The Monitoring of Cloud Movements with Robotic Camera Networks

Akihiro Sugiyama
FL11-23-1

24th, January, 2012

Introduction

Photovoltaic (PV) power generation [1]

The installation of PV generation systems is rapidly growing due to concerns related to environment.

Merit

- A clean and environmentally-friend source of energy

Demerit

- Depending on weather condition and cloud-cover
- Low conversion efficiency

Monitoring [2]

Satellite and ground-based sky imaging tools can be used to monitor and forecast cloud movements.

Forecast of PV energy production is available in advance.

Introduction

Monitoring cloud movements

Robotic camera networks

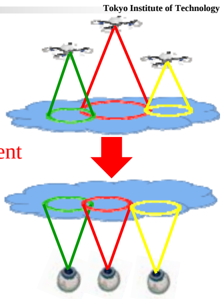
- Control for a group of aerial robots [3]

  Positioning and orienting multiple robotic cameras to collectively monitor environment

  Maximize aerial robots coverage

- The learning algorithm for potential game

  Payoff-based Inhomogeneous Partially Irrational Play (PIPIP) [4]



All-sky imaging [5]

- The system includes a camera and wide-angle lens or hemispheric mirror.
- Analysis of image is based on the red-to-blue ratio.

Control for a group of aerial robots [3]

Camera networks

- Surveillance
- Tracking
- Catching speeding drivers
- Gathering scientific data

Approach

Minimum information per pixel

Get a cost function that represents how well a group of cameras covers an environment

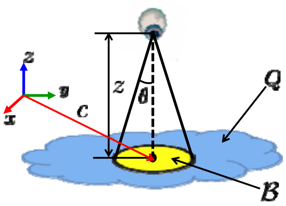
Obtain a control law by taking the negative gradient of the cost function

The controller is adaptive to the deletion or addition of cameras and to a changing environment

---

Control for a group of aerial robots [3]

Optimal camera placement



The state of camera i $\quad p_i \in \mathcal{P}$

$$i = 1, 2, \cdots, n$$

$$p = [c^T, z]^T \quad c \in \mathbb{R}^2, \; z \in \mathbb{R}$$

Bounded environment $\quad Q \supset \mathbb{R}^2$

The field of view $\mathcal{B}$

$$\mathcal{B} = \{q \mid \frac{\|q-c\|}{z} \le \tan\theta\}$$

We want to control n cameras such that their placement minimizes the information per camera pixel over the environment.

$$\min_{(p_1,\cdots,p_n)\in\mathcal{P}^n} \int_Q \frac{\text{info}}{\text{pixel}} dq$$

Control for a group of aerial robots [3]

Single camera

$$\min_p \int_Q f(p,q)\phi(q)dq$$

$\frac{\text{area}}{\text{pixel}}$ $\quad \frac{\text{info}}{\text{area}}$

The area per pixel $\quad f(p,q)$

Derived from the optics of the camera and geometry of the environment

Inside $\mathcal{B}$

$f(p, q)$ is equal to the inverse of the area magnification factor (defined from classical optics) times the area of one pixel.

$$f(p,q) = \frac{(b-z)^2}{b^2} \times (\text{the area of one pixel})$$

$$a = \frac{(\text{the area of one pixel})}{b^2}$$

b : The focal length

Outside $\mathcal{B}$

There are no pixel.

$$f(p,q) = \begin{cases} a(b-z)^2 & \text{for } q \in \mathcal{B} \\ \infty & \text{otherwise} \end{cases}$$

The information per area $\quad \phi(q)$

A positive weighting of importance over environment

Specified beforehand

Require more resolution, the value of $\phi(q)$ should be large.

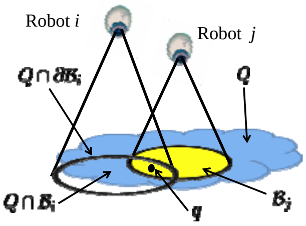
Control for a group of aerial robots [3]

Multiple cameras

An area that is observed by two cameras is better covered than by one camera.

