# On the Collaborative Object Search Team: a Formulation

Yiming Ye and John K. Tsotsos*

Department of Computer Science
University of Toronto
Toronto, Ontario, Canada M5S 1A4

**Abstract.** This paper gives a formulation of a collaborative object search team and studies the learning, interaction and organization within this multiagent environment. Each team member is assumed to be a mobile platform equipped with an active camera and recognition algorithms so that images of the environment can be taken and analyzed for the target object. The goal of the team is to find the target within a given time constraint. In order to do this, the agents must interact and collaborate with each other and must learn and modify the various cooperation styles based on the search results.

## 1 Introduction

Many researchers in the field of Distributed Artificial Intelligence are beginning to build agents that can work in a complex, dynamic multiagent domains[13]. Such domains include virtual theater[1], realistic virtual training environments [10] [11] [13], RoboCup robotic and virtual soccer [6] and robotic collaboration by observation [7].

This paper focuses on our recent research effort aimed at developing theories and systems for multiagent object search —– the task of searching for a $3D$ object in a $3D$ environment by a group of robots. Constructing multiagent object search systems requires facing many hard research challenges, such as goal-driven behavior, reactivity, real-time performance, planning, learning, coordination, and spatial and temporal reasoning. We believe that given its *real-world nature*, the multiagent object search task reflects many of the key characteristics of other real-world tasks in a dynamic multiagent environment.

* The authors are grateful to Eric Harley and the two reviewers of the ICMAS-96 workshop on Learning, Interactions and Organizations in Multiagent Environment for their valuable comments. This work is supported by a research fellowship from IBM Centre For Advanced Studies and by the 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. John K. Tsotsos is a fellow of the Canadian Institute for Advanced Research. Yiming Ye would like to thank Karen Bennet for her help.

Here we study the problem of object search by a team of *collaborative* robotic agents. By collaborative we mean that the agents are working together collaboratively in order to realize a common goal —– finding the target. A multi-agent object search system is quite different from the task of object search by a single robotic agent, on which we have done an extensive research and experiments [15] [16]. The multiagent team activities are not merely a union of simultaneous, independent individual activities —– each agent is not merely searching alone without considering the actions of other agents. Focusing on the multiagent aspect of the object search task brings to attention a number of challenging issues where learning is involved, such as:

- Interaction: How do agents communicate? How do agents coordinate in a team in order to find the target in the environment as early as possible? How should the patterns of interaction that characterize coordinated behavior be modeled, and how do agents learn from the effectiveness of the previous patterns of interaction?
- Knowledge representation and organization: How do search agents represent their local views of the world? How is the local knowledge updated or learned as a consequence of the agent's own action? How do search agents represent their local views of other agents? How do agents revise and learn beliefs about other agents based on exchanged information? How do agents organize knowledge about self, other agents, and the world such that newly learned facts can be easily integrated into the representation?
- Planning and Reasoning: How do agents plan their actions based on the knowledge about themselves, the knowledge about other agents, the knowledge learned from the interaction during the search process, and the knowledge learned from the effectiveness of its geometric relationships with other agents?

In this paper, we first formulate the task of multiagent object search and analyze its structure and complexity, then we study various issues concerning learning.

## 2 The Object Search Agent

In this section, we describe some concepts about the object search agent in the multiagent environment, such as the $3D$ region to be searched, the model of the search agent, the state parameters of a search agent, the search agent's local knowledge about the world, the operation of a search agent and its cost. We assume throughout this paper that there are a total of $m$ search agents $a_1$, $a_2$, ..., $a_m$ available in the object search team.

The search region $\Omega$ can be of any form, and it is assumed that we know the boundary of $\Omega$ and its internal geometric configuration exactly. In practice, we tessellate the region $\Omega$ into a series of elements $c_i$: $\Omega = \bigcup_{i=1}^{n} c_i$ and $c_i \bigcap c_j = 0$ for $i \neq j$. We call each of the element $c_i$ a cell of the environment and we assume in this paper that there are a total of $n$ cells. In addition to this, we introduce

another "cell" $c_{\text{out}}$ to refer to the region that is outside the search region $\Omega$. Usually, the search region is an office-like environment, and it is tessellated into little cubes of equal size. The goal of the search team is to determine which cube $c_i$ $(1 \leq i \leq n, \text{out})$ contains the target center. When we say $c_{\text{out}}$ contains the target center, we mean that the target is outside the region $\Omega$.

The model of the search agent (Fig. 1(a)(b)) is taken to be a mobile platform with a robotic head and a camera that can pan, tilt and zoom (Note: this model is based on the ARK robot and the Laser Eye [9]). The camera's image plane is assumed to be always coincident with its focal plane.

The state $s_a$ of a search agent $a$ is uniquely determined by 7 parameters $(x_a, y_a, z_a, w_a, h_a, p_a, t_a)$, where $(x_a, y_a, z_a)$ is the position of the camera center; $w_a, h_a$ are the width and height of the solid viewing angle of the camera; and $p_a, t_a$ are the the camera's viewing direction (Fig. 2). The position $(x_a, y_a, z_a)$ can be adjusted by moving the mobile platform. In the case of a mobile platform, $z_a$ is a fixed constant (the height of the camera), thus, only $x_a$ and $y_a$ are adjustable. The viewing angle size $\langle w_a, h_a \rangle$ can be adjusted by the zoom lens of the camera. Pan and tilt $\langle p_a, t_a \rangle$ can be adjusted by the motors on the robotic head.

The agent's knowledge about the possible target position is specified by a probability distribution function $\mathbf{p}$. The term $\mathbf{p}(a_i, c_j, \tau)$ gives the belief of agent $a_i$ regarding the probability that the center of the target is within cell $c_i$ at time $\tau$. Before the team search process begins, each agent should obtain its own probability distribution function for the search region. This represents the agent's local knowledge about the world. This knowledge should satisfy the following constraint,

$$\sum_{j=1}^{n} \mathbf{p}(a_i, c_j, \tau_0\text{-}) + \mathbf{p}(a_i, c_{\text{out}}, \tau_0\text{-}) = 1 \quad , \tag{1}$$

where $\tau_0$- means the time before the search process, and $\mathbf{p}(a_i, c_{\text{out}}, \tau_0\text{-})$ refers to agent $a_i$'s belief that the target is outside the search region before the search process. If agent $a_i$ is not able to determine whether the target is more likely to be outside the search region or inside the search region, then $\mathbf{p}(a_i, c_{\text{out}}, \tau_0\text{-}) = 0.5$. If agent $a_i$ has no knowledge about the possible target distribution within the search region, then it assumes a uniform distribution

$$\mathbf{p}(a_i, c_j, \tau_0\text{-}) = \frac{1 - \mathbf{p}(a_i, c_{\text{out}}, \tau_0\text{-})}{n} \quad , \tag{2}$$

for $j = 1, \ldots n$.

An operation $\mathbf{f}(a_i, s_{a_i}, r_{a_i}^{(j)})$ for agent $a_i$ entails two steps: (1) take a *perspective* projection image according to state $s_{a_i}$, and then (2) search the image for the target using the recognition algorithm $r_{a_i}^{(j)}$. We assume that each agent can have several recognition algorithms that can be used to detect the target; $r_{a_i}^{(j)}$ means agent $a_i$'s $j$th recognition algorithm.

The cost $\mathbf{t}(\mathbf{f})$ for an action $\mathbf{f} = \mathbf{f}(a_i, s_{a_i}, r_{a_i}^{(j)})$ gives the total time needed for $a_i$ to execute the action $\mathbf{f}(a_i, s_{a_i}, r_{a_i}^{(j)})$. It includes (1) time to manipulate the hardware to the status $s_{a_i}$ specified by $\mathbf{f}$; (2) time to take a picture using the

camera on $a_i$; (3) time to update $a_i$'s inner representation of the world; and (4) time to run the recognition algorithm $r_{a_i}^{(j)}$ specified by $\mathbf{f}$. We assume that: (2) and (3) are same for all the actions; (4) is known for any recognition algorithm and is constant for a given recognition algorithm.

