FL seminar 2008-12-15

Cooperative Optimal Search Control with Distributed Control

FL08-21-2

Mamoru Saito

Search Control Problem

Search Problem

To locate a target to be found, deploying agents with the available resources.

- search and rescue operations
- detecting lost objects

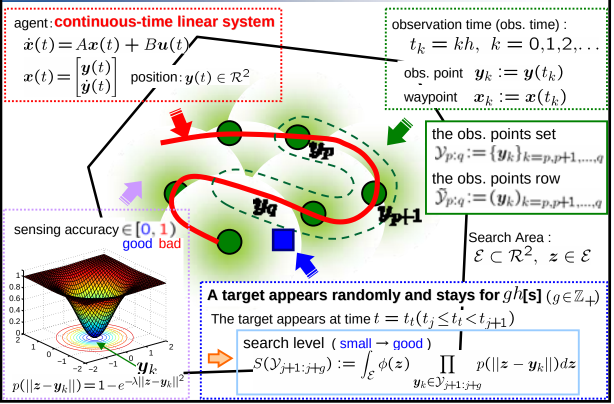
Objective

- formulate the Optimal Search Control Problem
  - maximize the probability of finding the target
  - reduce the control energy consumption
- analyze the agent's behavior

Outline

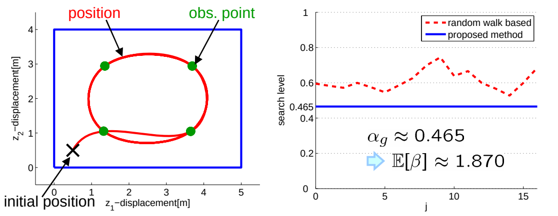
- review (single agent)
- Distributed Cooperative Optimal Search Control
- おまけ

Problem Setting



Main Result

- formulate Optimal Search Control Problem
  - semi-optimization problem ⇨ approximate solution
- the analysis of the agent's behavior
  - the expectation value of how many times the targets appears without being found
  - a necessary and sufficient condition for the agent's state and input to converge respectively to a periodic trajectory



Cooperative Search Control

Cooperative Optimal Search Control

- Centralized Control
- Decentralized Control
- Distributed Control

n_a : the number of the agent

The superscript (l) means the state of the l-th agent.

Problem Setting

Each agent is assumed to receive
the observation point information from its neighbors.

the neighborhood of the l-th agent at time t_k, $k = 1, 2, \ldots$

$$\mathcal{N}_k^{(l)} := \left\{ l' \middle| \|\boldsymbol{y}_k^{(l)} - \boldsymbol{y}_k^{(l')}\| \le D \right\}, \quad l = 1, 2, \ldots, n_a$$

$$l' \in \mathcal{N}_k^{(l)}$$

$$\Updownarrow$$

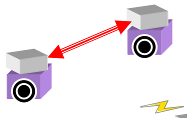
The l-th agent receives the observation point information from the l'-th agent at time t_k.

Observation Point Information

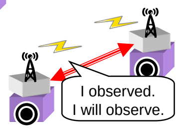
the observation point information: $\mathcal{B}_k^{(l)}$

⇨ the own and the other agents' information at time t_k

- $\mathcal{B}_k^{(l)} := \bigcup_{l' \in \mathcal{N}_k^{(l)}} \{\boldsymbol{y}_k^{(l')}\}$
  (only use visual information)



- $\mathcal{B}_j^{(l)} = \mathcal{B}_j^{(l)} \cup \left(\bigcup_{l' \in \mathcal{N}_k^{(l)}} \{\boldsymbol{y}_j^{(l')}\} \right)$
  $j = 1, 2, \ldots, k+g-1$
  (communicate each other)



the observation point information set from t_P to t_Q

$$\mathcal{B}_{P:Q}^{(l)} := \bigcup_{k=P}^{Q} \mathcal{B}_k^{(l)}$$

Objective

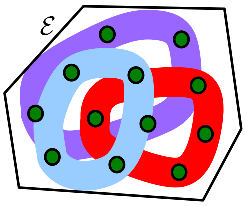
Each agent is assumed to receive
the observation point information from its neighbors.

