Computational Intelligence, Volume 17, Number 4, 2001

A COMPLEXITY-LEVEL ANALYSIS OF THE SENSOR PLANNING TASK FOR OBJECT SEARCH

YIMING YE

IBM T.J. Watson Research Center, Yorktown Heights, New York

JOHN K. TSOTSOS

Department of Computer Science, York University, Toronto, Ontario, Canada M3J 1P3

Object search is the task of searching for a given $3D$ object in a given $3D$ environment by a controllable camera. Sensor planning for object search refers to the task of how to select the sensing parameters of the camera so as to bring the target into the field of view of the camera and to make the image of the target to be easily recognized by the available recognition algorithms. In this paper, we study the task of sensor planning for object search from the theoretical point of view. We formulate the task and point out many of its important properties. We then analyze this task from the complexity level and prove that this task is NP-Complete.

Key words: computer vision, robotics, sensor planning, object search, complexity.

1. INTRODUCTION

The research described in this paper conforms to the complexity level analysis of the sensor planning task for object search. Object search is the task of searching for a given $3D$ object by a controllable camera in a given $3D$ environment. In general, this task consists of three subtasks. The first subtask is the selection of the sensing parameters so as to bring the target into the field of view of the sensor with sufficient image quality that it can be detected by the recognition algorithms. This is called the sensor planning problem for object search, and it is the main concern of this paper. The second subtask is the manipulation of the hardware so that the sensing operators can reach the state specified by the planner. The third subtask involves searching for the target within the image. This is the object recognition and localization problem, which attracts considerable attention within the computer vision community.

Sensor planning for object search is very important if a robot wants to interact intelligently and effectively with its environment. It has been examined in a variety of ways by the computer vision community. Connell (1989) constructs a robot that roams an area searching for and collecting soda cans. Garvey (1976) proposes the idea of indirect search for the target: first the sensor is directed to search for an "intermediate" object that commonly participates in a spatial relationship with the target, and then the sensor is directed to examine the restricted region specified by this relationship. Wixson and Ballard (1994) present a mathematical model of search efficiency and predict that indirect search can improve efficiency in many situations. Maver and Bajcsy (1990) develop a strategy that can determine the sequence of different views to check hidden regions.

There is no previous research within the computer vision community that attempts to formalize the sensor planning task for object search in general and to analyze this problem at the complexity level. This paper is an attempt along this direction. Complexity considerations are commonplace in the biological and computational vision

Address correspondence to Dr. Yiming Ye, IBM T.J. Watson Research Center, P.O. Box 704, Yorktown Heights, NY 10598, USA. e-mail: *yiming@watson.ibm.com.*

literature. For example, Tsotsos (1990) shows that the general problem of visual search (search for a target within an image) is computationally intractable in a formal, complexity-theoretic sense. Tsotsos (1992) also ties the concept of active perception to attentive processing in general and to his complexity level analysis of visual search and proves that active unbounded visual search is NP-Complete. Kirousis and Papadimitriou (1985) show that the problem of polyhedral scene labeling is inherently NP-Complete. Many other vision researchers (Grimson 1986; Plantinga and Dyer 1986, etc.) routinely provide an analysis of the complexity of their proposed algorithms. Complexity level analysis of robotics and vision problems is important because it can reveal basic insights into the structure of the problem and delimit the space of permissible solutions in a formal and theoretical fashion.

In this paper, we formulate the sensor planning task for object search, discuss several properties of this task, and prove that this task is *NP*-Complete. The theoretical result provided in this paper has been used as a guideline in designing the practical sensing strategies (see Ye 1997; Ye and Tsotsos 1995 for details).

2. PROBLEM FORMULATION

Although it is important to examine different aspects of object search individually and in some degree of isolation, it is even more important to study their relationship and integrate them into a whole search system.

The search region Ω is tessellated into a series of little elements c_i, $\Omega = \cup_{i=1}^{n} c_i$ and $c_i \cap c_j = 0$.

The search agent is a controllable camera. It can be a a mobile platform equipped with a camera that can pan, tilt and zoom or a robotic arm holding a zoom camera, etc. The state $\mathbf{s}$ of the searcher is uniquely determined by the position of the camera center, the viewing direction and the viewing angle size of the camera.

An operation $\mathbf{f} = \mathbf{f}(\mathbf{s}, a)$ is an action of the searcher within the region Ω, where a is a recognition algorithm. An operation $\mathbf{f}$ entails: adjust the camera configuration to the status specified by $\mathbf{s}$, take a *perspective* projection image, and search the image using recognition algorithm a. The cost for action $\mathbf{f}$ is represented by $C(\mathbf{f})$, which gives the total time to execute $\mathbf{f}$.

The knowledge of the search agent on the possible target position is encoded as a target probability distribution $\mathbf{p}(c_i, \tau)$, which gives the probability that the center of the target is within cell c_i at time τ. Usually this distribution is assumed to be known at the beginning of the search process, and it is determined by the agent's knowledge of the world. If the agent knows nothing about the target distribution, then a uniform distribution is assumed at the beginning. Note, we use $\mathbf{p}(c_0, \tau)$ to represent the probability that the target is outside the search region at time τ. The following constraint holds throughout the search process: $\sum_{i=0}^{n} \mathbf{p}(c_i, \tau) = 1$.

The detection function on Ω is a function $\mathbf{b}$, such that $\mathbf{b}(c_i, \mathbf{f})$ gives the conditional probability of detecting the target given that the center of the target is located within c_i and the operation is $\mathbf{f}$. For any operation, if the projection of the center of the cube c_i is outside the image, we assume $\mathbf{b}(c_i, \mathbf{f}) = 0$; if it is too far from the camera or too near to the camera, we also have $\mathbf{b}(c_i, \mathbf{f}) = 0$. In general (Ye 1997; Ye and Tsotsos 1995), $\mathbf{b}(c_i, \mathbf{f})$ is determined by various factors, such as intensity, occlusion, and orientation etc. It is obvious that the probability of detecting the target by applying action $\mathbf{f}$ is given by

$$P(\mathbf{f}) = \sum_{i=1}^{n} \mathbf{p}(c_i, \tau_{\mathbf{f}})\mathbf{b}(c_i, \mathbf{f}) \tag{1}$$

where $\tau_{\mathbf{f}}$ is the time just before $\mathbf{f}$ is applied.

