Distributed Control of Autonomous Mobile Robots: Covering an Area

FL07-07-1
David Asikin

Outline

- Introduction
- Llyod's Algorithm in 1D
- Llyod's Algorithm in 2D
- Block Diagram Representation
- Convergence from Lyapunov Theory
- Conclusion
- Future Works

Introduction

- The goal:
  All robots, placed initially at random, spread out as much as possible (efficiently) over a given area.
- Example:
  Deploying a team of robots to act as guards, finding survivors on a disaster area, planet exploration, etc.

How? ⟶ Llyod's algorithm.

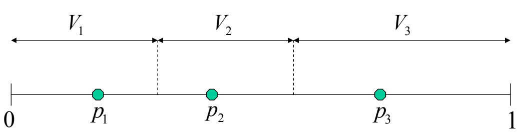
Llyod's Algorithm in 1D (1)

- $(n=3)$ Let $p_1 \le p_2 \le p_3$ be three arbitrary points in $[0,1]$.
- Construct the optimum partition $\{V_1, V_2, V_3\}$ regarding p1, p2, p3

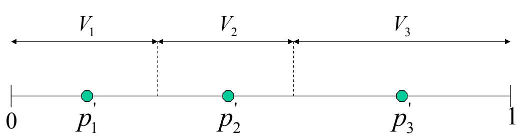
$V_1 = \left[0, \frac{p_1+p_2}{2}\right]$ $V_2 = \left[\frac{p_1+p_2}{2}, \frac{p_2+p_3}{2}\right]$ $V_3 = \left[\frac{p_2+p_3}{2}, 1\right]$

Llyod's Algorithm in 1D (2)

- The result:



- Next: Update p_i to be the midpoint of V_i .



Lloyd's Algorithm

- Step 0: Start with an arbitrary partition $\{W_i\}$ and arbitrary points $\{p_i\}, p_i \in W$
- Step 1: Construct the unique Voronoi partition $\{V_i\}$ generated by $\{p_i\}$
- Step 2: Update p_i to be the centroid of V_i
  Return to Step 1.

Llyod's Algorithm in 1D (3)

$p_1^{'} = \frac{p_1 + p_2}{4}$ $\quad p_2^{'} = \frac{p_1 + 2p_2 + p_3}{4}$ $\quad p_3^{'} = \frac{p_2 + p_3 + 2}{4}$

- Discrete time state equation:

$$p(k+1) = Ap(k) + b$$

$$\begin{bmatrix} p_1^{'} \\ p_2^{'} \\ p_3^{'} \end{bmatrix} = \frac{1}{4}\begin{bmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 1 \end{bmatrix}\begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1/2 \end{bmatrix}$$

- p will converge to: $(1/6, 1/2, 5/6)$

Llyod's Algorithm in 1D (4)

- Continuous time state equation:

$$\dot{p} = Ap + b$$

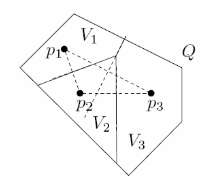
$\dot{p}_1 = p_1^{'} - p_1$ $\quad \dot{p}_2 = p_2^{'} - p_2$ $\quad \dot{p}_3 = p_3^{'} - p_3$

$$\begin{bmatrix} \dot{p}_1 \\ \dot{p}_2 \\ \dot{p}_3 \end{bmatrix} = \frac{1}{4}\begin{bmatrix} -3 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -3 \end{bmatrix}\begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1/2 \end{bmatrix}$$

- p will converge to: $(1/6, 1/2, 5/6)$

Llyod's Algorithm in 2D (1)

- Voronoi Partitions:
  The set of all points *q* whose distance from p_i is less than or equal to the distances from all other p_j

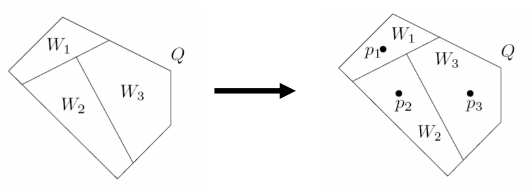


$$V_i = \{q : (\forall j \neq i) \|q - p_i\| \leq \|q - p_j\|\}$$

Point:
Voronoi partition is uniquely determined by p

Llyod's Algorithm in 2D (2)

- Fixed Partition:
  Place a sensor at a location *p* in a given partition *W* to optimize coverage.