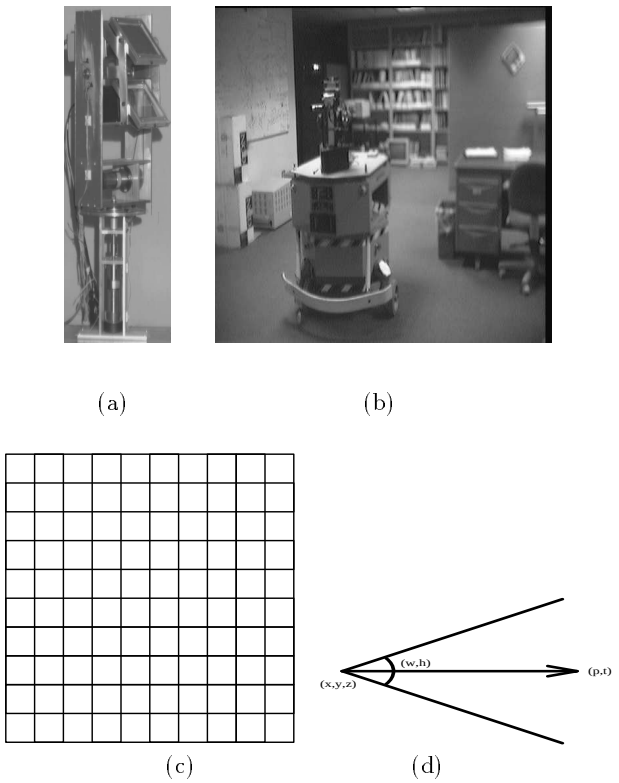


**Fig. 1.** (a) The Laser Eye (the active camera model for the search agent). At the top is the Optech laser range finder; at the bottom is the zoom and focus controlled lens. The two mirrors are used to ensure collinearity of effective optical axes of the camera lens and the range finder. The pan-tilt unit is operated by two DC servocontrolled motors from Micromo Electronics, equipped with gearboxes and optical encoders. (b) The ARK mobile platform (the mobile platform for the search agent). It is a 3 wheeled Cybermotion K2A platform equipped with a small number of sonar transducers.The Laser Eye can be mounted on a robotic head on the platform. The ARK mobile platform can move freely in an office-like environment or a known industrial environment. (c) $2D$ illustration of the tessellation of the search region $\Omega$. (d) $2D$ illustration of the state $s_a$ of a search agent $a$. Where $(x, y, z)$ are the coordinates of the camera center, $(w, h)$ is the viewing angle size of the camera, and $(p, t)$ is the viewing direction of the camera.

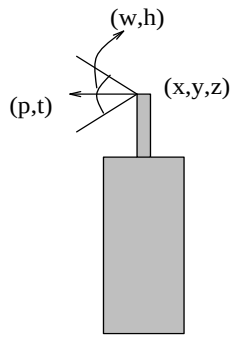


**Fig. 2.** A 2D illustration of the state of the hardware. Coordinate $(x, y, z)$ indicate the position of the center of the camera; $(w, h)$ is the viewing angle size of the camera; and $(p, t)$ is the viewing direction of the camera.

## 3 Awareness and Knowledge Adaptation

Awareness means that the agent has knowledge about itself and the world. Knowledge adaptation means that the agent perceives the environment through actions and incorporates the results of perception into its own internal representation.

Each agent $a_i$ has inner knowledge about its ability to detect the target by applying a given action $\mathbf{f}$. The detection ability is represented by the detection function $\mathbf{b}(a_i, c_j, \mathbf{f})$, which gives the conditional probability that agent $a_i$ will detect the target given that the center of the target is located within cell $c_j$, and the operation is $\mathbf{f}$. This function is approximated by assuming that the target center is located *at the center* of $c_j$. For any operation, if the projection of the center of the cell $c_j$ is outside the image, we assume $\mathbf{b}(a_i, c_j, \mathbf{f}) = 0$; if the cell is occluded or too far from the camera or too near to the camera, we also have $\mathbf{b}(a_i, c_j, \mathbf{f}) = 0$. In general [17], $\mathbf{b}(a_i, c_j, \mathbf{f})$ is determined by various factors, such as illumination intensity, occlusion, and orientation, etc. The value of the detection function $\mathbf{b}(a_i, c_j, \mathbf{f})$ for operation $\mathbf{f}$ can be obtained by transforming from a pre-recorded standard detection function for the recognition algorithm used by $\mathbf{f}$.

It is obvious that agent $a_i$'s knowledge about the probability of detecting the target by applying action $\mathbf{f}$ by itself is given by

$$P(\mathbf{f}) = \sum_{j=1}^{n} \mathbf{p}(a_i, c_j, \tau_{\mathbf{f}}) \mathbf{b}(a_i, c_j, \mathbf{f}) \quad , \tag{3}$$

where $\tau_{\mathbf{f}}$ is the time just before $\mathbf{f}$ is applied. It is obvious that if a cell $c_j$ is outside the field of view of the camera determined by $\mathbf{f}$ or is occluded with respect to the geometry specified by $\mathbf{f}$, then $\mathbf{b}(a_i, c_j, \mathbf{f}) = 0$. We use

$$\Omega(\mathbf{f}) = \left\{ c \mid c \in \Omega \wedge \mathbf{b}(a_i, c, \mathbf{f}(a_i, s_{a_i}, r_{a_i}^{(j)})) \neq 0 \right\} \tag{4}$$

to represent those cells whose detection function values are not zero with respect to $a_i$ and $\mathbf{f}$. $\Omega(\mathbf{f})$ is called the *influence range* for $\mathbf{f}$. Of course,

$$P(\mathbf{f}) = \sum_{c \in \Omega(\mathbf{f})} \mathbf{p}(a_i, c, \tau_{\mathbf{f}}) \mathbf{b}(a_i, c, \mathbf{f}) \ . \tag{5}$$

For any agent $a_i$, its beliefs on the possible target positions are represented by $\mathbf{p}(a_i, c, \tau)$, and these beliefs change over time as the agent perceives the world. If an action by agent $a_i$ finds the target, then $a_i$ will report the result and the task is accomplished. If an action by agent $a_i$ fails to detect the target, then $a_i$ needs to incorporate this result into its *local* knowledge representation regarding the target distribution. This is the process of *learning from action*. The agent uses Bayes' formula to update its *local* knowledge base. Let $\alpha_j$ be the event that the center of the target is in cell $c_j$, and $\alpha_{\text{out}}$ be the event that the center of the target is outside the search region. Let $\beta$ be the event that after applying a recognition action $\mathbf{f}$ by agent $a_i$, the recognizer successfully detects the target. Then $P(\neg\beta \mid \alpha_j)$ gives the probability of not detecting the target by action $\mathbf{f}$ of agent $a_i$ given that the target center is within cube $c_j$. Since $\mathbf{b}(a_i, c_j, \mathbf{f})$ gives the probability of detecting the target by action $\mathbf{f}$ given that the target center is within cube $c_j$. We have

$$P(\neg\beta \mid \alpha_j) = 1 - \mathbf{b}(a_i, c_j, \mathbf{f}) \ . \tag{6}$$

It is obvious that $P(\alpha_j \mid \neg\beta)$ gives the probability that the target center is within the cube $c_j$ given that $\mathbf{f}$ failed to detect the target. If we represent the updated probability for cube $c_j$ as $\mathbf{p}(a_i, c_j, \tau_{\mathbf{f}_+})$, where $\tau_{\mathbf{f}_+}$ is the time after $\mathbf{f}$ is applied. Then we should have

$$\mathbf{p}(a_i, c_j, \tau_{\mathbf{f}_+}) = P(\alpha_j \mid \neg\beta) \ . \tag{7}$$

Since the events $\alpha_1, \ldots, \alpha_n, \alpha_{\text{out}}$ are mutually complementary and exclusive, from Bayes' formula, we get

$$P(\alpha_i \mid \neg\beta) = \frac{P(\alpha_i) P(\neg\beta \mid \alpha_i)}{\sum_{j=1}^{n,\text{out}} P(\alpha_j) P(\neg\beta \mid \alpha_j)} \ , \tag{8}$$

for $i = 1, \ldots, n, \text{out}$

Thus, by replacing $P(\alpha_i \mid \neg\beta)$ with $\mathbf{p}(a_i, c_j, \tau_{\mathbf{f}_+})$, $P(\alpha_i)$ with $\mathbf{p}(a_i, c_j, \tau_{\mathbf{f}})$, and $P(\neg\beta \mid \alpha_i)$ with $1 - \mathbf{b}(a_i, c_j, \mathbf{f})$, we obtain the following probability updating rule

$$\mathbf{p}(a_i, c_j, \tau_{\mathbf{f}_+}) \leftarrow \frac{\mathbf{p}(a_i, c_j, \tau_{\mathbf{f}}) \left(1 - \mathbf{b}(a_i, c_j, \mathbf{f})\right)}{\sum_{k=1}^{n,\text{out}} \mathbf{p}(a_i, c_k, \tau_{\mathbf{f}}) \left(1 - \mathbf{b}(a_i, c_k, \mathbf{f})\right)} \ , \tag{9}$$

where $j = 1, \ldots, n, \text{out}$.

