to the previous discussion this is necessary when the quality of the information is so low that the reasoner is likely to draw the wrong conclusions about the situation. This corresponds to some perception being too inaccurate or too irrelevant. A sensor planning algorithm should keep track of the resulting accuracy and relevance of the perceptions as actions are executed, and enforce a sensing action if the accuracy or relevance has become too low.

In sensor planning, an important difference between autonomous and tele-operated robots is their type of reasoning (ρ). For autonomous robots, a well-defined mapping from perceptions to actions is used, while for tele-operated robots, a human expert decides upon what action to execute next. In effect, autonomous robots may use (8), where the resulting information qualities are calculated before the sequence is executed. For tele-operated robots, the situation is different. Here, the sensor planning algorithm has no knowledge about what action will be executed next. Therefore, it must keep track of the information quality values during operation and stop the robot if the information quality has become too low. This implies that (11) should be used instead.

6 Maximum entropy methods

The information theoretical entropy concept has been applied in a variety of scientific disciplines. For a survey of entropy optimization techniques, we refer to Kapur and Kesavan[14]. In robotics, Saridis[20, 21, 22] and Valavanis[24] have described robot systems that use the concept of entropy as a global performance measure. In their approach, optimal robot behavior is achieved through the minimization of the total system entropy. Sanderson[19] uses the entropy concept to describe the complexity of differently shaped geometrical objects, measured in bits. Finally, the thorough study of relations between information theory and search theory conducted by Pierce[18] has inspired the development of the relevance measure for the case study described in *Section 7.*

7 Case Study - Sensor planning on a tele-operated indoor intervention robot

This case study demonstrates how the previously introduced concepts can be instantiated in a real-world application for a mobile robot equipped with a manipulator arm. The robot obtains information about the external state by using laser and video cameras. Information about the internal state is obtained through odometry and angle counters on the joints on the manipulator arm. This robot type is very general, but by constraining either the mobility or the manipulability capabilities, more restricted robot types are obtained. In the case study, a representative set of perception and action classes are introduced. Furthermore, to properly control the system, the robot must have a system model with low discrepancy. Because of limited computational power, the model must have as low complexity as possible. In order to establish a compact but yet useful model, it is important to exploit structure in the robot operations. In the case study, a composite 2D/3D model developed for the discussed robot type is elaborated.

Robot operation

During operation, the state changing actions executed by the robot can be classified as being either navigational or manipulatory. Navigation corresponds to the movement of the robot to a new location, while manipulation corresponds to the reorientation of the manipulator arm (the purpose of the manipulator arm is to change the state of the environment). This suggests the introduction of the two action classes A_N and A_M, denoting the navigational and manipulatory action class respectively. When the robot executes a sequence of navigational actions, it is said to be in navigational mode. Similarly, when the robot executes a sequence of manipulatory actions, it is said to be in manipulatory mode.

Assuming a one-agent application, the world consists of two entities, agent and environment respectively. The state of the agent will be denoted the internal state while the state of the environment will be referred to as the external state. Accordingly, measurements of the internal state will be referred to as gauging, reflecting the contact-type of internal measurements, while measurements of the external state will be referred to as sensing.

This gauging/sensing action class division is natural. First, the division conforms to the two-mode robot operation since only the internal state changes when in navigational mode. Second, the complexity of the internal state description is much lower than the complexity of the external state description. Consequently, a division is necessary of the external state description into representations at different abstraction levels. As a comparison, it is possible to establish a basic description of the internal state using a 9-dim. vector, while sensor data of the external state may well contain 10,000 data points.

The introduced four action classes is listed in (12):

$$\{A_S, A_G, A_N, A_M\} \tag{12}$$

with the following respective interpretation:

- A_S corresponds to Sensing actions, which measure the external state.
- A_G corresponds to Gauging actions, which measure the internal state.
- A_N corresponds to Navigational actions, which change the global state.
- A_M corresponds to Manipulatory actions, which change the local state.

Perceptual classes

Having introduced action classes in the previous section, this section presents a step-wise partitioning of the world description into a set of perceptual classes.

Perceptions describe different aspects of the world and thus represents a world model. Typically, the data from one or more sensor is refined and transformed into a perception (a perception is similar to *virtual sensor*[4] and *logic sensor*[15] respectively). In turn, a perception may be used to generate a more abstract perception, or be combined with previous perceptions to generate a perception with higher accuracy.