It is not twice as well covered

The area per pixel at q



$$\frac{\text{area}}{\text{pixel}} = \left(\sum_{i=1}^{n} f(p_i, q)^{-1} \right)^{-1}$$

It is not use the area per pixel ,but the pixel per area.

We assume that the robots have knowledge of the geometry of Q , and some notion of information content over it $\phi(q)$.

$$\frac{\text{area}}{\text{pixel}} = \left(\sum_{i=1}^{n} f(p_i, q)^{-1} + w^{-1} \right)^{-1}$$

A prior area per pixel $w \in (0, \infty)$

Arbitrarily vague

$$h_{N_q}(p_1, \cdots, p_n, q) = \left(\sum_{i \in N_q} f(p_i, q)^{-1} + w^{-1} \right)^{-1} \qquad N_q = \{ i \mid q \in B_i \}$$

Control for a group of aerial robots [3]

Multiple cameras

The cost function

$$\mathcal{H}(p_1, \cdots, p_n) = \int_Q h_{N_q}(p_1, \cdots, p_n, q) \phi(q) dq$$

The cost function is valid for any area per pixel function $f(p_i, q)$, and for any camera state space $\mathcal{P}$.

The multiple camera optimization problem

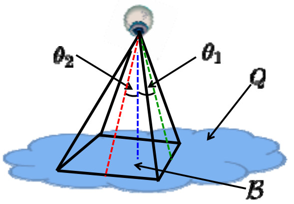
$$\min_{(p_1, \cdots, p_n) \in \mathcal{P}^n} \mathcal{H}$$

Notice that $\mathcal{H} > 0$ for all $(p_1, \cdots, p_n)$

Control for a group of aerial robots [3]

Rectangular field of view

Actual cameras have a rectangular change coupled device (CCD) array , and therefore a rectangular field of view.



The state of camera i $\quad p_i = [c_i^T \ z_i \ \psi_i]^T$

ψ_i : The yaw angle

The rotation matrix

$$R(\psi_i) = \begin{bmatrix} \cos\psi_i & \sin\psi_i \\ -\sin\psi_i & \cos\psi_i \end{bmatrix}$$

The field of view B

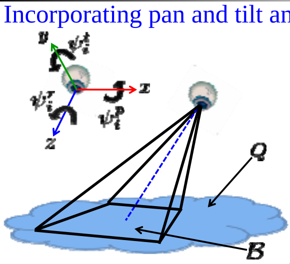
$$B_i = \{ q \mid |R(\psi_i)(q - c_i)| \le z_i \tan\theta \}$$

$$\theta = [\theta_1 \ \theta_2]^T$$

The cost function $\mathcal{H}(p_1, \cdots, p_n)$ is the same as for the circular case, as is the area per pixel function $f(p_i, q)$.

Control for a group of aerial robots [3]

Incorporating pan and tilt angles



The rotation (or yaw) angle ψ_i^r

(+) The field of view spins clockwise

The pan (or roll) angle ψ_i^p

(+) The field of view sweeps to the left

The tilt (or pitch) angle ψ_i^t

(+) The field of view sweeps to upward

The state of ith camera

$$p_i = [x_i \ y_i \ z_i \ \psi_i^r \ \psi_i^p \ \psi_i^t]^T$$

The position of the focal point

$$\rho_i = [x_i \ y_i \ z_i]^T$$

Consider two coordinate frames

The camera fixed frame of ith robot (CF_i)

Fixed to the camera, centered at focal point

The global fixed frame (GF)

Centered at a fixed origin on the ground

Control for a group of aerial robots [3]

To express vectors in the CF_i

Rotation matrices

$$R_i^r = \begin{bmatrix} \cos\psi_i^r & \sin\psi_i^r & 0 \\ -\sin\psi_i^r & \cos\psi_i^r & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad R_i^p = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi_i^p & \sin\psi_i^p \\ 0 & -\sin\psi_i^p & \cos\psi_i^p \end{bmatrix}$$

$$R_i^t = \begin{bmatrix} \cos\psi_i^t & 0 & -\sin\psi_i^t \\ 0 & 1 & 0 \\ \sin\psi_i^t & 0 & \cos\psi_i^t \end{bmatrix}$$