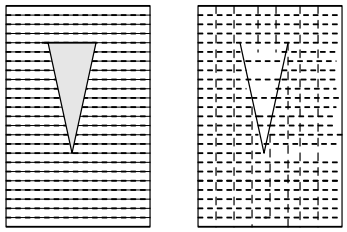

**Fig. 3.** A 2D illustration of the belief updating process. When an agent applies an action in the environment but fails to detect the target, it's belief on the target distribution within the influence range of the action is decreased and its belief on the target distribution outside the influence range of the action is increased. This phenomena is modeled by Bayes Law.

## 4 The Agent's Sensing Ability

In this section we discuss how to evaluate the detection function for each agent and explain the concept of *effective volume* for an action of a given agent.

The value of the detection function $\mathbf{b}(a_i, c_j, \mathbf{f})$ for action $\mathbf{f}$ can be obtained by transforming from a pre-recorded standard detection function for $a_i$ and $c_j$. For a given camera hardware (which is determined by the corresponding agent that uses this camera) and a given recognition algorithm $r$, the standard detection function $\mathbf{b_0}((\theta, \delta, l), \langle r, w, h\rangle)$ gives a measure of the detecting ability of the recognition algorithm $r$ when no previous action has been applied. In this function, $\langle w, h\rangle$ is the viewing angle size of the camera; $(\theta, \delta, l)$ is the relative position of the center of the target to the camera:

$$\theta = \arctan\left(\frac{x}{z}\right) \quad , \tag{10}$$

$$\delta = \arctan\left(\frac{y}{z}\right) \quad , \tag{11}$$

$$l = z \quad , \tag{12}$$

where $(x, y, z)$ is the coordinates of the target center in the camera coordinate system. The value of $\mathbf{b_0}((\theta, \delta, l), \langle r, w, h\rangle)$ can be obtained by experiments. We can first put the target at $(\theta, \delta, l)$ and then perform experiments under various conditions, such as light intensity, background situation, and the relative orientation of the target with respect to the camera center. The final value of $b_0$ is the total number of successful recognitions divided by the total number of experiments. These values can be stored in a look up table indexed by $\theta, \delta, l$ and retrieved when needed. Sometimes we may approximate these values by analytic formulas.

We only need to record the detection values of one angle size $\langle w_0, h_0\rangle$. Those of other sizes can be approximately transformed to those of size $\langle w_0, h_0\rangle$. Suppose $(\theta, \delta, l)$ is the target position for angle size $\langle w, h\rangle$, we want to find the value $(\theta_0, \delta_0, l_0)$ for angle size $\langle w_0, h_0\rangle$ such that

$$\mathbf{b_0}((\theta_0, \delta_0, l_0), \langle r, w_0, h_0 \rangle) \approx \mathbf{b_0}((\theta, \delta, l), \langle r, w, h \rangle) \ . \tag{13}$$

To guarantee this, the images taken with parameter $\langle \theta_0, \delta_0, l_0, w_0, h_0 \rangle$ and $\langle \theta, \delta, l, w, h \rangle$ should be almost same. Thus, the area and the position of the projected target image on the image plane should be almost the same for both images. We obtain

$$l_0 = l \sqrt{\frac{\tan\left(\frac{w}{2}\right) \tan\left(\frac{h}{2}\right)}{\tan\left(\frac{w_0}{2}\right) \tan\left(\frac{h_0}{2}\right)}} \ , \tag{14}$$

$$\theta_0 = \arctan\left[\tan(\theta) \frac{\tan\left(\frac{w_0}{2}\right)}{\tan\left(\frac{w}{2}\right)}\right] \ , \tag{15}$$

$$\delta_0 = \arctan\left[\tan(\delta) \frac{\tan\left(\frac{w_0}{2}\right)}{\tan\left(\frac{w}{2}\right)}\right] \ . \tag{16}$$

When the configurations of two operations are very similar, they might correlate with each other (refer to [17] for detail). Repeated actions are avoided during the search process. When independence is assumed, $\mathbf{b}(a_i, c_j, \mathbf{f})$ is calculated as follows. First, calculate the corresponding $(\theta, \delta, l)$ of the center of $c_j$ with respect to operation $\mathbf{f}$ of agent $a_i$. Second, transform $(\theta, \delta, l)$ into the corresponding $(\theta_0, \delta_0, l_0)$ of angle size $\langle w_0, h_0 \rangle$. Third, retrieve the detection value from the look up table, or get the detection value from a formula.

Now we explain the concept of effective volume $EV(\mathbf{f})$ for a given action $\mathbf{f}$. The ability of the recognition algorithm and the value of the detection function are influenced by the image size of the target. Only when the target can be totally brought into the field of view of the camera and the features detected within a certain precision, can the recognition algorithm be expected to function correctly. So, for an agent's action with a given recognition algorithm and a fixed viewing angle, the probability of successfully recognizing the target is high only when the target is within a certain distance range. We call this range the *effective range* and the viewing volume within this range the *effective volume*. For a given action, only those cubes that are within the *effective volume* can be examined with high detection probability. Figure 4 illustrates in $2D$ the effective volume for a given action.

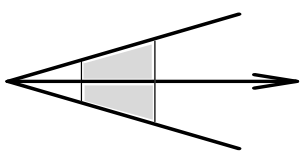

**Fig. 4.** $2D$ illustration of the effective volume of a given action (the shaded area).

Please note that the influence range $\Omega(\mathbf{f})$ of $\mathbf{f}$ is different from the effective volume $EV(\mathbf{f})$ of $\mathbf{f}$, $\Omega(\mathbf{f}) \neq EV(\mathbf{f})$. The influence range $\Omega(\mathbf{f})$ refers to the region such that if the target center is within this region, the probability of detecting the target is not 0. The effective volume $EV(\mathbf{f})$ refers to the region such that if the target center is within this region, the probability of detecting the target is high enough (higher than a given threshold). So, $EV(\mathbf{f}) \subseteq \Omega(\mathbf{f})$.

## 5 The Collaborative Multi-agent Search Team: a Global View

A global view of the activities of the multiagent search team is as follows:

- Before the search process, each agent $a_i$ proposes an initial target distribution ($\mathbf{p}(a_i, c, \tau_0\text{-})$ for all $c \in \Omega$) according to its own perception of the world (Fig. 5).

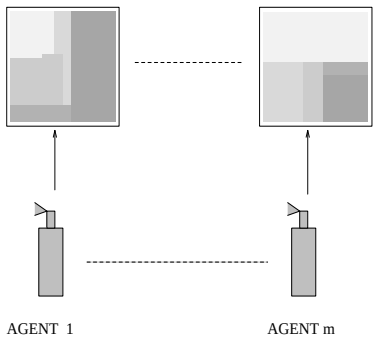


**Fig. 5.** Each agent has an initial target distribution.

- The team forms the common initial target distribution ($\mathbf{p}(c, \tau_0)$ for all $c \in \Omega$) by combining the initial distributions of all the agents together (Fig. 6).
- Each agent $a_i$ uses the common initial distribution $\mathbf{p}(c, \tau_0)$ as its own initial distribution $\mathbf{p}(a_i, c, \tau_0)$ used during the search process (Fig. 7). That is, each agent $a_i$ performs the substitution

$$\forall c, \mathbf{p}(a_i, c, \tau_0) \leftarrow \mathbf{p}(c, \tau_0) \quad . \tag{17}$$

- The selection and execution of actions for different agents occurs in parallel (Fig. 8). However, action selection and execution for each agent is sequential. This means that a single agent can select and execute the next action only after the current action is finished.
  - Each agent selects its own action based on its own knowledge and the knowledge *learned* from interaction and communication with other agents.