In the previous section, the two information acquiring action classes A_G and A_S was introduced. The corresponding perceptions resulting from information acquiring actions from respective action class are denoted internal and external perceptions respectively. As mentioned, the mobile robot can be viewed as being in either a navigational or a manipulatory operation mode. This observation suggests the partitioning of the perceptions into global vs. local perceptions. Global perceptions are perceptions that provide vital information when in navigational mode, while local perceptions are perceptions that provide vital information when in manipulatory mode. This leads to the division of the perceptions into four perceptual classes (Table 7-1).

	Local	Global
Internal	$P_{I,L}$	$P_{I,G}$
External	$P_{E,L}$	$P_{E,G}$

TABLE 7-1 Perceptual Classes

For the robot at hand, examples of perceptions belonging to each perceptual class are suggested below:

- $P_{I,L}$: The pose of the robot arm
- $P_{I,G}$: The pose of the robot
- $P_{E,L}$: The pose of a manipulable object
- $P_{E,G}$: The spatial description of the building

2D representations of the global perceptions P_{EG} and P_{IG} suffice in most applications since the robot moves on almost flat surfaces and detected obstacles may be assumed to have infinite height. Having infinite height implies that it suffice to describe their projections on the 2D plane. Thus, navigational actions are described within a 2D model, where the pose description contains three parameters (two for position and one for orientation).

General manipulatory actions involve manipulation of objects arbitrarily oriented in 3D space. Since the range of the robot arm is limited, a 3D model will be reasonable. For each object that is to be manipulated, a 3D description of its closest environment is used. The local 3D descriptions are connected to the global (navigational) 2D model, thus providing a composite 2D/3D model.

Accuracy

Smith, Self, and Cheeseman's formalism[23] may be applied for estimating the internal state, consisting of the robot pose (x, y, φ), where x, y describe the position and φ the orientation of the robot, and of the position and orientation of the robot arm $(x, y, z, \phi, \theta, \psi)$, where x, y, z corresponds to the position and ϕ, θ, ψ corresponds to the orientation of the end effector using euler angels. The introduced information quality threshold value, T, will in this case correspond to a matrix where the elements should not be exceeded by the corresponding elements in the covariance matrix for the 9-dim. vector describing the compound internal state.

Relevance

The relevance of a perception is affected by spatial and temporal factors. While the former concerns the topology of the environment, the latter provides means for compensating for the one-agent assumption in applications actually containing several agents. Typically, the aging of the perceptions is modelled by simply scaling the perceptions with a monotonously decreasing weight function. The spatial factors affecting the relevance of perceptions of the global external state will be elaborated below, yielding a relevance measure applicable to laser range data*, although it may be modified to be used for sonar range data instead.

An assumption being made is that the gathering of spatial information is costly in time and resources. Therefore, new spatial information should be acquired only when absolutely necessary. To determine when this is the case, a measure indicating the relevance of a perception is needed.

Consider the situation in *Fig.3*. Spatial information obtained from position p1 will lack information about the region N.E. of p2. The sector indicated in the figure will accordingly be referred to as a *hidden sector*. The hidden sector indicates that the spatial information from p1 is not as relevant for describing the environment of p2 as it is for describing the environment of p1. If the only spatial information available is the one from p1, it is impossible to tell whether the spatial information not contained in p1 is important or not. Let R denote the statement "Relevant information" and ¬R the statement "Not relevant information" (e.g, in search operations, indications of the object being searched for are considered relevant). Letting $p_i(R)$ denote the probability that scan direction i contains relevant information, then Laplace's principle of insufficient reason states that $p_i(R) = p_i(\neg R) = 1/2$ if no prior information is available.

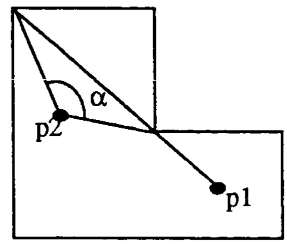

Fig. 3. Relevance of spatial information, hidden sector

Assume that the laser measures range in M consecutive directions in a horizontal plane. Then M probability functions are introduced. It is assumed that after a measurement has been made, it is clear whether a particular direction contains relevant information or not. Thus, after a measurement $p_i(R) \in \{0, 1\}$. For large M, the angle resolution will be high enough to justify the assumption that no new information will come out of an additional measurement of a previously scanned region.