The reason that the term $\tau_{\mathbf{f}}$ is introduced in the calculation of $P(\mathbf{f})$ is that the probability distribution needs to be updated whenever an action fails. Here Bayes' formula is used to incorporate the new recognition results into the old distribution. Let α_i be the event that the center of the target is in cell c_i, and α_0 be the event that the center of the target is outside the region, let β be the event that after applying a recognition action, the recognizer successfully detects the target. Then $P(\neg\beta \mid \alpha_i) = 1 - \mathbf{b}(c_i, \mathbf{f})$ and $P(\alpha_i \mid \neg\beta) = \mathbf{p}(c_i, \tau_{\mathbf{f}+})$, where $\tau_{\mathbf{f}+}$ is the time just after $\mathbf{f}$ is applied. Since the above events $\alpha_1, \ldots, \alpha_n, \alpha_o$ are mutually complementary and exclusive, we can get the following update rule:

$$\mathbf{p}(c_i, \tau_{\mathbf{f}+}) = \frac{\mathbf{p}(c_i, \tau_{\mathbf{f}})(1 - \mathbf{b}(c_i, \mathbf{f}))}{\mathbf{p}(c_0, \tau_{\mathbf{f}}) + \sum_{j=1}^{n} \mathbf{p}(c_j, \tau_{\mathbf{f}})(1 - \mathbf{b}(c_j, \mathbf{f}))}, \quad i = 1, \ldots, n, 0. \tag{2}$$

Since $\mathbf{p}(c_0, \tau_{\mathbf{f}}) + \sum_{j=1}^{n} \mathbf{p}(c_j, \tau_{\mathbf{f}}) = 1$ and $P(\mathbf{f}) = \sum_{j=1}^{n} \mathbf{p}(c_j, \tau_{\mathbf{f}})\mathbf{b}(c_j, \mathbf{f})$, the updating rule becomes

$$\mathbf{p}(c_i, \tau_{\mathbf{f}+}) = \frac{\mathbf{p}(c_i, \tau_{\mathbf{f}})(1 - \mathbf{b}(c_i, \mathbf{f}))}{1 - P(\mathbf{f})}, \quad i = 1, \ldots, n, 0. \tag{3}$$

Let $\mathbf{O}_\Omega$ be the set of all candidate operations that can be applied. The effort allocation $\mathbf{F} = \{\langle \mathbf{f}_1, \ldots, \mathbf{f}_k \rangle\}$ gives the ordered set of operations applied in the search, where $\mathbf{f}_i \in \mathbf{O}_\Omega$. $P(\mathbf{f}_1)$ gives the probability that the first action detects the target. $[1-P(\mathbf{f}_1)]P(\mathbf{f}_2)$ gives the probability that the first action does not detect the target, but the second action detects the target. Similar analysis can be applied to other actions in $\mathbf{F}$. Finally, $\{\prod_{i=1}^{k-1}[1-P(\mathbf{f}_i)]\}P(\mathbf{f}_k)$ gives the probability that $\mathbf{f}_1, \ldots, \mathbf{f}_{k-1}$ failed to detect the target, but $\mathbf{f}_k$ detects the target. It is clear that the probability of detecting the target by effort allocation $\mathbf{F}$ is given by:

$$P[\mathbf{F}] = P(\mathbf{f}_1) + [1 - P(\mathbf{f}_1)]P(\mathbf{f}_2) + \cdots + \left\{ \prod_{i=1}^{k-1} [1 - P(\mathbf{f}_i)] \right\} P(\mathbf{f}_k). \tag{4}$$

The total cost for applying this allocation is:

$$C[\mathbf{F}] = \sum_{i=1}^{k} C(\mathbf{f}_i). \tag{5}$$

Suppose K is the total time that can be allowed in the search, then the task of sensor planning for object search can be defined as finding an allocation $\mathbf{F} \subset \mathbf{O}_\Omega$, which satisfies $C[\mathbf{F}] \leq K$ and maximizes $P[\mathbf{F}]$.

3. SOME PROPERTIES OF THE OBJECT SEARCH PROCESS

It is interesting to explore the properties of the object search process defined above. Our purpose is to find regularities of this process so as to reveal some basic insights into the structure of the problem. First, we want to know the general expression for the probability distributions after we have executed an action sequence. Second, we want to know the general expression for the detection probability $P(\mathbf{f}_i)$ by applying an action $\mathbf{f}_i$ with respect to a given effort allocation $\mathbf{F}$. Third, we want to know the general expression for the expected probability of detecting the target by $\mathbf{F}$. We hope to express the above three quantities by using those quantities that are already known (the initial probability distribution and the detection probabilities of each action when no action has been applied before) as much as possible, and to avoid using the intermediate probability distributions. Although the updating rule looks complex, we can actually obtain these general expressions as stated below in Lemmas 1, 2, 3, and 4.

For any operation $\mathbf{f} \in \mathbf{O}_\Omega$, we define its influence range as $\Omega(\mathbf{f}) = \{c \mid \mathbf{b}(c, \mathbf{f}) \neq 0\}$. Intuitively, $\Omega(\mathbf{f})$ refers to the region that if the center of the target is within $\Omega(\mathbf{f})$, it might be detected by applying action $\mathbf{f}$. For a group of actions $\mathbf{f}_1, \ldots, \mathbf{f}_k$, we use $\Omega(\mathbf{f}_1 \cdots \mathbf{f}_k)$ to represent the intersection of their influence ranges, $\Omega(\mathbf{f}_1 \cdots \mathbf{f}_k) = \Omega(\mathbf{f}_1) \bigcap \cdots \bigcap \Omega(\mathbf{f}_k)$. Similarly, we use $\mathbf{b}(c, \mathbf{f}_1 \cdots \mathbf{f}_k)$ to represent the product of their detection function, $\mathbf{b}(c, \mathbf{f}_1 \cdots \mathbf{f}_k) = \mathbf{b}(c, \mathbf{f}_1) \times \cdots \times \mathbf{b}(c, \mathbf{f}_k)$. The complement of the influence range $\Omega(\mathbf{f})$ is: $\Omega(\bar{\mathbf{f}}) = \Omega - \{c \mid \mathbf{b}(c, \mathbf{f}) \neq 0\}$.

For any effort allocation $\mathbf{F} = \{\langle \mathbf{f}_1, \ldots, \mathbf{f}_q \rangle \mid \mathbf{f}_i \in \mathbf{O}_\Omega\}$, the initial probability distribution is denoted as $\mathbf{p}^{[0]}(c_1)$, $\mathbf{p}^{[0]}(c_2)$, $\ldots$, $\mathbf{p}^{[0]}(c_n)$, $\mathbf{p}^{[0]}(c_o)$. The distribution after the application of $\mathbf{f}_i$ is denoted as $\mathbf{p}^{[i]}(c_1)$, $\mathbf{p}^{[i]}(c_2)$, $\ldots$, $\mathbf{p}^{[i]}(c_n)$, $\mathbf{p}^{[i]}(c_o)$. Let $P(\mathbf{f}_i)$ represent the probability of detecting the target by applying the action $\mathbf{f}_i$ with respect to the allocation $\mathbf{F}$. Then of course we have $P(\mathbf{f}_i) = \sum_{j=1}^{n} \mathbf{p}^{[i-1]}(c_j)\mathbf{b}(c_j, \mathbf{f}_i)$. Let $P^{[0]}(\mathbf{f}_i)$ represents the probability of detecting the target by applying the action $\mathbf{f}_i$ when no action been applied before, then we have $P^{[0]}(\mathbf{f}_i) = \sum_{j=1}^{n} \mathbf{p}^{[0]}(c_j)\mathbf{b}(c_j, \mathbf{f}_i)$.