Llyod's Algorithm in 2D (3)

- How to evaluate optimization?
  By using cost function:

$$H(p,W) = \int_W f(\|q-p\|)\phi(q)dq$$

$\phi(q)$: density function

$f(\|q-p\|)$: sensing perfomance

f = big → poor sensing

Point: $\min H(p,W)$ will optimize coverage

Llyod's Algorithm in 2D (4)

$$H(p,W) = \int_W f(\|q-p\|)\phi(q)dq$$

↓

$$f(\|q-p\|) = \|q-p\|^2$$
$$\phi(q) = 1$$

↓

$$H(p,W) = \int_W \|q-p\|^2 dq$$

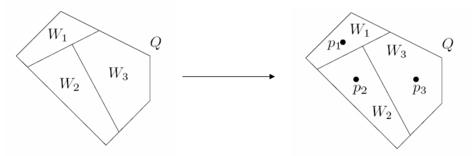
↓

$$H(p,W) = H(c_W, W) + A_W\|p - c_W\|^2$$

Conclusion: $\min H(p,W) \longrightarrow p = c_W$

Llyod's Algorithm in 2D (5)

- Objective: to optimize sensors' coverage



$$H(p,W)=H(p_1,W_1)+\ldots+H(p_n,W_n)$$

$$\min H(p,W)=\min H(p_1,W_1)+\ldots+\min H(p_n,W_n)$$

$$p_1=c_{W_1} \quad \ldots\ldots\ldots\ldots \quad p_n=c_{W_n}$$

$$p_2=c_{W_2}$$

Robot Coverage

- Consider: *n* mobile robots moving in the plane. The goal: robots deploy themselves and provide adequate coverage of a given convex polytope.
- Take the usual kinematic model:

$$\dot{p}_i=u_i$$

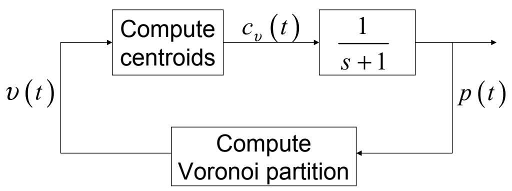
- Based on Llyod's algorithm (=robot should head for the centroid of its Voronoi):

$$\dot{p}_i=u_i \xrightarrow{u_i=c_i-p_i} \dot{p}=c_v-p$$

Block Diagram (Continuous time)

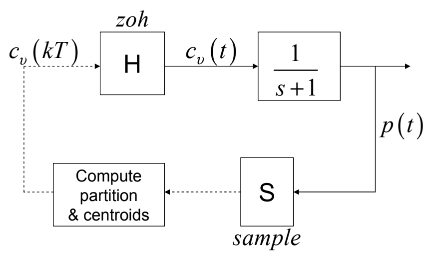
$v(t)$ = Voronoi partition generated by the points $p(t)=(p_1(t),\ldots,p_n(t))$.

$c_i(t)$ = centroid of $V_i(t)$ and define the vector $c_v(t)=(c_1(t),\ldots,c_n(t))$.



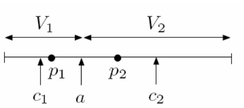
Block Diagram (Discrete time)

- For real-time, a sampled-data implementation is required.



Convergence – Lyapunov Theory (1)

- 2 robots $p_1 \le p_2$ arbitrarily placed in $[0,1]$:



- Step 1: $V_1=[0,a]$ $\quad V_2=[a,1]$ $\quad a=\frac{p_1+p_2}{2}$
- Step 2: $c_1=\frac{p_1+p_2}{4}$ $\quad c_2=\frac{p_1+p_2}{4}+\frac{1}{2}$
- Points evolve (cont. time) according to:

$$\dot{p}_1=c_1-p_1 \qquad \dot{p}_2=c_2-p_2$$

Convergence – Lyapunov Theory (2)

- Cost function:

$$L(p_1,p_2)=\int_0^a(q-p_1)^2dq+\int_a^1(q-p_2)^2dq$$

$$\Downarrow$$

$$L(p_1,p_2)=\int_0^a(q-c_1)^2dq+\int_a^1(q-c_2)^2dq+a(p_1-c_1)^2+(1-a)(p_2-c_2)^2$$

- Lyapunov function:

$L(x_1,x_2)>0$

for $\forall x$, $\dot{L}(x_1,x_2)<0$

Convergence – Lyapunov Theory (3)

- To use the function as a Lyapunov function, we need to study:

$$\frac{d}{dt}L\left(p_1(t),p_2(t)\right)=\sum_{i=1,2}\frac{\partial L}{\partial p_i}\left(p_1(t),p_2(t)\right)\dot{p}_i(t)$$

- By doing this, we get:

$$\frac{d}{dt}L=2a(p_1-c_1)\dot{p}_1+2(1-a)(p_2-c_2)\dot{p}_2$$
$$=-2a(p_1-c_1)^2-2(1-a)(p_2-c_2)^2\leq 0$$

- **Conclusion**: the robots converge to a local optimum of L where $p_i = c_i$

Conclusion

- Llyod algorithm gives the optimum coverage for robots.
- From the viewpoint of Lyapunov theory, Llyod algorithm’s convergence gives a stable result.

Future Works

- Llyod algorithm simulation in 1D and 2D.
- Read more papers regarding coverage control.
- etc.

Any question?