To take a point x in the GF and express it in the CF_i

Translate the vector by ρ_i

→ Rotate the vector about z-axis by $\pi/2$

→ Rotate the vector about x-axis by π

→ Rotate the vector through ψ_i^r, ψ_i^p, and ψ_i^t

$$R^f = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

$$R_i(\psi_i^t, \psi_i^p, \psi_i^r)(x - \rho_i)$$

$$R_i(\psi_i^t, \psi_i^p, \psi_i^r) = R_i^t R_i^p R_i^r R^f$$

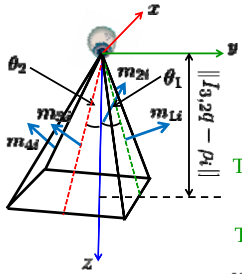
Control for a group of aerial robots [3]

The four outward facing unit normal vectors of faces of the pyramid

$$m_{1i} = [0 \ \cos\theta_1 \ -\sin\theta_1]^T \qquad m_{2i} = [\cos\theta_2 \ 0 \ -\sin\theta_2]^T$$

$$m_{3i} = [0 \ -\cos\theta_1 \ -\sin\theta_1]^T \qquad m_{4i} = [-\cos\theta_2 \ 0 \ -\sin\theta_2]^T$$

Denote them in CF_i frame



The vector from the focal point ρ_i to a point in the leg q is perpendicular to the normal of the kth pyramid face.

$$m_{ki}^T R_i (I_{3,2} q - \rho_i) = 0 \qquad I_{3,2} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

The field of view

$$B_i = \{ q \mid m_{ki}^T R_i (I_{3,2} q - \rho_i) \le 0, \ k = 1,2,3,4 \}$$

The area per pixel function

$$f(p_i, q) = \begin{cases} a(b - \|I_{3,2} q - \rho_i\|)^2 & \text{for } q \in B_i \\ \infty & \text{otherwise} \end{cases}$$

Application of PIPIP

$$\min_p \underbrace{\int_Q f(p,q)\phi(q)dq}_{=\mathcal{H}>0} \Rightarrow \max_p \underbrace{\left(-\int_Q f(p,q)\phi(q)dq\right)}_{=U_i}$$

$=U_i$ The utility function

Potential game

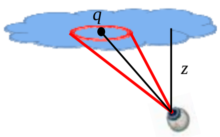
Actions

Pan and Tilt angles

$f(p,q)$: Derived from the optics of the camera

$$f(p_i,q)=\begin{cases} a(b-\|I_{3,2}q-p_i\|)^2 & \text{for } q\in B_i \\ \infty & \text{otherwise}\end{cases}$$

Given z, $f(p,q)$ is acquired



$\phi(q)$: The separation of cloudy from clear sky

Cloud : high

Clear sky : low

N. Gans et al [7]

Shannon's entropy[7, 8]

The entropy has found wide spread use in information, communication estimation theory

Maximum entropy can indicate an information rich data set

$$H(X)=-\sum_{x\in X} p(x)\log_2 p(x)$$

The entropy H

The pixel value X, x

Discrete random variable

The occurrence probability $p(x)$

The entropy gives a measurement of the uncertainty associated with X

The random variable x is uniformly distributed

→ The entropy takes its maximum

The random variable x is completely determined

→ The entropy takes its minimum

N. Gans et al [7]

Shannon's entropy [7, 8]

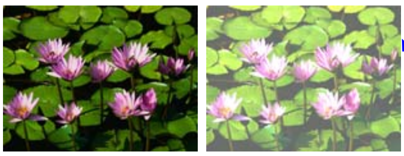
In the case of images

The image with a single shade or color has zero entropy

A blank image such as a camera facing a wall

The image with many shades and colors has high entropy

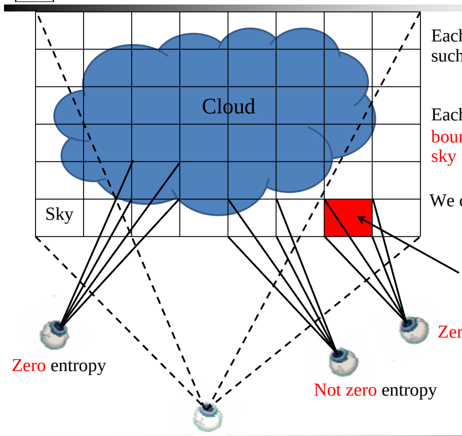
Example of similar images with different entropy levels