In the case of *Fig.3* this implies that the only new information coming out of a measurement from position p2 is information about the region that is not described by the measurement from p1. Assuming the angle between consecutive scan directions to be $\Delta\varphi$ and that the hidden sector has angle α, a scan from p2 will provide new information from $d = \alpha/\Delta\varphi$ directions. Letting $\bar{p}$ and $\bar{q}$ denote the ensemble of probability functions corresponding to the measurements from p2 and p1 respectively, then the amount of new information obtainable from p2 given the information from p1 is

$$H(\bar{p}|\bar{q}) = -\sum_{i=1}^{d}\sum_{j} p_i(j)\log(p_i(j)) \quad (13)$$

where $j \in \{R, \neg R\}$. Using the maximum entropy principle, the sum in (13) is equal to $d \cdot \log 2$, which is the amount of new information obtained if scanning from position p2, or equivalently stated the amount of information about p2:s environment not included in the spatial description from p1. As is seen, the obtained information is proportional to the width of the hidden sector.

The amount of information obtained from the initial scan is $M \cdot \log 2$. No later scan will give that much information. This suggests that the information quality thresholds for the external perceptions should belong to the interval $[0, M \cdot \log 2]$.

* Using range data from either laser or sonar is a common way to build up maps of indoor environments[3, 5, 9]

In the general case there are k hidden sectors, corresponding to k sub-intervals in the interval 1,2,..., M that provide new information (see *Fig.4*).

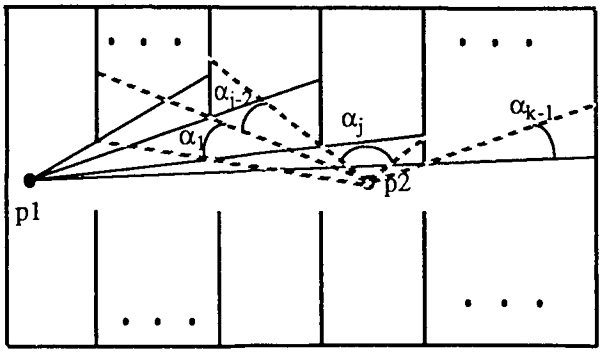

Fig. 4. k hidden sectors

Extending (13) to cover k hidden sectors yields

$$H(\bar{p}|\bar{q}|) = -\sum_{h=1}^{k}\sum_{i=1}^{d_h}\sum_{j} p_i(j)\log(p_i(j)) \quad (14)$$

which, according to the maximum entropy assumption is equal to

$$\log 2 \cdot \sum_{h=1}^{k} d_h \quad (15)$$

Notice that if the laser range data is segmented by using polyline segmentation[6], the hidden sectors can be calculated directly, since the information needed to calculate the hidden sectors is information about the coordinates of the endpoints of the line segments.

Information quality

As mentioned, there is a close connection between perceptions and actions in one-agent applications. This section describes this dependency. *Fig.5* captures the dependency between the action and perception classes. The actions positive/negative influence on the information qualities of the perceptions are indicated by the "+" or "-" superscript.

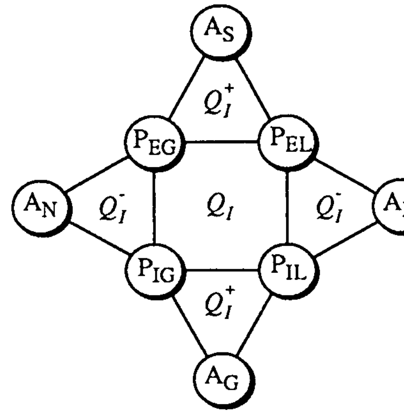

Fig. 5. Perception/Action class dependency

In *Fig.5*, the dependency within a pair of perception/action classes is bidirectional. The information qualities of the perceptions are either increased or decreased depending on what actions that have been executed. A similar dependency is present in the opposite direction since the information qualities of the perceptions must be sufficiently high in order for the reasoner to perceive the situation correctly and select proper actions. In *Fig.5*, the perceptual classes have the following interpretations:

- Internal global state (P_{IG}) corresponds to the position and orientation of the robot, and is measured by odometry.
- Internal local state (P_{IL}) corresponds to the position and orientation of the end effector (robot hand), and is measured by combining measures from angle counters on the joints of the robot arm.
- External global state (P_{EG}) corresponds to a line segment description of the robot environment. The description is created by line segmentation[5] of laser data, and is measured by a laser range finder.
- External local state (P_{EL}) is measured by vision (TV camera).