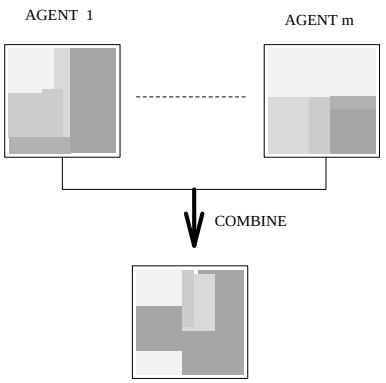


**Fig. 6.** Common initial target distribution formation.

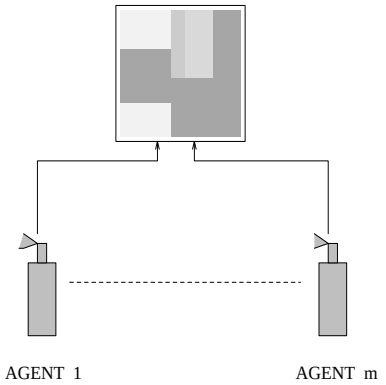


**Fig. 7.** Initial distribution used before the search process for each agent.

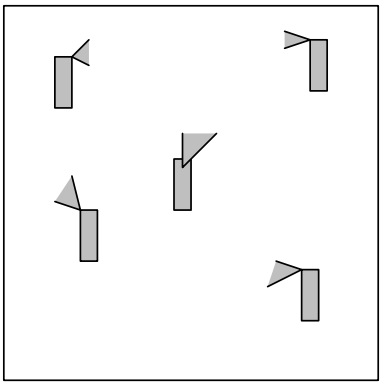

**Fig. 8.** Agents search for the target together at the same time.

- If the selected action succeeds, the corresponding agent announces the success result immediately and the team search process terminates.
- If the selected action fails, the corresponding agent will broadcast the failure news to the rest of the agents, together with the action's state parameters and effective volume. Then the corresponding agent updates its own knowledge base about the target distribution and starts selecting its next action.

– The team search process will terminate when the available time is used up.

Let $\Omega_{a_i}^{\mathbf{F}}$ be the set of all the actions which can be selected by agent $a_i$ $(1 \leq i \leq m)$. Suppose

$$\mathbf{F}_{a_i} = \left\{ \mathbf{f}_{a_i}^{(1)}, \mathbf{f}_{a_i}^{(2)}, \ldots, \mathbf{f}_{a_i}^{(N_{a_i})} \right\} \tag{18}$$

is the set of actions actually selected by agent $a_i$ $(1 \leq i \leq m)$ from $\Omega_{a_i}^{\mathbf{F}}$ during the search process, where $N_{a_i}$ is the number of actions selected by agent $a_i$. Then we call $\mathbf{F}_{a_i}$ the *effort allocation* for agent $a_i$. The *effort allocation for the search team* is the union of the effort allocations of all the agents of the team

$$\mathbf{F} = \mathbf{F}_{a_1} \bigcup \mathbf{F}_{a_2} \bigcup \ldots \bigcup \mathbf{F}_{a_m} \quad . \tag{19}$$

Let $P[\mathbf{F}]$ be the probability of detecting the target by the team effort allocation $\mathbf{F}$ and let $K$ be the total time that can be used by the team. Then the task of object search by a multiagent team can be defined as finding a team effort allocation $\mathbf{F}$ such that $P[\mathbf{F}]$ is maximized within the time constraint $K$.

### 5.1 The Initial Common Target Distribution

Before the search process, each agent has some knowledge of the whereabouts of the target. This knowledge can come from different sources. Because the object search team is collaborative, the team should form a common initial target distribution by considering each agent's knowledge. One way to get a common initial distribution is to use the Dempster-Shafer theory [12]. Here we simply generate the initial common target distribution $\mathbf{p}(c, \tau_0)$ (for $c \in \Omega$) by forming a weighted sum of the probability distributions of all the agents. This means that for all $c \in \Omega$, perform the following operation:

$$\mathbf{p}(c, \tau_0) \leftarrow \sum_{k=1}^{m} w_k \mathbf{p}(a_k, c, \tau_0 -) \quad . \tag{20}$$

The weights $w_1, \ldots, w_m$ must satisfy

$$\sum_{k=1}^{m} w_k = 1 \qquad \text{and} \qquad w_k \geq 0, \ 1 \leq k \leq m \quad . \tag{21}$$

It is obvious that the newly generated target distribution satisfies

$$\begin{aligned}\sum_{i=1}^{n,\text{out}} \mathbf{p}(c_i, \tau_0) &= \sum_{i=1}^{n,\text{out}} \left( \sum_{k=1}^{m} w_k \mathbf{p}(a_k, c_i, \tau_0 \text{-}) \right) \\ &= \sum_{k=1}^{m} \left( w_k \sum_{i=1}^{n,\text{out}} \mathbf{p}(a_k, c_i, \tau_0 \text{-}) \right) \\ &= \sum_{k=1}^{m} w_k \\ &= 1 \ . \end{aligned} \tag{22}$$

In order to make this strategy work effectively, it is important to obtain a set of values $w_1, \ldots, w_m$ which reflect the relative credibility of the target distributions from different agents. If the multiagent team has not been working together before, then the relative reliability is not known and we set

$$w_1 = \ldots = w_m = \frac{1}{m} \ . \tag{23}$$

In general, more reasonable values of $w_k$ can be obtained by training and learning, as follows.

A single training set is obtained by running the team search process for $m$ cases, each case using a different agent's initial probability distribution as the initial global probability distribution (Note: there is no weighted sum operation).

For each case of testing, we can obtain a time needed to detect the target for the search team. Suppose that the time needed to detect the target is minimum for the case when agent $k$'s initial probability distribution is used. Then we say that agent $k$ wins this training set. The result illustrates that agent k's knowledge sources are more reliable than other agent's knowledge sources for this training.

So, for each training set we can select an agent whose initial knowledge is most reliable. Suppose we have performed $Q$ training sets, then the value of $w_k$ for agent $k$ will be the number of trainings won by agent $k$ divided by $Q$.

### 5.2 Detection Probability for an Effort Allocation

We derive a general expression to calculate $P[\mathbf{F}]$ in this section in a global view. By global view we mean that we track in a globally consistent fashion the probability of detecting the target by applied actions and update the probability distribution after each action is applied. In other words, if an action can detect the target, then the success news is broadcast at the moment this action is selected, and if an action cannot find the target, then the probability distribution is updated at the moment this action is selected.

In order to get a general expression, let us first consider a special case by assuming that the actions in $\mathbf{F}$ are applied in sequence (Note: in real applications, the actions of different agents can be executed at the same time).

Let

$$\mathbf{F} = \{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_q\} \quad , \tag{24}$$

where

$$q = N_{a_1} + \ldots + N_{a_m} = \mid \mathbf{F}_{a_1} \mid + \mid \mathbf{F}_{a_2} \mid + \ldots + \mid \mathbf{F}_{a_m} \mid \tag{25}$$

and

$$\tau_{\mathbf{f}_1} < \tau_{\mathbf{f}_2} < \ldots < \tau_{\mathbf{f}_q} \quad . \tag{26}$$

The correspondence of the action index $i$ for $\mathbf{f}_i$ with the actions executed by each agent is illustrated in the following

$$\mathbf{F} = \left\{ \overbrace{\mathbf{f}_1, \ldots, \mathbf{f}_{N_{a_1}}}^{\mathbf{F}_{a_1}}, \ldots, \overbrace{\mathbf{f}_{N_{a_1}+\ldots+N_{a_{m-1}}+1}, \ldots, \mathbf{f}_q}^{\mathbf{F}_{a_m}} \right\} \quad . \tag{27}$$

In other words, $\mathbf{f}_{a_i}^{(j)}$, the $j$th action of agent $a_i$ in its own effort allocation $\mathbf{F}_{a_i}$, is represented as $\mathbf{f}_{N_{a_1}+N_{a_2}+\ldots+N_{a_{i-1}}+j}$ in above representation (27).