Poor lightning condition

Entropy High Low

N. Gans et al [7]



Each camera is controlled such that entropy is maximum

↓

Each camera monitors boundary between cloud and sky

↓

We can know the size of cloud

Application of PIPIP

Shannon's entropy

$$H(X)=-\sum_{x\in X} p(x)\log_2 p(x)$$

We want to maximize the entropy

$$\max\underbrace{\left(-\sum_{x\in X} p(x)\log_2 p(x)\right)}_{=U_i}$$

$=U_i$ The utility function

Potential game

Actions

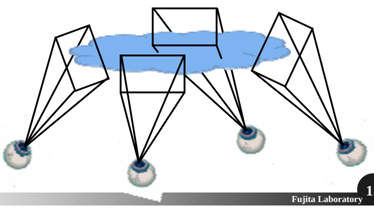
Pan and Tilt angles

$p(x)$: Acquired from the image

Image

0	0	0	0	0	0
0	2	2	1	1	0
0	2	4	4	3	0
0	2	5	2	3	0
0	1	1	1	3	0
0	0	0	0	0	0

Pixel value



Summary

- Formulate a cost function for single camera which has a circle field of view
- Formulate a cost function for multiple cameras which have a circle field of view
- Single camera which have a rectangular field of view
- Incorporating pan and tilt angles
- Study of Payoff based Inhomogeneous Partially Irrational Play (PIPIP)
- Study of the monitoring of cloud with Shannon's entropy

Future Works

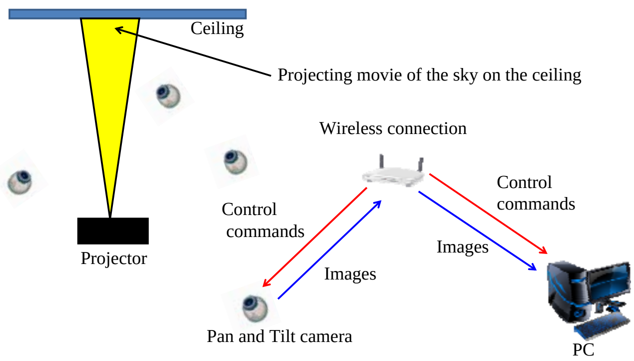
- The separation of cloudy from clear sky
  - Imaging processing
  - The threshold in the red-to-blue ratio [5]
- Modeling in the case of singe camera (six degrees of freedom)

$$[x_i, y_i, z_i, \psi_i^r, \psi_i^p, \psi_i^t]$$

- Modeling in the case of multiple cameras (six degrees of freedom)
- Compose experimental system

Future Works

Experimental system[6]



References

[1] T. T. N. Khatib, A. Mohamed and N. Amin, "A New Controller Scheme for Photovoltaics Power Generation Systems," *European Journal of Scientific Research ISSN*, Vol. 33, No. 3, pp.515-524, 2009.

[2] E. Bitar, P. P. Khargonekar and K. Poolla, "Systems and Control Opportunities in the Integration of Renewable Energy into the Smart Grid," *the 18th World Congress of the International Federation of Automatic Control*, Milano, Italy, Aug.-Sep., 2011.

[3] M. Schwager, B. J. Julian, M. Angermann and D. Rus, "Eys in the Sky: Decentralized Control for the Deployment of Robotic Camera Networks," *Proc. of the 2011 IEEE Multiconference on Systems and Control*, Denver, Colorado, USA, Sep., Vol. 99, No. 9, pp. 1541-1561, 2011.

[4] T. Goto, T. Hatanaka and M. Fujita, "Payoff-based Inhomogeneous Partially Irrational Play for Potential Game Theoretic Cooperative Control of Multi-agent Systems," (available at arXiv:1107.4838), 2011.