For navigation tasks, the operator uses the line segment description of the robot environment, while for manipulatory tasks views from TV cameras are used. The perception/action dependency may then be illustrated by the dependency graph in *Fig.6*, which emphasizes the two-mode operation. The nodes in the leftmost column correspond to sensors, the nodes in the middle column correspond to perceptions, and the nodes in the rightmost column correspond to actions.

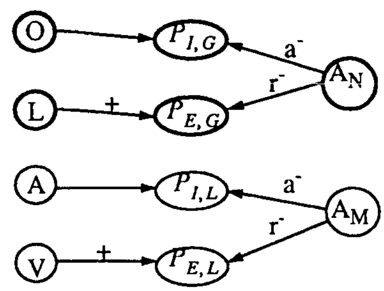

Fig. 6. Information quality dependency graph

Odometry (O) gauges the internal global state, the description of which is stored in $P_{I,G}$. New measurements cannot reduce the increasing positional uncertainty as the robot moves, although Kalman filtering techniques can reduce this problem. For this reason, the arrow from O to $P_{I,G}$ in *Fig.6* is unlabeled. Laser (L) senses the external global state, the description of which is stored in $P_{E,G}$. Acquiring new laser data improves the relevance of $P_{E,G}$, indicated by a "+"-sign on the arrow between L and $P_{E,G}$ in *Fig.6*. Angle counters (A) gauge the internal local state, the description of which is stored in $P_{I,L}$. Although new measurements cannot reduce the uncertainty in $P_{I,L}$, the uncertainty will not increase monotonously as the uncertainty corresponds to lashes in the joints. Vision (V) senses the external local state, the description of which is stored in $P_{E,L}$.

In this application, the external local states corresponds to local descriptions of particularly interesting regions in the environment (e.g. regions containing manipulatable objects). The relevance measure introduced in *Section 4* is intended for external local perceptions (which are 2D). A similar measure for the external local perceptions (which are 3D) may be developed based on the concept of hidden cones although this is not described in this paper.

Navigational actions, A_N, corresponds to the transport of the robot to a new position in the environment. The accuracy of the internal global perception $P_{I,G}$ is decreased due to accumulation of positional uncertainty. Also, the relevance of the external global perception is decreased, in accordance with the fact that the applicability of the external global perception may decrease as the robot moves away from the position at which the external global perception was generated.

Manipulatory actions, A_M, corresponds to the movement of the manipulator arm, which may affect the accuracy of the internal local perception and/or the relevance of the external local perception.

Finally, as mentioned, the increased uncertainty in the estimation in the robot position is impossible to eliminate by solely using odometry. However, it is possible to combine sensing and gauging actions to improve the information quality of $P_{I,G}$.[7].

Platform evaluation

Having a method for maintaining the information quality at an acceptable level, the problem arises what threshold values to use. By solving a task, typical for the application at hand, with distinct information quality threshold value settings, the setting providing the best trade-off between safety and speed should be used. To compare different robot configurations (i.e., using different sensors and effectors), the optimal parameter setting for each robot is determined whereafter their optimal performances are compared. If an information quality value of a perception could be held close to zero without affecting the performance, this suggests that the corresponding perception is of little use in the application and might be omitted. By assigning each action (a) a cost C(a), for example corresponding to execution time, the cost to use a particular platform is easily obtained by summing the cost of all actions in the action sequence that was executed during the mission. This value could then be combined with other factors, such as cost of sensors, number of errors during the mission etc., in a platform evaluation. Below, two platforms are described, one open-loop (without sensors) and one closed-loop (with sensors).

Open-loop robot platform

The mission cost (C_M) for this system is expressed as

$$C_M = \sum_{z \in \{N,M\}} \sum_i C\left(a_z^i\right) \tag{16}$$

As is seen, no sensing actions are executed, which corresponds to setting the information quality thresholds to zero. This system is appropriate only in a highly structured world.

Closed-loop robot platform

The mission cost in this case is expressed as

$$C_M = \sum_{z \in \{S,G,N,M\}} \sum_i C\left(a_z^i\right) \tag{17}$$

By repeating the solution of a task with different information quality threshold values, a compromise between safety and speed can be found. If a low information quality threshold value provides fairly high safety, this suggests that the corresponding perception is of little use and may be omitted.