Lemma 1. Suppose $\langle \mathbf{f}_1, \ldots, \mathbf{f}_k \rangle$ are the ordered actions applied during the search process, then the resulting target probability distribution is given by:

$$\mathbf{p}^{[k]}(c) = \mathbf{p}^{[0]}(c) \frac{\prod_{i=1}^{k} [1 - \mathbf{b}(c, \mathbf{f}_i)]}{\prod_{i=1}^{k} [1 - P(\mathbf{f}_i)]}. \tag{6}$$

Lemma 1 can be easily proved by repeatedly using Term (2) and Term (3). It reveals many interesting properties. For example, if a cube c is outside the effective ranges of actions $\mathbf{f}_1, \ldots, \mathbf{f}_k$, then the probabilities associated with c increases after $\mathbf{f}_1, \ldots, \mathbf{f}_k$ are applied.

Lemma 2. Suppose two cubes c and c^* have the same probability value before an action $\mathbf{f}$ is applied, $\mathbf{p}(c, \tau_{\mathbf{f}}) = \mathbf{p}(c^*, \tau_{\mathbf{f}})$, where c belongs to the influence range of $\mathbf{f}$, $c \in \Omega(\mathbf{f})$, and c^* does not belong to the influence range of $\mathbf{f}$, $c^* \bar{\in} \Omega(\mathbf{f})$. Then after $\mathbf{f}$ is applied and fails, we have

$$\mathbf{p}(c, \tau_{\mathbf{f}+}) < \mathbf{p}(c^*, \tau_{\mathbf{f}+}). \tag{7}$$

Lemma 2 can be easily proved by repeatedly using Term (2).

Lemma 3. For allocation $\mathbf{F} = \{\langle \mathbf{f}_1 \cdots \mathbf{f}_q \rangle\}$, we have

$$
\begin{aligned}
P(\mathbf{f}_k) = \frac{1}{\prod_{i=1}^{k-1}[1-P(\mathbf{f}_i)]} \Bigg\{ & P^{[0]}(\mathbf{f}_k) + (-1)^1 \sum_{i_1=1}^{k-1} \left(\sum_{c \in \Omega(\mathbf{f}_{i_1}\mathbf{f}_k)} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_k) \right) \\
& + (-1)^2 \sum_{1 \le i_1 < i_2 \le k-1} \left(\sum_{c \in \Omega(\mathbf{f}_{i_1}\mathbf{f}_{i_2}\mathbf{f}_k)} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}\mathbf{f}_k) \right) \\
& + (-1)^3 \sum_{1 \le i_1 < i_2 < i_3 \le k-1} \left(\sum_{c \in \Omega(\mathbf{f}_{i_1}\mathbf{f}_{i_2}\mathbf{f}_{i_3}\mathbf{f}_k)} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}\mathbf{f}_{i_3}\mathbf{f}_k) \right) \\
& + \cdots + (-1)^{k-1} \sum_{c \in \Omega(\mathbf{f}_1 \cdots \mathbf{f}_{k-1}\mathbf{f}_k)} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_1\mathbf{f}_2 \cdots \mathbf{f}_{k-1}\mathbf{f}_k) \Bigg\}
\end{aligned}
\tag{8}
$$

for $k = 2, \ldots, q$.

Proof.

$$
\begin{aligned}
P(\mathbf{f}_k) &= \sum_{j=1}^{n} \mathbf{p}^{[k-1]}(c_j)\mathbf{b}(c_j, \mathbf{f}_k) = \sum_{c \in \Omega} \mathbf{p}^{[k-1]}(c)\mathbf{b}(c, \mathbf{f}_k) \\
&= \sum_{c \in \Omega} \mathbf{p}^{[0]}(c) \frac{\prod_{i=1}^{k-1}[1-\mathbf{b}(c, \mathbf{f}_i)]}{\prod_{i=1}^{k-1}[1-P(\mathbf{f}_i)]} \mathbf{b}(c, \mathbf{f}_k) \\
&= \frac{1}{\prod_{i=1}^{k-1}[1-P(\mathbf{f}_i)]} Q_{k-1}
\end{aligned}
\tag{9}
$$

where

$$
\begin{aligned}
Q_{k-1} = \sum_{c \in \Omega} \Bigg[\mathbf{p}^{[0]}(c) \Bigg(& 1 + (-1)^1 \sum_{i_1=1}^{k-1} \mathbf{b}(c, \mathbf{f}_{i_1}) + (-1)^2 \sum_{1 \le i_1 < i_2 \le k-1} \mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}) \\
& + (-1)^3 \sum_{1 \le i_1 < i_2 < i_3 \le k-1} \mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}\mathbf{f}_{i_3}) + \cdots \\
& + (-1)^{k-1}\mathbf{b}(c, \mathbf{f}_1 \cdots \mathbf{f}_{k-1}) \Bigg) \mathbf{b}(c, \mathbf{f}_k) \Bigg] \\
= \sum_{c \in \Omega} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_k) & + \sum_{c \in \Omega} \left[\mathbf{p}^{[0]}(c)(-1)^1 \sum_{i_1=1}^{k-1} \mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_k) \right] \\
+ \sum_{c \in \Omega} & \left[\mathbf{p}^{[0]}(c)(-1)^2 \sum_{1 \le i_1 < i_2 \le k-1} \mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}\mathbf{f}_k) \right] \\
+ \cdots & + \sum_{c \in \Omega} \mathbf{p}^{[0]}(c)(-1)^{k-1}\mathbf{b}(c, \mathbf{f}_1 \cdots \mathbf{f}_{k-1}\mathbf{f}_k)
\end{aligned}
$$

$$= P^{[0]}(\mathbf{f}_k) + (-1)^1 \sum_{i_1=1}^{k-1} \left[\sum_{c\in\Omega} \left(\mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_k) \right) \right]$$
$$+(-1)^2 \sum_{1\le i_1<i_2\le k-1} \left[\sum_{c\in\Omega} \left(\mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}\mathbf{f}_k) \right) \right]$$
$$+ \cdots + (-1)^{k-1} \sum_{c\in\Omega} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_1 \cdots \mathbf{f}_{k-1}\mathbf{f}_k) \tag{10}$$

Since:

$$\sum_{c\in\Omega} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}\cdots\mathbf{f}_{i_r}\mathbf{f}_k) = \sum_{c\in\Omega(\mathbf{f}_{i_1}\cdots\mathbf{f}_{i_r}\mathbf{f}_k)} \mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}\cdots\mathbf{f}_{i_r}\mathbf{f}_k)$$
$$+ \sum_{c\in\overline{\Omega(\mathbf{f}_{i_1}\cdots\mathbf{f}_{i_r}\mathbf{f}_k)}} \mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}\cdots\mathbf{f}_{i_r})\mathbf{b}(c, \mathbf{f}_k)$$
$$= \sum_{c\in\Omega(\mathbf{f}_{i_1}\cdots\mathbf{f}_{i_r}\mathbf{f}_k)} \mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}\cdots\mathbf{f}_{i_r}\mathbf{f}_k) + 0$$
$$= \sum_{c\in\Omega(\mathbf{f}_{i_1}\cdots\mathbf{f}_{i_r}\mathbf{f}_k)} \mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}\cdots\mathbf{f}_{i_r}\mathbf{f}_k) \tag{11}$$