Cooperative Semi-optimal Search Control Algorithm

to minimize $S\left(\bigcup_{l'=1}^{n_a} \mathcal{Y}_{k+1:k+g}^{(l')} \right)$

as $k \to \infty$

while reducing the
control energy consumption



Semi-optimal Search

the gradient method

$$\boldsymbol{y}_{k+g}^{(l)} = \boldsymbol{y}_k^{(l)} + \alpha_k \frac{\partial S\left(\bigcup_{l'=1}^{n_a} \mathcal{Y}_{k-g+1:k}^{(l')} \right)}{\partial \boldsymbol{y}_k^{(l)}}$$

can't be used

⇨ minimize $S\left(\bigcup_{l'=1}^{n_a} \mathcal{Y}_{k+1:k+g}^{(l')} \right)$ as $k \to \infty$

⇩

the gradient method

$$\boldsymbol{y}_{k+g}^{(l)} = \boldsymbol{y}_k^{(l)} + \alpha_k \frac{\partial S\left(\mathcal{B}_{k-g+1:k}^{(l)} \right)}{\partial \boldsymbol{y}_k^{(l)}}$$

semi-optimal solution!

⇨ minimize $S\left(\mathcal{B}_{k+1:k+g}^{(l)} \right)$ as $k \to \infty$

Semi-optimal Search Algorithm

Algorithm 5 *Semi-optimal Search Algorithm (Distributed Control)*

$k \leftarrow 0$
while 1 — renew the observation points
  if $k = 0$
    $\mathcal{Y}_{1:g}^{(l)} = y\_int\_random(g)$
  else
    Compute $\boldsymbol{y}_{k+g}^{(l)}$ from $\boldsymbol{y}_i^{(l)} = \boldsymbol{y}_k^{(l)} + \alpha_k \frac{\partial S\left(\mathcal{B}_{k-g+1:k}^{(l)}\right)}{\partial \boldsymbol{y}_k^{(l)}}$
  end if
  $k \leftarrow k + 1$
end while

Theorem

- $\mathcal{B}_k^{(l)} := \bigcup_{l' \in \mathcal{N}_k^{(l)}} \{\boldsymbol{y}_k^{(l')}\}$

  k' satisfying $\mathcal{N}_k^{(l)} = \{1, 2, \ldots, n_a\}, \; k = k'+1, k'+2, \ldots$ exists.

- $\mathcal{B}_j^{(l)} = \mathcal{B}_j^{(l)} \cup \left(\bigcup_{l' \in \mathcal{N}_k^{(l)}} \{\boldsymbol{y}_j^{(l')}\} \right) \quad j = 1, 2, \ldots, k+g-1$

  k' satisfying $\bigcup_{j=k-g+1}^{k} \mathcal{N}_j^{(l)} = \{1, 2, \ldots, n_a\}, \; k = k'+1, k'+2, \ldots$ exists.

⇨ The proposed algorithm

minimizes $S\left(\bigcup_{l'=1}^{n_a} \mathcal{Y}_{k+1:k+g}^{(l')} \right)$ as $k \to \infty$

(proof : omitted)

Semi-optimal Search Control Algorithm

Algorithm 6 *Semi-optimal Search Control Algorithm (Distributed Control)*

$k \leftarrow 0$
while 1
  if $k = 0$
    $\mathcal{Y}^{(l)}_{1:g} = y\_int\_random(g)$
  else
    Compute $\boldsymbol{y}^{(l)}_i$, $i = k+g, k+2g, \ldots$ from $\boldsymbol{y}^{(l)}_i = \boldsymbol{y}^{(l)}_k + \alpha_k \dfrac{\partial S\left(\mathcal{B}^{(l)}_{k-g+1:k}\right)}{\partial \boldsymbol{y}^{(l)}_k}$
  end if
  if $mod(k, g) = 0$
    Compute $\hat{\mathcal{Y}}^{(l)}_{k+1:k+f}$ by solving [Problem E]
  end if
  Compute $\boldsymbol{u}^{(l)}(t), t \in [t_k, t_{k+f}]$ from $\hat{\mathcal{Y}}^{(l)}_{k+1:k+f}$,(11),(12)
  Input $\boldsymbol{u}^{(l)}(t), t \in [t_k, t_{k+1})$
  $k \leftarrow k+1$
end while

renew the observation points

renew the observation point row

minimize $J^*_{k:k+f}(\hat{\mathcal{Y}}^{(l)}_{k+1:k+f})$
s.t. $\hat{\mathcal{Y}}^{(l)}_{k+1:k+g} \in sort(\mathcal{Y}^{(l)}_{k+1:k+g})$
$\boldsymbol{y}^{(l)}_i = \boldsymbol{y}^{(l)}_{i-g}$
$i = k+g, k+1+g, \ldots, k+f$

RHC

Remark

Remark 1

The proposed algorithm is robust (adaptive)
for changing the number of agents (due to, e.g., failures).

Remark 2

The initial observation points is important
because ***the periodic trajectory largely depends on them***.