The search process is as follows: at the beginning, $\mathbf{f}_1$ is selected and applied, the probability distribution used by $\mathbf{f}_1$ is the initial common target distribution. If operation $\mathbf{f}_i$ $(1 \leq i \leq q)$ succeeds, then the news is broadcast to all agents and the search process terminates; if operation $\mathbf{f}_i$ fails, then the probability distribution is updated, and this updated probability distribution will be used by $\mathbf{f}_{i+1}$. It is clear that the expected probability of detecting the target by allocation $\mathbf{F}$ is:

$$P[\mathbf{F}] = P(\mathbf{f}_1) + [1 - P(\mathbf{f}_1)]P(\mathbf{f}_2) + \ldots + \left\{ \prod_{i=1}^{q-1} [1 - P(\mathbf{f}_i)] \right\} P(\mathbf{f}_q) \quad . \tag{28}$$

Formula (28) is not helpful for finding regularities about P[$\mathbf{F}$], and it is not appropriate in defining the probability of detecting the target by an effort allocation when the actions of different agents can be executed in parallel. In order to obtain a representation of the detection probability in the general sense, we need to do some algebraic transformations.

The initial probability distribution is denoted as

$$\mathbf{p}^{[0]}(c_1), \mathbf{p}^{[0]}(c_2), \ldots, \mathbf{p}^{[0]}(c_n), \mathbf{p}^{[0]}(c_{\text{out}}) \quad . \tag{29}$$

Of course, we have $\mathbf{p}^{[0]}(c_i) = \mathbf{p}(c_i, \tau_0)$ for $i = 1, \ldots, n, \text{out}$. Where $\mathbf{p}(c_i, \tau_0)$ $(i = 1, \ldots, n, \text{out})$ are the initial common target distribution formed by integrating the knowledge of all the participating agents before the search process. The probability distributions $\mathbf{p}^{[0]}(c_i)$ $(i = 1, \ldots, n, \text{out})$ are used by the first action $\mathbf{f}_1$ of $\mathbf{F}$. After the application of operation $\mathbf{f}_1$, the distributions are denoted by $\mathbf{p}^{[1]}(c_1)$, $\mathbf{p}^{[1]}(c_2)$, ..., $\mathbf{p}^{[1]}(c_n)$, $\mathbf{p}^{[1]}(c_{\text{out}})$. Generally, after the application of the $ith$ operation $\mathbf{f}_i$, the distributions are denoted by $\mathbf{p}^{[i]}(c_1)$, $\mathbf{p}^{[i]}(c_2)$, ..., $\mathbf{p}^{[i]}(c_n)$, $\mathbf{p}^{[i]}(c_{\text{out}})$, where $1 \leq i \leq q$. If the target is detected by $\mathbf{f}_i$, then $\mathbf{p}^{[i]}(c^*) = 1$ and

$\mathbf{p}^{[i]}(c) = 0$ for $c \neq c^*$, where $c^*$ is the cube that contains the target center. Otherwise, $\mathbf{p}^{[i]}(c_j)$ $(1 \leq j \leq n, \text{out})$ is obtained by using the probability updating rule (9). Let $P(\mathbf{f}_i)$ represent the probability of detecting the target by applying the action $\mathbf{f}_i$ with respect to the effort allocation $\mathbf{F}$. Then of course we have

$$P(\mathbf{f}_i) = \sum_{j=1}^{n} \mathbf{p}^{[i-1]}(c_j)\mathbf{b}(c_j, \mathbf{f}_i) \quad . \tag{30}$$

In (30), $\mathbf{b}(c_j, \mathbf{f}_i) = \mathbf{b}(a, c_j, \mathbf{f}_i)$, $a$ is the agent executing action $\mathbf{f}$.

Let $P^{[0]}(\mathbf{f}_i)$ represents the probability of detecting the target by applying the action $\mathbf{f}_i$ when no action has been applied before. This means that we assume $\mathbf{f}_i$ is the first action to be applied, thus the probability distribution used by $\mathbf{f}_i$ is $\mathbf{p}^{[0]}(c_1)$, ..., $\mathbf{p}^{[0]}(c_n)$, and $\mathbf{p}^{[0]}(c_{\text{out}})$. We have

$$P^{[0]}(\mathbf{f}_i) = \sum_{j=1}^{n} \mathbf{p}^{[0]}(c_j)\mathbf{b}(c_j, \mathbf{f}_i) \quad . \tag{31}$$

After some calculation, we obtain a general expression of the target probability distribution as stated in the following lemma.

**Lemma 1.** *For allocation* $\mathbf{F}$*, if* $\mathbf{f}_1$, ..., $\mathbf{f}_k$ *have not detected the target, then we have*

$$\mathbf{p}^{[k]}(c) = \frac{\mathbf{p}^{[0]}(c)[1 - \mathbf{b}(c, \mathbf{f}_1)] \ldots [1 - \mathbf{b}(c, \mathbf{f}_k)]}{(1 - P(\mathbf{f}_1))(1 - P(\mathbf{f}_2)) \ldots (1 - P(\mathbf{f}_k))} \quad . \tag{32}$$

By replacing the result of Lemma 1 and (31) into (30), after some complex transformations we obtain a general expression for the detection probability for any action in $\mathbf{F}$ as follows.

**Lemma 2.** *For allocation* $\mathbf{F}$*, we have*

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

$$\vdots$$

$$+(-1)^{k-1}\sum_{c\in\Omega(\mathbf{f}_1\ldots\mathbf{f}_{k-1}\mathbf{f}_k)}\mathbf{p}^{[0]}(c)\mathbf{b}(c,\mathbf{f}_1)\mathbf{b}(c,\mathbf{f}_2)\ldots\mathbf{b}(c,\mathbf{f}_{k-1})\mathbf{b}(c,\mathbf{f}_k)\Bigg\}\ , \tag{33}$$

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

By replacing the result of Lemma 2 into (28), we can prove the following important lemma.

**Lemma 3.** *For allocation* $\mathbf{F}$*, we have*

$$\begin{aligned}
P(\mathbf{F}) &= \sum_{i=1}^{q}\left(\sum_{j=1}^{n}\mathbf{p}(c_j,\tau_0)\mathbf{b}(c_j,\mathbf{f}_i)\right)\\
&+(-1)^{2+1}\sum_{1\le i_1<i_2\le q}\left(\sum_{c\in\Omega(\mathbf{f}_{i_1}\mathbf{f}_{i_2})}\mathbf{p}(c,\tau_0)\mathbf{b}(c,\mathbf{f}_{i_1})\mathbf{b}(c,\mathbf{f}_{i_2})\right)\\
&+\\
&\vdots\\
&+(-1)^{r+1}\sum_{1\le i_1<i_2<\ldots<i_r\le q}\left(\sum_{c\in\Omega(\mathbf{f}_{i_1}\mathbf{f}_{i_2}\ldots\mathbf{f}_{i_r})}\mathbf{p}(c,\tau_0)\mathbf{b}(c,\mathbf{f}_{i_1})\mathbf{b}(c,\mathbf{f}_{i_2})\ldots\mathbf{b}(c,\mathbf{f}_{i_r})\right)\\
&+\\
&\vdots\\
&+(-1)^{q+1}\left(\sum_{c\in\Omega(\mathbf{f}_1\mathbf{f}_2\ldots\mathbf{f}_q)}\mathbf{p}(c,\tau_0)\mathbf{b}(c,\mathbf{f}_1)\ldots\mathbf{b}(c,\mathbf{f}_q)\right)\ .
\end{aligned}\tag{34}$$

Lemma 3 explicitly reveals a very important property of $P[\mathbf{F}]$: the order of actions been applied does not have any influence on the value of $P[\mathbf{F}]$. This agrees with common knowledge, because whether an action $\mathbf{f}_{a_i}^{(j)}=\mathbf{f}(a_i,s_{a_i},r_{a_i}^{(j')})$ can detect the target is determined only by the agent $a_i$, the state parameters $s_{a_i}$ of $a_i$, the recognition algorithm $r_{a_i}^{(j')}$ used to analyze the image taken by $a_i$, the real target position and the real situation (lighting, background, etc.) surrounding the target. Since these factors will not change with time, the time when the action $\mathbf{f}(a_i,s_{a_i},r_{a_i}^{(j')})$ is applied has no effect on whether this action can detect the target. If action $\mathbf{f}$ detects the target in one application order in the effort allocation $\mathbf{F}$, then, it will detect the target in another application order in the effort allocation $\mathbf{F}$; if action $\mathbf{f}$ fails to detect the target in one application order, then, it will fail to detect the target in another application order.