From (9), (10), and (11), it is easy to know that the lemma is true. □

Lemma 4. For the given allocation $\mathbf{F} = \{\langle \mathbf{f}_1, \ldots, \mathbf{f}_q \rangle\}$, we have

$$P[\mathbf{F}] = \sum_{i_1=1}^{q} P^{[0]}(\mathbf{f}_i) + (-1)^{2+1} \sum_{1\le i_1<i_2\le q} \left(\sum_{c\in\Omega(\mathbf{f}_{i_1}\mathbf{f}_{i_2})} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}) \right)$$
$$+ \cdots + (-1)^{r+1} \sum_{1\le i_1<i_2<\cdots<i_r\le q} \left(\sum_{c\in\Omega(\mathbf{f}_{i_1}\mathbf{f}_{i_2}\cdots\mathbf{f}_{i_r})} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_{i_1}\mathbf{f}_{i_2}\cdots\mathbf{f}_{i_r}) \right)$$
$$+ \cdots + (-1)^{q+1} \left(\sum_{c\in\Omega(\mathbf{f}_1\mathbf{f}_2\cdots\mathbf{f}_q)} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_1\cdots\mathbf{f}_q) \right) \tag{12}$$

Lemma 4 can be proved by using Lemma 3 and mathematical induction. One important inference from this lemma is that the value of $P[\mathbf{F}]$ is not influenced by the order of the applied actions. This agrees with common-sense expectation.

Lemma 5. For an allocation $\mathbf{F} = \{\langle \mathbf{f}_1, \ldots, \mathbf{f}_q \rangle\}$, if we restrict the available actions such that there is no cube belongs to any two action ranges: $\Omega(\mathbf{f}_j) \bigcap \Omega(\mathbf{f}_i) = \phi$, if $i \ne j$, then $P[\mathbf{F}]$ can be calculated by the following:

$$P[\mathbf{F}] = \sum_{i=1}^{q} P^{[0]}(\mathbf{f}_i) \tag{13}$$

4. THE COMPLEXITY OF THE OBJECT SEARCH TASK

In this section, we prove that the sensor planning task for object search is *NP-complete*. First, the maximization problem is changed into the equivalent decision problem.

INSTANCE: A search region $\Omega = \cup_{i=1}^{n} c_i$, a finite candidate action set $\mathbf{O}_\Omega = \{\mathbf{f}_1^*, \mathbf{f}_2^*, \ldots, \mathbf{f}_m^*\}$, an expected probability function $P[\mathbf{F}]$ as described in (4) and a cost function $C[\mathbf{F}]$ as described in (5). Where $\mathbf{F} = \{\langle \mathbf{f}_1, \ldots, \mathbf{f}_k \rangle \mid \mathbf{f}_i \in \mathbf{O}_\Omega\}$ is the ordered set of operations applied in the search. An environmental updating rule as described in (2), a cost constraint K and an expected probability goal M.

QUESTION: Is there an effort allocation $\mathbf{F}$ such that $T[\mathbf{F}] \le K$ and $P[\mathbf{F}] \ge M$

Now, we describe the problem in a more theoretical fashion.

We use a one dimensional array A to represent the probability distributions: $A[i] = \mathbf{p}(c_i, \tau)$. Each $A[i] \ge 0$ is assumed to be a rational number $A[i] = \frac{p_i}{q_i} > 0$, where $p_i \in Z^+$ and $q_i \in Z^+$.

We use a $2D$ array B to represent the detection function, $B[i][j] = \mathbf{b}(c_i, \mathbf{f}_j^*)$. Each $B[i][j]$ ($0 \le i \le n$ and $0 \le j \le m$) is assumed to be a rational number between 0 and 1, $B[i][j] = \frac{p_{ij}}{q_{ij}}$, where $p_{ij} \in Z^+$, $q_{ij} \in Z^+$.

The probability updating rule can be represented as

$$A[i] \leftarrow \frac{A[i](1 - B[i,j])}{A[0] + \sum_{k=1}^{n} A[k](1 - B[k,j])}, \quad i = 0, 1, \ldots, n. \tag{14}$$

The probability of detecting the target by an action $P(\mathbf{f}_j^*)$ is represented as $P[\langle j \rangle]$, thus $P[\langle j \rangle] = \sum_{i=1}^{n} B[i,j]A[i]$. Here $A[i]$ is the most recent updated value of $A[i]$ just before $\mathbf{f}_j^*$ is applied.

The effort allocation $\mathbf{F} = \{\langle \mathbf{f}_{j_1}^*, \ldots, \mathbf{f}_{j_k}^* \rangle\}$ is represented as $\mathbf{F} = \langle j_1, \ldots, j_k \rangle$, and the probability of detecting the target by this allocation is given by

$$\begin{aligned} P[\langle j_1, \ldots, j_k \rangle] &= P[\langle j_1 \rangle] + [1 - P[\langle j_1 \rangle]]P[\langle j_2 \rangle] \\ &\quad + \cdots + \left\{ \prod_{i=1}^{k-1} [1 - P[\langle j_i \rangle]] \right\} P[\langle j_k \rangle]. \end{aligned} \tag{15}$$

The cost of $\mathbf{f}_i^*$ is represented as $C[i]$ ($C[i] = C(\mathbf{f}_i^*)$). So, the cost for $\langle j_1, \ldots, j_k \rangle$ is represented by $T[\langle j_1, \ldots, j_k \rangle] = \sum_{i=1}^{k} C[j_i]$.

With the above description, the task of sensor planning for object search can be described as

INSTANCE:

- an index set $\{1, \ldots, m\}$;
- a probability set $\{A[i] \mid 0 \le i \le n\}$, $A[i]$ ($0 \le i \le n$) are a rational numbers, $A[i] = \frac{p_i}{q_i}$, $p_i \in Z^+, q_i \in Z^+, 0 \le A[i] \le 1$, $A[i]$ satisfy $\sum_{i=0}^{n} A[i] = 1$;
- a detection function set $\{B[i][j] \mid 0 \le i \le n, 0 \le j \le m\}$, $B[i][j]$ ($0 \le i \le n, 0 \le j \le m$) are rational numbers, $B[i][j] = \frac{p_{ij}}{q_{ij}}$, $p_{ij} \in Z^+, q_{ij} \in Z^+, 0 \le B[i][j] \le 1$;
- a cost set $\{C[i] \mid 1 \le i \le m\}$, $C[i]$ ($1 \le i \le m$) are rational numbers, $C[i] = \frac{C_{pi}}{C_{qi}}$, $C_{pi} \in Z^+, C_{qi} \in Z^+$;

- a cost limit constant K (K is a rational number, $K = \frac{K_p}{K_q}$, $K_p, K_q \in Z^+$);
- a goal M (M is a rational number, $M = \frac{M_p}{M_q}$, $M_p, M_q \in Z^+$).

QUESTION:

Is there an ordered set $\langle j_1, \ldots, j_k \rangle$ ($j_i \in \{1, \ldots, m\}$ for $1 \le i \le k$), such that $T[\langle j_1, \ldots, j_k \rangle] \le K$ and $P[\langle j_1, \ldots, j_k \rangle] \ge M$?