Simulation

system

$$\dot{x}(t) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \boldsymbol{u}(t)$$

$$\boldsymbol{x}^{(1)}_0 = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 \end{bmatrix}^T$$
$$\boldsymbol{x}^{(2)}_0 = \begin{bmatrix} 4.5 & 1 & 0 & 0 \end{bmatrix}^T$$
$$\boldsymbol{x}^{(3)}_0 = \begin{bmatrix} 1 & 3.5 & 0 & 0 \end{bmatrix}^T$$

$h = 1$

$\mathcal{E} = [0, 5] \times [0, 4]$

$p(||\boldsymbol{z} - \boldsymbol{y}_k||) = 1 - e^{-\lambda||\boldsymbol{z} - \boldsymbol{y}_k||^2}, \ \lambda > 0$

$\phi(\boldsymbol{z}) = \frac{1}{20}$

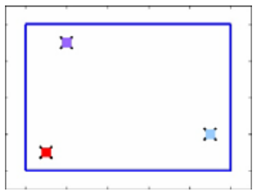
$R = diag(1, 1)$

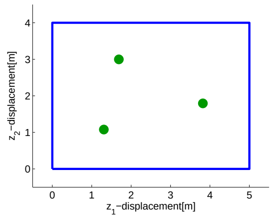
Simulation (g=1, D=8)

ex1. $\lambda = 0.5$
$f = 5$

$$\mathcal{B}^{(l)}_k = \bigcup_{l' \in \mathcal{N}^{(l)}_k} \{\boldsymbol{y}^{(l')}_k\} = \bigcup_{l=1}^{n_a} \{\boldsymbol{y}^{(l)}_k\}$$

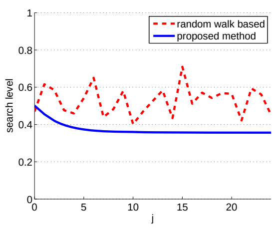
25 steps

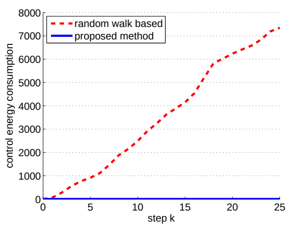




proposed method

Simulation (g=1, D=8)





search level $S(\mathcal{Y}_{j+1:j+g})$

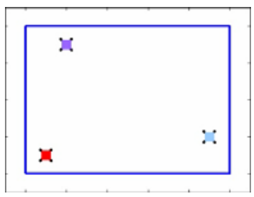
control energy consumption

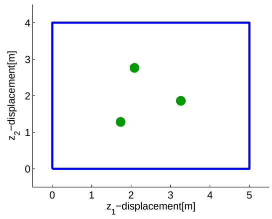
Simulation (g=1, D=1.5)

ex2. $\lambda = 0.5$
$f = 5$

$$\mathcal{B}^{(l)}_k := \bigcup_{l' \in \mathcal{N}^{(l)}_k} \{\boldsymbol{y}^{(l')}_k\}$$

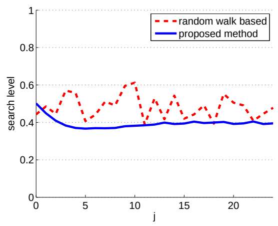
25 steps



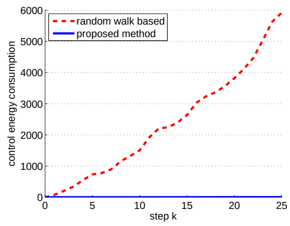


proposed method

Simulation (g=1, D=1.5)



search level $S(\mathcal{Y}_{j+1:j+g})$

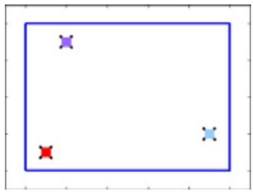


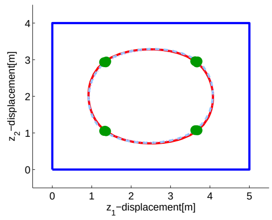
control energy consumption

Simulation (g=4, D=0)

ex3. $\lambda = 1$ $f = 10$ $\mathcal{B}_k^{(l)} = \bigcup_{l' \in \mathcal{N}_k^{(l)}} \{\boldsymbol{y}_k^{(l')}\} = \{\boldsymbol{y}_k^{(l)}\}$