Since (34) is independent of order of actions, it is used as the general definition of the expected probability of detecting the target by an effort allocation $\mathbf{F}$ in a multiagent search team.

In a real team search process where actions can be applied concurrently by different agents, we will use Formula (34) to calculate the expected probability of detecting the target by the given effort allocation.

In other words, suppose $\mathbf{F} = \mathbf{F}_{a_1} \bigcup \mathbf{F}_{a_2} \bigcup \ldots \bigcup \mathbf{F}_{a_m}$ is the effort allocation of the search team, where $\mathbf{F}_{a_i}$ $(1 \leq i \leq m)$ is defined in (18). Let $\sum_{\mathbf{F}(f_{a_{i_1}}^{(j_1)} \ldots f_{a_{i_k}}^{(j_k)})}$ be the sum operation over all the different subset $\{f_{a_{i_1}}^{(j_1)}, \ldots, f_{a_{i_k}}^{(j_k)}\}$ of $\mathbf{F}$, where $1 \leq k \leq m$. Let $|\mathbf{F}| = N_{a_1} + \ldots + N_{a_m}$ be the number of actions in the team effort allocation $\mathbf{F}$. Then the probability of detecting the target $P[\mathbf{F}]$ by the effort allocation $\mathbf{F}$ is calculated by the following formula:

$$
\begin{aligned}
P(\mathbf{F}) = {} & (-1)^{1+1} \sum_{\mathbf{F}(\mathbf{f}_{a_i}^{(j)})} \left( \sum_{c \in \Omega(\mathbf{f}_{a_i}^{(j)})} \mathbf{p}(c, \tau_0) \mathbf{b}(a_i, c, \mathbf{f}_{a_i}^{(j)}) \right) \\
& + (-1)^{2+1} \sum_{\mathbf{F}(\mathbf{f}_{a_{i_1}}^{(j_1)} \mathbf{f}_{a_{i_2}}^{(j_2)})} \left( \sum_{c \in \Omega(\mathbf{f}_{a_{i_1}}^{(j_1)} \mathbf{f}_{a_{i_2}}^{(j_2)})} \mathbf{p}(c, \tau_0) \mathbf{b}(a_{i_1}, c, \mathbf{f}_{a_{i_1}}^{(j_1)}) \mathbf{b}(a_{i_2}, c, \mathbf{f}_{a_{i_2}}^{(j_2)}) \right) \\
& + \\
& \vdots \\
& + (-1)^{r+1} \sum_{\mathbf{F}(\mathbf{f}_{a_{i_1}}^{(j_1)} \ldots \mathbf{f}_{a_{i_r}}^{(j_r)})} \left( \sum_{c \in \Omega(\mathbf{f}_{a_{i_1}}^{(j_1)} \ldots \mathbf{f}_{a_{i_r}}^{(j_r)})} \mathbf{p}(c, \tau_0) \mathbf{b}(a_{i_1}, c, \mathbf{f}_{a_{i_1}}^{(j_1)}) \ldots \mathbf{b}(a_{i_r}, c, \mathbf{f}_{a_{i_r}}^{(j_r)}) \right) \\
& + \\
& \vdots \\
& + (-1)^{|\mathbf{F}|+1} \left( \sum_{c \in \Omega(\mathbf{f}_{a_1}^{(1)} \ldots \mathbf{f}_{a_m}^{(N_{a_m})})} \mathbf{p}(c, \tau_0) \mathbf{b}(a_1, c, \mathbf{f}_{a_1}^{(1)}) \ldots \mathbf{b}(a_m, c, \mathbf{f}_{a_m}^{(N_{a_m})}) \right) . \qquad (35)
\end{aligned}
$$

We give an intuitive representation of (35). It is clear that $P^{[0]}(\mathbf{f}_{a_i}^{(j)}) = \sum_{c \in \Omega(\mathbf{f}_{a_i}^{(j)})} \mathbf{p}(c, \tau_0) \mathbf{b}(a_i, c, \mathbf{f}_{a_i}^{(j)})$. If we *define* the initial intersect probability of detecting the target by a set of actions $\mathbf{f}_{a_{i_1}}^{(j_1)}, \ldots, \mathbf{f}_{a_{i_r}}^{(j_r)}$ as

$$
P^{[0]}\left( \langle \mathbf{f}_{a_{i_1}}^{(j_1)} \ldots \mathbf{f}_{a_{i_r}}^{(j_r)} \rangle \right) = \sum_{c \in \Omega(\mathbf{f}_{a_{i_1}}^{(j_1)} \ldots \mathbf{f}_{a_{i_r}}^{(j_r)})} \mathbf{p}(c, \tau_0) \mathbf{b}(a_{i_1}, c, \mathbf{f}_{a_{i_1}}^{(j_1)}) \ldots \mathbf{b}(a_{i_r}, c, \mathbf{f}_{a_{i_r}}^{(j_r)}) . \qquad (36)
$$

Then, (35) can be more intuitively written as

$$
\begin{aligned}
P(\mathbf{F}) = {} & \sum_{i,j} P^{[0]}(\mathbf{f}_{a_i}^{(j)}) - \sum_{i_1, j_1, i_2, j_2} P^{[0]}\left( \langle \mathbf{f}_{a_{i_1}}^{(j_1)} \mathbf{f}_{a_{i_2}}^{(j_2)} \rangle \right) \\
& + \ldots + (-1)^{r+1} \sum_{i_1, j_1, \ldots, i_r, j_r} P^{[0]}\left( \langle \mathbf{f}_{a_{i_1}}^{(j_1)} \ldots \mathbf{f}_{a_{i_r}}^{(j_r)} \rangle \right) + \ldots
\end{aligned}
$$

$$+(-1)^{|\mathbf{F}|+1} P^{[0]}\left(\langle \mathbf{f}_{a_1}^{(1)} \ldots \mathbf{f}_{a_m}^{(N_{a_m})} \rangle\right) \quad . \tag{37}$$

### 5.3 The Task of Multiagent Search and its Complexity: a Global View

The object search task for a multiagent object search team $\{a_1, \ldots, a_m\}$ can be defined as the task for each agent $a_i$ to find a set of operations $\mathbf{F}_{a_i}$, such that the expected probability of detecting the target $P[\mathbf{F}]$ calculated by (35) for the effort allocation $\mathbf{F}$ (formed by the union of the effort allocations of all the agents of the team)

$$\mathbf{F} = \mathbf{F}_{a_1} \bigcup \mathbf{F}_{a_2} \bigcup \ldots \bigcup \mathbf{F}_{a_m} \tag{38}$$

is maximized with the constraint that the cost for the effort allocation $\mathbf{F}_{a_i}$ for each agent $a_i$ is less than or equal to a given constant $K$:

$$\begin{cases} \mathbf{t}(\mathbf{f}_{a_1^{(1)}}) + \mathbf{t}(\mathbf{f}_{a_1^{(2)}}) + \ldots + \mathbf{t}(\mathbf{f}_{a_1^{(N_{a_1})}}) \leq K \\ \mathbf{t}(\mathbf{f}_{a_2^{(1)}}) + \mathbf{t}(\mathbf{f}_{a_2^{(2)}}) + \ldots + \mathbf{t}(\mathbf{f}_{a_2^{(N_{a_2})}}) \leq K \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots \\ \mathbf{t}(\mathbf{f}_{a_m^{(1)}}) + \mathbf{t}(\mathbf{f}_{a_m^{(2)}}) + \ldots + \mathbf{t}(\mathbf{f}_{a_m^{(N_{a_m})}}) \leq K \end{cases} \tag{39}$$

where $\mathbf{f}_{a_i^{(j)}}$ $(1 \leq j \leq N_{a_i})$ refers to the $j$th actions of agent $a_i$.