4.1. The Sensor Planning for Object Search Task Is an NP-Problem

The following nondeterministic algorithm solves this problem in polynomial time:

```
procedure NondeterministicObjectSearch(A,B,C,K, M)
    rational A[0:n], B[0:n][1:m], C[1:m], K,M
    integer E[0:m], F[0:m], len
    for i ← 1 to m do
        E[i] ← choice(0, 1, 2, ..., m)
    end for
    for i ← 1 to m do
        F[E[i]] = i
    end for
    len = 0
    i = 1
    while F[i] ≠ 0
        len ← len + 1
        i++
    end while
    for i = 1, i ≤ len, i++
        P[i] = 0
        for j = 1, j ≤ n, j++
            P[i] = P[i] + B[j, i] * A[j]
        end for
        for j = 1, j ≤ n, j++
            Temp_a = A[j] * (1 − B[j, i])
            Temp_b = 0
            for k = 0, k ≤ n, k++
                Temp_b = Temp_b + A[k](1 − B[k, i])
            end for
            A[j] = Temp_a / Temp_b
        end for
    end for
    P_F = 0
    Term_a = 1; Term_b = 0
    F[0] = 0; P[0] = 0
    for i = 1, i ≤ len, i++
        Term_a = Term_a * (1 − P[F[i − 1]])
        Term_b = P[F[i]]
        P_F = P_F + Term_a * Term_b
    end for
```

$T_F = 0$
for $i = 1,\ i \le len,\ i{+}{+}$
    $T_F = T_F + C[F[i]]$
end for
if $T_F > K$ or $P_F < M$ **then**
    failure
else
    success
endif
end NondeterministicObjectSearch

4.2. The Sensor Planning for Object Search Task Is NP-Complete

In this section, we use proof by restriction to prove that object search task belongs to NP-complete class. An NP-completeness proof by restriction for a given problem $\Pi \in NP$ consists simply of showing that Π contains a known NP-complete problem Π' as a special case. The heart of such a proof lies in the specification of the additional restrictions to be placed on the instances of Π so that the resulting restricted problem will be identical to Π'.

Theorem 1. The sensor planning task for object search is NP-complete.

We prove the theorem by restriction.

The *KNAPSACK* problem is a well known NP-Complete problem [?]. We restrict the object search task to *KNAPSACK* by allowing instances in which any two operations in $\mathbf{O}_\Omega = \{\mathbf{f}_1^*, \mathbf{f}_2^*, \ldots, \mathbf{f}_m^*\}$ have no common influence region. This means that $\Omega(\mathbf{f}_i) \bigcap \Omega(\mathbf{f}_j) = \phi$ for any $i \neq j$.

After this restriction, by combining Lemma 4 and Lemma 13, the task becomes

INSTANCE:

- A finite action set $U = \{1, \ldots, m\}$
- An initial detection probability $P^{[0]}[u]$ for each $u \in U$
- A cost $C[u]$ for each $u \in U$
- a cost limit constant K (K is a rational number);
- a goal M (M is a rational number).

QUESTION: Is there a subset $\mathbf{F} \subseteq U$ such that

$$\sum_{u \in \mathbf{F}} P^{[0]}[u] \le K$$

and

$$\sum_{u \in \mathbf{F}} C[u] \ge M?$$

Obviously the above problem is equivalent to the *KNAPSACK* problem. Thus, the task of sensor planning for object search is NP-complete. □

Theorem 1 means that no provably optimal algorithm in polynomial time exists, and thus a good approximation algorithm is probably all one can get.

5. THE SIMPLIFIED VERSION: A ONE-STEP LOOK AHEAD PROBLEM

Since the sensor planning problem is NP-complete and usually the number of the available candidate actions are big, it is necessary to relax the original problem. Instead of looking for an algorithm that always generate an optimal solution, we will simply use heuristics that will generate a feasible solution for the original problem. Since the calculation of the detection probability for a given operation is time consuming, it is not practical to use complex strategies. Here we use greedy strategy. The greedy method suggests that one can device an algorithm which works in stages, considering one input at a time. At each stage, based on some optimization measure, the next candidate is selected and is included into the partial solution so far. Suppose we have already executed k $(k \geq 1)$ actions $\mathbf{F}_q = \{\mathbf{f}_1, \ldots, \mathbf{f}_q\}$. We now want to find the next action $\mathbf{f}_{q+1}$ to execute, with the hope that our strategy of finding the next action may lead to an *approximate* solution for the object search task.

When we execute a next action $\mathbf{f}$, the effort allocation becomes $\mathbf{F}_{q+1}=\{\mathbf{f}_1, \ldots, \mathbf{f}_q, \mathbf{f}\}$. The expected probability of detecting the target is $P[\mathbf{F}_{q+1}] = P[\mathbf{F}_q] + \Delta_P(\mathbf{f})$, where

$$\Delta_P(\mathbf{f}) = \left\{ \prod_{j=1}^{q} \left[1 - \sum_{i=1}^{n} \mathbf{p}(c_i, \tau_{\mathbf{f}_j}) \mathbf{b}(c_i, \mathbf{f}_j) \right] \right\} \times \left[\sum_{i=1}^{n} \mathbf{p}(c_i, \tau_{\mathbf{f}}) \mathbf{b}(c_i, \mathbf{f}) \right]. \tag{16}$$

The total cost becomes $C[\mathbf{F}_{q+1}] = C[\mathbf{F}_q] + \Delta_C(\mathbf{f})$, where $\Delta_C(\mathbf{f}) = \mathbf{C}(\mathbf{f})$. Our strategy for selecting the next action is: the next action $\mathbf{f}_{q+1}$ should be selected such that the term $\frac{\Delta_P(\mathbf{f})}{\Delta_C(\mathbf{f})}$ is maximized. Since $\{\prod_{j=1}^{q}[1 - \sum_{i=1}^{n} \mathbf{p}(c_i, \tau_{\mathbf{f}_j}) \mathbf{b}(c_i, \mathbf{f}_j)]\}$ is fixed no matter what the next action $\mathbf{f}$ is selected, our strategy becomes: the next action $\mathbf{f}_{q+1}$ should be selected that maximizes the term

$$\mathbf{E}(\mathbf{f}) = \frac{\sum_{i=1}^{n} \mathbf{p}(c_i, \tau_{\mathbf{f}}) \mathbf{b}(c_i, \mathbf{f})}{C(\mathbf{f})}. \tag{17}$$

In some situations, the simple greedy strategy may even generate an optimal answer as stated in Theorem 2.

Theorem 2. Suppose the available operations $\mathbf{O}_\Omega = \{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_m\}$ satisfy:

(1) $C(\mathbf{f}_i) = C(\mathbf{f}_j)$, $1 \leq i, j \leq m$.
(2) $\Omega(\mathbf{f}_j) \bigcap \Omega(\mathbf{f}_i) = \phi$, $1 \leq i, j \leq m, i \neq j$.

Then, the above greedy strategy generates an optimal answer.

Proof. Because $C(\mathbf{f}_i) = C(\mathbf{f}_j)$ (for $1 \leq i, j \leq m$), it is obvious that the number of actions selected by the best allocation should equal to that selected by our greedy algorithm. Suppose the optimal allocation is $\mathbf{F}^* = \langle \mathbf{f}_1^*, \ldots, \mathbf{f}_k^* \rangle$. The actions selected by our algorithm is $\mathbf{F} = \langle \mathbf{f}_1, \ldots, \mathbf{f}_k \rangle$.