References

1 R.A. Brooks, A Robust Layered Control System For A Mobile Robot, IEEE J. Robotics Automation, RA-2 (1) (1986) 14-23.

2 R.A. Brooks, Elephants Don´t Play Chess, Robotics Autonomous Systems 6 (1990) 3-15.

3 R. Chatila and J-P. Laumond, Position Referencing and Consistent World Modeling for Mobile Robots, Proc. IEEE Int. Conf. Robotics Automat., St. Louis, US (1985) 138-145.

4 B.R. Donald, J. Jennings, and R. Brown, Constructive recognizability for task-directed robot programming, Robotics Autonomous Systems 9 (1992) 41-74.

5 A. Ekman and D. Strömberg, Incremental map-making based on range images, M. Sc. Thesis, Defence Res. Establ., Sweden, 1992.

6 A. Ekman, D. Strömberg, and A. Lauberts, PSA - a Recursive Polyline Segmentation algorithm using an Eigenvector fit criterion, Technical Rep., Defence Res. Establ., Sweden 1993.

7 A. Ekman, D. Strömberg, and P. Klöör, Incremental map-making in indoor environments, Proc. IFAC workshop on Intelligent Autonomous Vehicles, Southampton, U K (1993).

8 M. Erdmann, Towards Task-Level Planning: Action-Based Sensor Design, Tech. Report, Carnegie Mellon University, Pittsburgh, PA, 1992.

9 O.D. Faugeras, Object representation, identification, and positioning from range data, Proc. 1st Int. Symp. Robotics Res., Cambridge, MA, (1984) 425-446.

10 B.R. Fox and K.G. Kempf, Planning, Scheduling, and Uncertainty in the Sequence of Future Events, in: J.F. Lemmer and L.N. Kanal, eds., Uncertainty in AI 2 (North-Holland, Amsterdam, 1988) 395-401.

11 I.J. Good, Dynamic Probability, Computer Chess, and the Measurement of Knowledge, in: E.W. Elcock and D. Michie eds., Machine Intelligence 8 (John Wiley & Sons, New York, 1977) 139-150.

12 I.J. Good, Rationality, Evidence, and Induction in Scientific Inference, in: E.W. Elcock and D. Michie eds., Machine Intelligence 8 (John Wiley & Sons, New York, 1977) 171-174.

13 E.T. Jaynes, Where do we stand on maximum entropy, in R.D. Levine and M. Tribus, eds., The maximum entropy formalism (MIT Press, 1979) 15-118.

14 J.N. Kapur and H.K. Kesavan, Entropy Optimization Principles with Applications, (Academic Press Inc., 1992).

15 S. Lee, Intelligent Sensing and Control for Advanced Teleoperation, IEEE Control Systems, 13 (3) (1993) 19-28.

16 J.S.B. Mitchell, D.W. Payton, and D.M. Keirsey, Planning and Reasoning for Autonomous Vehicle Control, Int. J. Intell. Systems 2 (2) (1987) 129-198.

17 J. Pearl, Fusion, Propagation, and Structuring in Belief Networks, Artificial Intelligence, 29 (1986) 241-288.

18 J.G. Pierce, A new look at the relation between information theory and search theory, in: R.D. Levine and M. Tribus, eds., The maximum entropy formalism (The MIT Press, Cambridge, Ma, 1978) 339-402.

19 A.C. Sanderson, Parts Entropy Methods for Robotic Assembly System Design, in: P.M. Jackson ed., 1983 annual report, Robotics Institute, Carnegie Mellon University (Pittsburgh, PA, 1983) 33-40.

20 G.N. Saridis, Toward the Realization of Intelligent Controls, Proc. IEEE, 67 (8) (1979).

21 G.N. Saridis, Entropy Formulation of Optimal and Adaptive Control, IEEE trans. Automatic Control, 33 (8) (1988) 713-721.

22 G.N. Saridis, Analytic Formulation of the Principle of Increasing Precision with Decreasing Intelligence for Intelligent Machines, Automatica 25 (3) (1989) 461-467.

23 R. Smith, M. Self, and P. Cheeseman, Estimating Uncertain Spatial Relationships in Robotics, in: J.F. Lemmer and L.N. Kanal eds., Uncertainty in Artificial Intelligence 2 (North-Holland, Amsterdam, 1988) 435-461.

24 K.P. Valavanis and G.N. Saridis, Intelligent Robotic Systems: Theory, Design and Applications, (Kluwer Academic Publishers, 1992).

25 L.A. Zadeh, Fuzzy sets as a basis for a theory of possibility, Fuzzy Sets and Systems 1 (1978) 3-28.