This is an NP-hard problem. Proof is by restriction. Assume that $m = 1$. The task is to find an effort allocation for agent $a_1$ such that $P[F]$ is maximized and

$$\mathbf{t}(\mathbf{f}_{a_1^{(1)}}) + \mathbf{t}(\mathbf{f}_{a_1^{(2)}}) + \ldots + \mathbf{t}(\mathbf{f}_{a_1^{(N_{a_1})}}) \leq K \quad . \tag{40}$$

Since this simplified problem ($m = 1$) is NP-hard (please refer to [16] for proofs), the multiagent object search problem is also NP-hard. Thus, it is impractical to design a team search strategy that can always generate an effort allocation that maximizes the probability of detecting the target. Thus, the goal becomes for each member of the team to cooperate with the others so as to find the target as early as possible.

## 6 Learning, Interaction and Planning

The multiagent object search task described in this paper is an ideal task to study the relationships between learning, interaction and planning in multiagent environments.

Ideally, an agent should plan its actions based on a perfect knowledge about the current situation of the world (such as the target distribution), the ability of itself and all the other agents, and the activities of the whole team performed so far. But because of limited computation power, limited memory, and limited

communication power, an agent is not able to keep track everything. For example, it is not possible for an agent to update its *own* knowledge base every time *other* agents execute actions that fail to detect the target, because the process of updating the probability distribution is time consuming. In the multiagent search team, the agent updates the probability distribution only after its *own* action is applied.

It is also not efficient if an agent performs the search task only based on its own knowledge without considering other agents actions. This may cause many extraneous activities. Thus, a certain amount of communication is essential for the team search task. The question is how much communication is needed and how much memory an agent must maintain to make the behavior of the whole team satisfactorily coordinated.

An agent's internal knowledge can be divided into two parts: local knowledge and global knowledge. The local knowledge is the agent's knowledge about itself and the influence of its own action on the world. The global knowledge is its knowledge about other agents and the effects of the actions of other agents on the world. Each agent should have a perfect local knowledge, but it is not able to have a perfect global knowledge because of various limitations. Usually the global knowledge is obtained by interaction, communication and learning from other agents during the search process. A search agent's planning system is influenced by both its local knowledge and the learned global knowledge.

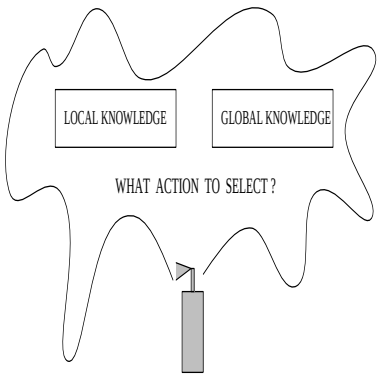


**Fig. 9.** A search agent's planning system is influenced by both its local knowledge and global knowledge.

### 6.1 Learning and Where to Look Next

The task of "where to look next" for an agent $a_i$ can be defined as: for a fixed position $(x_c, y_c, z_c)$, select the state parameters $(w, h, p, t, r)$ for the next action $\mathbf{f}$ of agent $a_i$ such that the probability of detecting the target is maximized.

**Local Knowledge.** If only local knowledge is considered, the task is the same as object search by a single agent, on which we have done extensive research. We have designed a strategy to decompose the huge space of possible sensing actions determined by different $(w, h)$, $p$, $t$, and $r$ into a finite set of actions that must be considered and a strategy to select the best action among the resulted actions from the decomposition such that

$$P(\mathbf{f}) = \sum_{c \in \Omega(\mathbf{f})} \mathbf{p}(a_i, c, \tau_{\mathbf{f}}) \mathbf{b}(a_i, c, \mathbf{f}) \tag{41}$$

is maximized [15].

In a multiagent environment, the probability $P(\mathbf{f})$ calculated in this way does not perfectly reflect the real probability of detecting the target by action $\mathbf{f}$, because the knowledge about the target distribution $\mathbf{p}(a_i, c, \tau_{\mathbf{f}})$ used in the calculation is agent $a_i$'s local knowledge, which does not reflect the influences of the actions of other agents (because agent $a_i$ is not able to update the probability distribution whenever other agents apply an action on the environment). So, even though $a_i$ thinks that the region within the effective volume $EV(\mathbf{f})$ of $\mathbf{f}$ has high probability, it might not be so, since other agents may have already examined this region extensively. From the global point of view, the probability within $EV(\mathbf{f})$ may not be so high as $a_i$ thinks at the moment $\tau_{\mathbf{f}}$.

Thus, $a_i$ needs to integrate the influences of other agents' applied actions into its action planning process.

**Learned Global Knowledge.** Although each agent $a_i$ is not able to update its target distribution whenever another agent executes an action, it is able to record *crude* information about to what extent cells of the region have been checked by other agents.

As discussed before, for a given action $\mathbf{f}$, only those cubes that are within $\mathbf{f}$'s effective volume $EV(\mathbf{f})$ can be checked with high confidence. Thus, whenever an action $\mathbf{f}$ is executed, only the examination situation of those cubes that are within $EV(\mathbf{f})$ need be recorded; the examination situation of other cubes need not be recorded. To do this, each agent $a_i$ maintains the Examination Situation Map $E(a, c)$. $E(a, c)$ gives the number of times that the cell $c$ falls into the effective volume of the actions executed by agent $a$. Each agent $a_i$ must maintain the Examination Situation Map for every other agent $a_j$ $(1 \leq j \leq m, j \neq i)$ of the search team and every cell $c$ $c \in \Omega$ of the search region $\Omega$ (Fig. 10).

For each agent, the Examination Situation Map is maintained through a process of learning by communication. During the search process, as soon as an agent $a_i$ selects an action $\mathbf{f}(a_i, s_{a_i}, r_{a_i}^{(j)})$ to execute, it broadcasts a list of all the cells that are within the effective volume $EV(\mathbf{f})$ of action $\mathbf{f}$ to all the other members of the team. Upon receiving the broadcast from $a_i$, each other team member $a_j$ $(1 \leq j \leq m,\ j \neq i)$ updates its Examination Situation Map as following:

$$\forall c \in EV(\mathbf{f}) \text{ perform } E(a_i, c) \leftarrow E(a_i, c) + 1 \ . \tag{42}$$

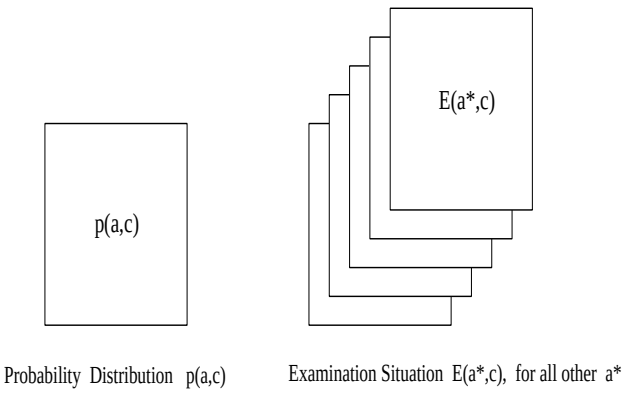


**Fig. 10.** An agent's local and global knowledge.

**The Influence of the Learned Global Knowledge on Planning.** The next action $\mathbf{f}$ is selected with reference to both local knowledge and external stimuli. The local knowledge is the probability of detecting the target $P(\mathbf{f})$ by action $\mathbf{f}$. The external stimuli $ES(\mathbf{f})$ is derived from the *learned global knowledge*, which is represented by the Examination Situation Map. The objective function for agent $a_i$ is: select an action $\mathbf{f}$ that maximizes the following weighted sum

$$w_1 P(\mathbf{f}) - w_2 ES(\mathbf{f}) \quad , \tag{43}$$

where $w_1$, $w_2$ are weights, and $ES(\mathbf{f})$ is defined as follows

$$ES(\mathbf{f}) = \sum_{c \in \Omega(\mathbf{f})} \sum_{j \neq i}^{m} E(a_j, c) \quad . \tag{44}$$

### 6.2 Learning, Coordination and Where to Move Next

The goal of "where to move next" for an agent $a_i$ in a multiagent object search team can be defined as: select the agent position $(x_c, y_c)$ for agent $a_i$ such that the chances of detecting the target for agent $a_i$ and the whole team is maximized.