If $\mathbf{F}$ is the same as $\mathbf{F}$, then the theorem is true.

Otherwise, suppose $\mathbf{F}$ is different from $\mathbf{F}^*$. Then, there must exist a value r (where $1 \leq r \leq k$) such that $\mathbf{f}_r \neq \mathbf{f}_r^*$, and when $r > 1$, $\mathbf{f}_1 = \mathbf{f}_1^*, \ldots, \mathbf{f}_{r-1} = \mathbf{f}_{r-1}^*$.

For action $\mathbf{f}_r$, there are two situations: $\mathbf{f}_r \in \{\mathbf{f}_{r+1}^*, \ldots, \mathbf{f}_k^*\}$, or $\mathbf{f}_r \notin \{\mathbf{f}_{r+1}^*, \ldots, \mathbf{f}_k^*\}$.

If $\mathbf{f}_r \in \{\mathbf{f}_{r+1}^*, \ldots, \mathbf{f}_k^*\}$, then switch the position of $\mathbf{f}_r$ and $\mathbf{f}_r^*$ in the effort allocation $\mathbf{F}^*$, resulting in an effort allocation $\mathbf{F}^{*'}$. From Lemma 13, we have $P[\mathbf{F}^{*'}] = P[\mathbf{F}^*]$. Thus $\mathbf{F}^{*'}$ is also an optimal effort allocation and the first r actions of $\mathbf{F}^{*'}$ are equal to the first r actions of $\mathbf{F}$.

If $\mathbf{f}_r \notin \{\mathbf{f}_{r+1}^*, \ldots, \mathbf{f}_k^*\}$, then replace $\mathbf{f}_r^*$ with $\mathbf{f}_r$, resulting in an effort allocation $F^{*''}$.

It is easy to know that

$$P(\mathbf{F}^{*''}) - P(\mathbf{F}^{*}) = \sum_{i=1}^{r-1} P^{[0]}(\mathbf{f}_i^*) + P^{[0]}(\mathbf{f}_r) + \sum_{i=r+1}^{k} P^{[0]}(\mathbf{f}_i^*) - \sum_{i=1}^{k} P^{[0]}(\mathbf{f}_i^*)$$
$$= P^{[0]}(\mathbf{f}_r) - P^{[0]}(\mathbf{f}_r^*)$$

Because the first $r-1$ actions are the same for $\mathbf{F}^*$ and $\mathbf{F}$, the probability distribution $\mathbf{p}$ are the same just before the application of action $\mathbf{f}_r^*$ and that of action $\mathbf{f}_r$, from the greedy strategy, we know

$$\frac{\sum_{c\in\Omega(\mathbf{f}_r)} \mathbf{p}(c)\mathbf{b}(c, \mathbf{f}_r)}{C(\mathbf{f}_r)} \geq \frac{\sum_{c\in\Omega(\mathbf{f}_r^*)} \mathbf{p}(c)\mathbf{b}(c, \mathbf{f}_r^*)}{C(\mathbf{f}_r)} \tag{18}$$

From Lemma 1 and the two conditions of this lemma, we have:

$$\sum_{c\in\mathbf{f}} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_r^*) \geq \sum_{c\in\mathbf{f}^*} \mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f}_r^*) \tag{19}$$

This is actually

$$P^{[0]}(\mathbf{f}_r^*) \geq P^{[0]}(\mathbf{f}_r) \tag{20}$$

Thus, $P(\mathbf{F}^{*''}) \geq P(\mathbf{F}^*)$. Since $\mathbf{F}^*$ is an optimal allocation, $\mathbf{F}^{*''}$ must also be an optimal allocation.

Thus, for any situations of $\mathbf{f}_r$, we are able to find an optimal effort allocation such that the first r actions of $\mathbf{F}$ are the same as the first r actions of the new optimal effort allocation. If this new optimal effort allocation is the same as $\mathbf{F}$, then the lemma is proved. Otherwise, we can repeat the above process until all the actions of $\mathbf{F}$ are equal to an optimal effort allocation. And finally the theorem can be proved. □

In our approach, the task of sensor planning for object search is formulated as a task to maximize the probability of detecting the target with a given cost. We can also prove from Lemma 4 that for a given effort allocation, the probability of detecting the target is not influenced by the order of the applied actions. But this result does not suggest that we can apply a given set of actions in any random order. Although the criterion of detection probability is very important, it is not the only criteria that a search agent should take into the consideration. Another very important criterion is the time used to detect the target. It is obvious that the search agent wants to find the target as early as possible. For this reason, we introduce the concept of "expected time to detect the target" for a given effort allocation $\mathbf{F} = \{\mathbf{f}_1, \ldots, \mathbf{f}_k\}$.

Definition 1. For a given effort allocation $\mathbf{F} = \{\mathbf{f}_1, \ldots, \mathbf{f}_k\}$, the expected time to detect the target by applying $\mathbf{F}$ is defined as

$$\Theta[\mathbf{F}] = C(\mathbf{f}_1) \times P(\mathbf{f}_1) + \big(C(\mathbf{f}_1) + C(\mathbf{f}_2)\big) \times [1 - P(\mathbf{f}_1)]P(\mathbf{f}_2)$$
$$+ \cdots + \big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}_k)\big) \times \left\{\prod_{i=1}^{k-1}[1 - P(\mathbf{f}_i)]\right\}P(\mathbf{f}_k) \tag{21}$$

By using Lemma 3, it is easy to get the following lemma.

Lemma 6. For an allocation $\mathbf{F} = \{\mathbf{f_1}, \ldots, \mathbf{f_q}\}$, if we restrict the available actions such that there is no cube belonging to any two influence ranges: $\Omega(\mathbf{f}_j) \bigcap \Omega(\mathbf{f}_i) = \phi$, if $i \neq j$, then $\Theta[\mathbf{F}]$ can be calculated by the following:

$$\Theta[\mathbf{F}] = C(\mathbf{f}_1) \times P^{[0]}(\mathbf{f}_1) + \big(C(\mathbf{f}_1) + C(\mathbf{f}_2)\big) \times P^{[0]}(\mathbf{f}_2) + \cdots + \big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}_k)\big) \times P^{[0]}(\mathbf{f}_k) \quad (22)$$

Before pointing out some advantages of our approach, we first give the following property of our strategy.

Lemma 7. For a group of actions $\{\mathbf{f}_1, \ldots, \mathbf{f}_q\}$, suppose that

(1) There is no cube belonging to any two action ranges: $\Omega(\mathbf{f}_j) \bigcap \Omega(\mathbf{f}_i) = \phi$, if $i \neq j$;
(2) $\frac{P^{[0]}(\mathbf{f}_1)}{C(\mathbf{f}_1)} > \frac{P^{[0]}(\mathbf{f}_2)}{C(\mathbf{f}_2)} > \cdots > \frac{P^{[0]}(\mathbf{f}_k)}{C(\mathbf{f}_k)}$.

Then the order of applying these actions by our one step look ahead strategy must be: $\mathbf{f}_1 \rightarrow \mathbf{f}_2 \rightarrow \cdots \rightarrow \mathbf{f}_k$.