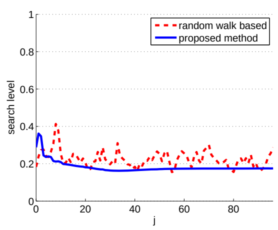
100 steps



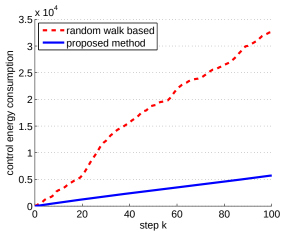


proposed method

Simulation (g=4, D=0)



search level $S(\mathcal{Y}_{j+1:j+g})$

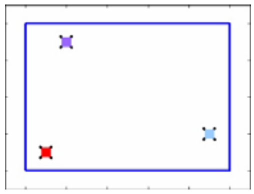


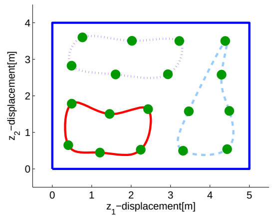
control energy consumption

Simulation (g=6, D=8)

ex4. $\lambda = 2$ $f = 10$ $\mathcal{B}_k^{(l)} = \bigcup_{l' \in \mathcal{N}_k^{(l)}} \{\boldsymbol{y}_k^{(l')}\} = \bigcup_{l=1}^{n_a} \{\boldsymbol{y}_k^{(l)}\}$

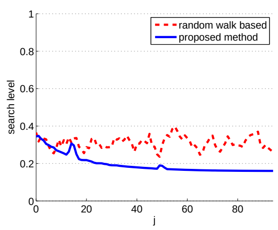
100 steps



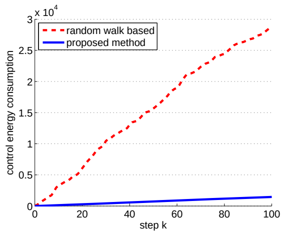


proposed method

Simulation (g=6, D=8)



search level $S(\mathcal{Y}_{j+1:j+g})$

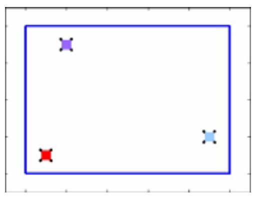


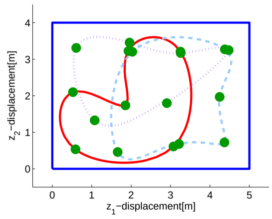
control energy consumption

Simulation (g=6, D=2)

ex5. $\lambda = 2$ $f = 10$ $\mathcal{B}_k^{(l)} := \bigcup_{l' \in \mathcal{N}_k^{(l)}} \{\boldsymbol{y}_k^{(l')}\}$

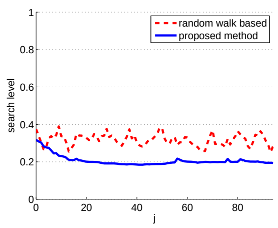
100 steps

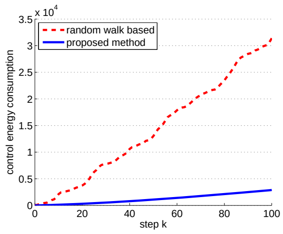




proposed method

Simulation (g=6, D=2)





search level $S(\mathcal{Y}_{j+1:j+g})$

control energy consumption

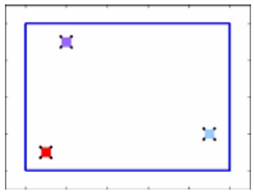
Simulation (g=6, D=2)

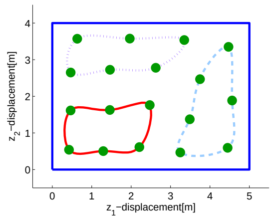
ex6. $\lambda = 2$
$f = 10$

100 steps

$$\mathcal{B}_j^{(l)} = \mathcal{B}_j^{(l)} \cup \left(\bigcup_{l' \in \mathcal{N}^{(l)}} \{ \boldsymbol{y}_j^{(l')} \} \right)$$

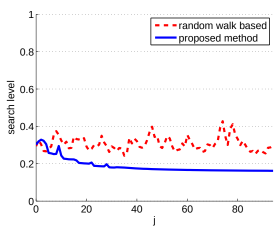
$$j = 1, 2, \dots, k+g-1$$

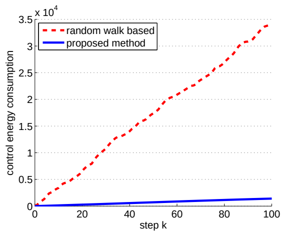




proposed method

Simulation (g=6, D=2)





search level $S(\mathcal{Y}_{j+1:j+g})$

control energy consumption

Conclusion

Conclusion

Distributed Cooperative Optimal Search Control Algorithm was proposed.