**The Local Knowledge and the Learned Global Knowledge.** If only local knowledge is considered, then the task is the same as the "where to move next" task for a single agent. The strategy of selecting the next agent position is straight forward. For each candidate position $(x, y)$, there is a range of space that can be checked by the camera without occlusion. We call this range the sensed sphere $SS(x, y)$ for position $(x, y)$ (please refer to [15] [17] for more detail). The sum of the probability of all the cells within the corresponding sensed sphere is called the sensible probability for this position $S_{prob}(x, y)$. The task is to find a position $(x, y)$ such that the sensible probability is maximized.

As in the previous discussion, the local knowledge $S_{prob}(x, y)$ does not perfectly reflect the real sensible probability for the position $(x, y)$. The agent should also include the learned global knowledge $E_s(x, y)$ into its decision making. $E_s(x, y)$ gives the measurement of how much the region within the sensed sphere of position $(x, y)$ has been checked by other agents

$$E_S(x, y) = \sum_{c \in SS(x,y)} \sum_{j \neq i} E(a_j, c) \quad . \tag{45}$$

**Coordination.** For the "where to move next" task of the multiagent search team, coordination among different agents should also be taken into consideration. For example, if all the agents find the same region to have the highest sensible probability, then when coordination is not considered, all the agents will be gathered around this region to perform actions. This is not desirable for the team as a whole since other regions of the search space are not checked while many of the actions may be redundant. Coordination is needed when deciding the next positions of agents in a multiagent team. From the global point of view, the agents should be distributed evenly across the search space. Thus, a coordination factor $D(x, y)$ should be considered in planning the next action. $D(x, y)$ specifies how far the new position $(x, y)$ is to other agents of the team (Fig. 11).

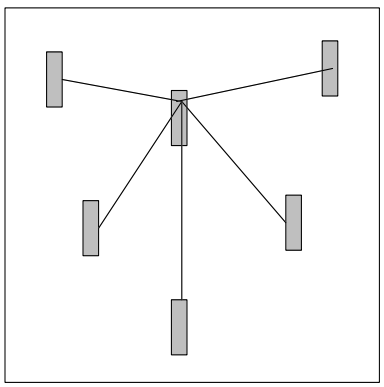

**Fig. 11.** The relative positions for different agents are also very important for efficient search. The positions of the agents in the team should be carefully selected such that the extraneous activities are avoided.

**Planning.** The next position $(x, y)$ for agent $a_i$ should be selected by integrating the local knowledge, the learned global knowledge, and the coordination factor together. The goal is to maximize the following weighted term

$$w_1 S_{prob}(x, y) - w_2 E_s(x, y) + w_3 D(x, y) \quad , \tag{46}$$

where $w_1$, $w_2$, and $w_3$ are weights, and

$$D(x,y) = \sum_{j=1,j\neq i}^{m} \left[ w_{ij} \sqrt{(x-x_j)^2 + (y-y_j)^2} \right] \quad . \tag{47}$$

The value of $w_{ij}$ $(1 \leq j \leq m,\ j \neq i)$ is used to balance the importance of the distance between agent $a_j$ and agent $a_i$. These values must satisfy

$$\sum_{j=1,j\neq i}^{m} w_{ij} = 1 \quad . \tag{48}$$

If there is no preference, then we set

$$w_{12} = \ldots = w_{1m} = \frac{1}{m-1} \quad . \tag{49}$$

## 7 Conclusion

In this paper, we formulate the multiagent object search task and prove that this task is NP-Complete from a global point of view. We analyze various issues that involve learning in a multiagent object search system, including: (A) the local knowledge updating rule for an agent; (B) a method to obtain the initial common target distribution; (C) learning of the global knowledge through interaction and communication with other agents; (D) learning and interaction so as to improve the "where to look next" task; (E) learning and coordination so as to improve the "where to move next" task.

In future work we will endeavor to develop a more detailed interaction and learning theory for the multiagent object search task. Given the highly cooperative nature of this task, we believe that a further study of this problem will reveal deeper relationships between learning, interaction and organizations in multiagent environments.

## References


1. Hayes-Roth, B., Brownston, L., Gen, R., V.: Multiagent collaboration in directed improvisation. International Conference on Multi-Agent Systems, U.S.A. (1995)
2. Huber, H., Durfee: On acting together: without communication. AAAI Spring Symposium on Reasoning about Mental States, U.S.A. (1995)
3. Ingrand, F., F., Georgeff, M., P., Rao, A., S.: An architecture for real-time reasoning and system control. IEEE EXPERT. **6** (1992)
4. Jennings, N.: Controlling cooperative problem solving in industrial multi-agent systems using joint intentions. Artificial Intelligence, **75** (1995)
5. Kinny, D., Ljungberg, M., Rao, A., Sonenberg, E., Tidhard, G., Werner, E.: Planned team activity. Artificial Social Systems, Lecture notes in AI 830, C. Castelfranchi and E. Werner, editors. Springer Verlag, New York, U.S.A. (1992)

6. Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., Osawa, E.: The robot world cup initiative. IJCAI-95 Workshop on Entertainment and AI/Alife, Montreal, Canada (1995)
7. Kuniyoshi, Y., Rougeaux., S., Ishii, M., Kita, N., Sakane, S., Kakikura, M.: Cooperation by observation: the framework and the basic task pattern. IEEE International Conference on Robotics and Automation (1994)
8. Levesque, H., J., Cohen, P., R., Nunes, J.: On acting together. National Conference on Artificial Intelligence, Calif, U.S.A. (1990)
9. Nickerson, S. B., Jenkin, M., Milios, E., Down, B., Jasiobedzki, P., Jepson, A., Terzopoulos, D., Tsotsos, J., Wilkes, D., Bains, N., Tran, K.: ARK: autonomous navigation of a mobile robot in a known environment. Intelligent Autonomous Systems-3, Pennsylvania, USA. (1993) 288–296
10. Pimentel, K., Teixeira, K.: Virtual reality: through the new looking glass. Windcrest/McGraw-Hill, Blue Ridge Summit (1994)
11. Rao, A., S., Lucas, A., Morley, D., Selvestrel, M., Murray, G.: Agent-oriented architecture for air-combat simulation. Technical Note 42, The Australian Artificial Intelligence Institute (1993)
12. Schubert, J.: Cluster-based specification techniques in Dempster-Shafer theory for an evidential intelligence analysis of multiple target tracks. Ph.D Thesis, Department of Numerical Analysis and Computing Science, Royal Institute of Technology, S-100 44 Stockholm, Sweden (1994)
13. Tambe, M., Rosenbloom, P., S.: RESC: An approach for real-time, dynamic agent tracking. Proceedings of the International Joint Conference on Artificial Intelligence, Montreal Canada (1995)
14. Weiss G.: Learning to Coordinate Actions in Multi-Agent Systems. Proceedings of the International Joint Conference on Artificial Intelligence, France (1993) 311–316
15. Ye, Y., Tsotsos, J.: Where to look next in 3D object search. 1995 IEEE International Symposium for Computer Vision, Florida, U.S.A. (1995) 539–544
16. Ye, Y., Tsotsos, J.: Sensor planning in 3D object search: its formulation and complexity. The 4th International Symposium on Artificial Intelligence and Mathematics, Florida, U.S.A. (1996)
17. Ye, Y., Tsotsos, J.: Sensor Planning for Object Search. Technical Report, RBCV-TR-94-47, Computer Science Department, University of Toronto. (1994)

This article was processed using the LaTeX macro package with LLNCS style