Proof. We prove it by mathematical induction on the number of actions selected.

(A) When $i = 1$, from condition (1) and object function used by the greedy algorithm, it is easy to show that $\mathbf{f}_1$ is selected.
(B) Suppose when $i = r$, the selected actions are $\mathbf{f}_1 \rightarrow \mathbf{f}_2 \rightarrow \cdots \rightarrow \mathbf{f}_r$.
(C) When $i = r + 1$, we need to select the next action. For any action $\mathbf{f}$ that belongs to $\{\mathbf{f}_{r+1}, \mathbf{f}_{r+2}, \ldots, \mathbf{f}_k\}$, we have:

$$P(\mathbf{f}) = \sum_{j=1}^{n} \mathbf{p}^{[r]}(c_j)\mathbf{b}(c_j, \mathbf{f}) = \sum_{c \in \Omega(\mathbf{f})} \mathbf{p}^{[r]}(c)\mathbf{b}(c, \mathbf{f}).$$

Since $c \in \Omega(\mathbf{f})$, thus $c \overline{\in} \Omega(\mathbf{f}_1), \ldots, c \overline{\in} \Omega(\mathbf{f}_r)$, so

$$\begin{aligned} P(\mathbf{f}) &= \sum_{c \in \Omega(\mathbf{f})} \frac{\mathbf{p}^{[0]}(c)(1 - \mathbf{b}(c, \mathbf{f}_1))(1 - \mathbf{b}(c, \mathbf{f}_2)) \cdots (1 - \mathbf{b}(c, \mathbf{f}_r))}{(1 - P(\mathbf{f}_1))(1 - P(\mathbf{f}_2)) \cdots (1 - P(\mathbf{f}_r))} \mathbf{b}(c, \mathbf{f}) \\ &= \sum_{c \in \Omega(\mathbf{f})} \frac{\mathbf{p}^{[0]}(c)}{(1 - P(\mathbf{f}_1))(1 - P(\mathbf{f}_2)) \cdots (1 - P(\mathbf{f}_r))} \mathbf{b}(c, \mathbf{f}) \\ &= \frac{1}{(1 - P(\mathbf{f}_1))(1 - P(\mathbf{f}_2)) \cdots (1 - P(\mathbf{f}_r))} \sum_{c \in \Omega(\mathbf{f})} \big[\mathbf{p}^{[0]}(c)\mathbf{b}(c, \mathbf{f})\big] \\ &= \frac{1}{(1 - P(\mathbf{f}_1))(1 - P(\mathbf{f}_2)) \cdots (1 - P(\mathbf{f}_r))} P^{[0]}(\mathbf{f}). \end{aligned}$$

From condition (2), we know:

$$\frac{P(\mathbf{f}_{r+1})}{C(\mathbf{f}_{r+1})} > \frac{P(\mathbf{f}_{r+2})}{C(\mathbf{f}_{r+2})} > \cdots > \frac{P(\mathbf{f}_k)}{C(\mathbf{f}_k)}. \quad (23)$$

Thus, the next action selected by the greedy algorithm is $\mathbf{f}_{r+1}$.
From (A), (B), and (C), we conclude that the property is true.

Note: when equality occurs in condition (2), then the two actions with the equality relation can be switched.

Theorem 3. For a group of actions $\{\mathbf{f}_1, \ldots, \mathbf{f}_q\}$, if there is no cube belonging to any two action ranges, i.e., $\Omega(\mathbf{f}_j) \bigcap \Omega(\mathbf{f}_i) = \phi$, for $i \neq j$, then the application order of the actions selected by our strategy minimize the expected time of detecting the target with respect to this group.

Proof. Without loss of generality, assume:

$$\frac{P^{[0]}(\mathbf{f}_1)}{C(\mathbf{f}_1)} \geq \frac{P^{[0]}(\mathbf{f}_2)}{C(\mathbf{f}_2)} \geq \cdots \geq \frac{P^{[0]}(\mathbf{f}_q)}{C(\mathbf{f}_q)}. \tag{24}$$

Then, from Lemma 7, the order of applied actions will be: $\mathbf{f}_1 \to \mathbf{f}_2 \to \cdots \to \mathbf{f}_q$. (Note: When equality occurs, the applying order might be different, but only those with equality can interchange, and this change will not influence the expected time of detecting the target. Therefore, we assume that the above is the application order selected by our algorithm.)

Let $\mathbf{F} = \{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_q\}$. Suppose $\mathbf{F}' = \{\mathbf{f}'_1, \mathbf{f}'_2, \ldots, \mathbf{f}'_q\}$ is an effort allocation that generates the minimum expected time of detecting the target, we will prove in the following that $\Theta[\mathbf{F}] \leq \Theta[\mathbf{F}']$.

If $\mathbf{F}'$ is not the same as $\mathbf{F}$, then there must exist an integer r such that $\mathbf{f}_1 = \mathbf{f}'_1$, $\mathbf{f}_2 = \mathbf{f}'_2, \ldots, \mathbf{f}_{r-1} = \mathbf{f}'_{r-1}$, but $\mathbf{f}_r \neq \mathbf{f}'_r$.

Thus, the index of $\mathbf{f}_r$ in $\mathbf{F}'$ is bigger than r. From the greedy algorithm,

$$\frac{P^{[0]}(\mathbf{f}_r)}{C(\mathbf{f}_r)} \geq \frac{P^{[0]}(\mathbf{f}'_{r+i})}{C(\mathbf{f}'_{r+i})}, \tag{25}$$

where $i \geq 0$.

Suppose the index of $\mathbf{f}_r$ in $\mathbf{F}'$ is $r + k$. Then,

$$\mathbf{F}' = \left\{\mathbf{f}_1, \ldots, \mathbf{f}_{r-1}, \mathbf{f}'_r, \ldots, \mathbf{f}'_{r+k-1}, \mathbf{f}_r, \ldots, \mathbf{f}'_q\right\}. \tag{26}$$

Now let's consider another effort allocation $\mathbf{F}''$,

$$\mathbf{F}'' = \left\{\mathbf{f}_1, \ldots, \mathbf{f}_{r-1}, \mathbf{f}'_r, \ldots, \mathbf{f}_r, \mathbf{f}'_{r+k-1}, \ldots, \mathbf{f}'_q\right\}. \tag{27}$$

We have:

$$\begin{aligned}\Theta[\mathbf{F}'] = {} & C(\mathbf{f}_1) \times P^{[0]}(\mathbf{f}_1) + \big(C(\mathbf{f}_1) + C(\mathbf{f}_2)\big) \times P^{[0]}(\mathbf{f}_2) \\ & + \cdots + \big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}'_{r+k-1})\big) \times P^{[0]}(\mathbf{f}'_{r+k-1}) \\ & + \big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}'_{r+k-1}) + C(\mathbf{f}_r)\big) \times P^{[0]}(\mathbf{f}_r) \\ & + \cdots + \big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}'_q)\big) \times P^{[0]}(\mathbf{f}'_q)\end{aligned} \tag{28}$$

$$\begin{aligned}\Theta[\mathbf{F}''] = {} & C(\mathbf{f}_1) \times P^{[0]}(\mathbf{f}_1) + \Big(C(\mathbf{f}_1) + C(\mathbf{f}_2)\Big) \times P^{[0]}(\mathbf{f}_2) \\ & + \cdots + \Big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}_r)\Big) \times P^{[0]}(\mathbf{f}_r) \\ & + \big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}_r) + C(\mathbf{f}'_{r+k-1})\big) \times P^{[0]}(\mathbf{f}'_{r+k-1}) \\ & + \cdots + \Big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}'_q)\Big) \times P^{[0]}(\mathbf{f}'_q). \end{aligned} \tag{29}$$

Thus,

$$\begin{aligned}\Theta[\mathbf{F}''] - \Theta[\mathbf{F}'] = {} & \Big\{\big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}_r)\big) \times P^{[0]}(\mathbf{f}_r) \\ & \quad + \big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}_r) + C(\mathbf{f}'_{r+k-1})\big) \times P^{[0]}(\mathbf{f}'_{r+k-1})\Big\} \\ & - \Big\{\big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}'_{r+k-1})\big) \times P^{[0]}(\mathbf{f}'_{r+k-1}) \\ & \quad + \big(C(\mathbf{f}_1) + \cdots + C(\mathbf{f}'_{r+k-1}) + C(\mathbf{f}_r)\big) \times P^{[0]}(\mathbf{f}_r)\Big\} \\ = {} & C(\mathbf{f}_r)P^{[0]}(\mathbf{f}'_{r+k-1}) - C(\mathbf{f}'_{r+k-1})P^{[0]}(\mathbf{f}_r) \\ = {} & C(\mathbf{f}_r)C(\mathbf{f}'_{r+k-1})\left\{\frac{P^{[0]}(\mathbf{f}'_{r+k-1})}{C(\mathbf{f}'_{r+k-1})} - \frac{P^{[0]}(\mathbf{f}_r)}{C(\mathbf{f}_r)}\right\} \\ \leq {} & 0. \end{aligned}$$

Thus,

$$\begin{aligned}& \Theta\big[\{\mathbf{f}_1, \ldots, \mathbf{f}_{r-1}, \mathbf{f}'_r, \ldots, \mathbf{f}_r, \mathbf{f}'_{r+k-1}, \ldots, \mathbf{f}_q\}\big] \\ & \quad \leq \Theta\big[\{\mathbf{f}_1, \ldots, \mathbf{f}_{r-1}, \mathbf{f}'_r, \ldots, \mathbf{f}'_{r+k-1}, \mathbf{f}_r, \ldots, \mathbf{f}'_q\}\big]. \end{aligned}$$

Continue the above switching process. We can finally move the action $\mathbf{f}_r$ forward until the index of $\mathbf{f}_r$ is r in $\mathbf{F}'$, while at the same time, the expected detection time will not increase.

So, there exists another effort allocation whose first r actions are same as **F** and whose expected time of detecting the target is also minimum. Continue this process, we can finally prove that **F** also minimizes the expected time of detecting the target. □

6. DISCUSSION

We believe that it is very important to study the sensor planning problem of object search from a theoretical point of view. Complexity level analysis of this problem can reveal basic insights into its structure and delimit the space of permissible solutions in a formal and theoretical fashion. Only by this way can we get a better understanding of the problem and design an efficient and powerful algorithm in practice.

This paper formalizes the sensor planning problem, points out several properties of this task and analyzes the complexity of this task. The result presented in this paper has been used as a guideline in designing the practical sensor planning system (see Ye 1997; Ye and Tsotsos 1995 for details). According to the properties of the image formation process, we decompose the huge space of all possible sensing actions into a small number of actions that must be tried. These small number of actions have the property that, in most situations, the intersection of the influence ranges of any two of them is an empty set. The greedy strategy described in section 5 is used in the action selection process. Many of the properties discussed in this paper are considered in the real implementation. Our sensor planning system has been extensively tested by simulation experiments and real experiments. The experimental results are satisfactory.

ACKNOWLEDGMENT

This work is supported by IBM Centre for Advanced Studies and ARK (Autonomous Robot for a Known environment) Project, which receives its funding from PRECARN Associates Inc., Industry Canada, the National Research Council of Canada, Technology Ontario, Ontario Hydro Technologies, and Atomic Energy of Canada Limited. The authors would like to thank Professor Hector Levesque for his help and comments on the NP-complete proof of this paper. Yiming Ye would like to thank Chris Brown, Eugene Fiume, Andrew Goldenberg, Allan Jepson, Hector Levesque, and Demetri Terzopoulos for their valuable comments on his Ph.D thesis.

REFERENCES

BAJCSY, R. 1985. Active perception vs. passive perception. *In* Third IEEE Workshop on Vision, 55–59.

CONNELL, J. 1989. An Artificial Creature. Ph.D. Dissertation, AI Lab, MIT.

GARVEY, T. D. 1976. Perceptual strategies for purposive vision. Technical Report Technical Note 117, SRI International.

GRIMSON, W. 1986. The combinatorics of local constraints in model-based recognition and localization from sparse data. Journal of the Association for Computing Machinery **33**(4):658–686.

KIROUSIS, L., and C. PAPADIMITRIOU. 1985. The complexity of recognizing polyhedral scenes. *In* 26 Annual Symposium on Foundations of Computer Science.

KOOPMAN, B. O. 1980. Search and Screen: general principles with historical applications. Elmsford, N.Y: Pergaman Press.

MAVER, J., and R. BAJCSY. 1990. How to decide from the first view where to look next. *In* Proceedings of the DARPA Image Understanding Workshop.

PLANTINGA, W. H., and C. R. DYER. 1986. An algorithm for constructing the aspect graph. *In* 27 Annual Symposium on Foundations of Computer Science, 123–131.

TSOTSOS, J. 1990. Analyzing vision at the complexity level. The Behavioural and Brain Science **13**: 423–469.

TSOTSOS, J. 1992. Active versus passive perception, which is more efficient? International Journal of Computer Vision **7**(2).

WIXSON, L., and D. BALLARD. 1994. Using intermediate objects to improve the efficiency of visual search. IJCV **18**(3):209–230.

YE, Y., and J. K. TSOTSOS. 1995. Where to look next in 3d object search. *In* Proceedings of the IEEE International Symposium on Computer Vision.

YE, Y., and J. K. TSOTSOS. 1996. Sensor planning in 3d object search: its formulation and complexity. *In* Fourth International Symposium on Artificial Intelligence and Mathematics.

YE, Y., and J. K. TSOTSOS. 1997b. Knowledge difference and its influence on a search agent. *In* First International Conference on Autonomous Agents.

YE, Y. 1997. Sensor Planning for Object Search. Ph.D. Dissertation, University of Toronto, Toronto, Canada.

APPENDIX: LIST OF SYMBOLS

a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z

A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z

0,1,2,3,4,5,6,7,8,9,0

Ω, $\bigcup$, $\neq$, $\sum$, α, β, $\neg$, $\ldots$, $\prod$, $\bigcap$, $\bigcap$, $\times$, Δ, (,), [,], {, }, θ, δ, $>$, $<$, $\widehat{}$, $\varnothing$, σ, Θ