[5] G. Pfister, R. L. Mckenzie, J. B. Liley, A. Thomas, B. W. Forgan and C. N. Long, "Cloud Coverage Based on All-Sky Imaging and Its Impact on Surface Solar Irradiance," *Journal of the 2003 American Meteorological Society*, Vol. 42, Issue 10, pp. 1421-1434, 2003.

[6] T. Dinh, Q. Yu, G. Medioni, "Real Time Tracking using an Active Pan-Tilt-Zoom Network Camera," *the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, Missouri, USA, Oct., pp. 3786-3793, 2009.

References

[7] Y. Zhang, M. Rotea, N. Gans, "Sensors Searching for Interesting Things: Extremum Seeking Control on Entropy Maps," *the 50th IEEE Conference on Decision and Control and European Control Conference*, Orlando, Florida, USA, Dec. 14th, 2011.

[8] 奥富, "ディジタル画像処理," CG-ARTS協会, 2008.

Appendix

Control for a group of aerial robots [3]

Decentralize gradient based controller (Multiple cameras)

Theorem 1

- The lateral component

$$\frac{\partial \mathcal{H}}{\partial c_i} = \int_{Q \cap \partial \mathcal{B}_i} (h_{N_q} - h_{N_q \setminus \{i\}}) \frac{q - c_i}{\|q - c_i\|} \phi(q) dq$$

The integral component causes the robot to move to increase the amount of the environment in its field of view, while also moving away from other robots *j* whose field of view overlaps with its own.

- The vertical component

$$\frac{\partial \mathcal{H}}{\partial z_i} = \int_{Q \cap \partial \mathcal{B}_i} (h_{N_q} - h_{N_q \setminus \{i\}}) \phi(q) \tan\theta dq - \int_{Q \cap \mathcal{B}_i} \frac{2 h_{N_q}^2}{a (b - z_i)^3} \phi(q) dq$$

The first integral causes the robot to move up to bring more of the environment into its field of view.

The second integral causes it to move down to get a better look at the environment already in its field of view.

Control for a group of aerial robots [3]

Decentralize gradient based controller (Multiple cameras)

Use a gradient control law in which every robot follows the negative of its own gradient component

$$u_i = -k\frac{\partial \mathcal{H}}{\partial p_i}$$

The control input for robot i u_i

Control gain $k \in (0, \infty)$

Assuming integrator dynamics for the robots

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

Theorem 2

$$\lim_{t\to\infty} \frac{\partial \mathcal{H}}{\partial p_i} = 0 \quad \forall i \in \{1, \cdots, n\}$$

An equilibrium $(p_1^*, \cdots, p_n^*)$ defined by $\left.\frac{\partial \mathcal{H}}{\partial p_i}\right|_{p_i = p_i^*} = 0 \ \forall i \in \{1, \cdots, n\}$

is Lyapunov stable if and only if it is a local minimum of $\mathcal{H}$.

Control for a group of aerial robots [3]

The gradient function of the cost function (Rectangular field of view)

- The lateral component

$$\frac{\partial \mathcal{H}}{\partial c_i} = \sum_{k=1}^{4} \int_{Q \cap l_{ki}} (h_{N_q} - h_{N_q \setminus \{i\}}) R(\psi_i)^T n_{ki} \phi(q) dq$$

- The vertical component

$$\frac{\partial \mathcal{H}}{\partial z_i} = \sum_{k=1}^{4} \int_{Q \cap l_{ki}} (h_{N_q} - h_{N_q \setminus \{i\}}) \tan\theta^T n_{ki} \phi(q) dq - \int_{Q \cap B_i} \frac{2h_{N_q}^2}{a(b - z_i)^3} \phi(q) dq$$

- The angular component

The component rotates the robot to get more of its field of view into the environment, while also rotating away from other robots whose field of view intersects its own.

$$\frac{\partial \mathcal{H}}{\partial \psi_i} = \sum_{k=1}^{4} \int_{Q \cap l_{ki}} (h_{N_q} - h_{N_q \setminus \{i\}}) \cdot (q - c_i)^T R(\psi_i + \pi/2))^T n_{ki} \phi(q) dq$$

Control for a group of aerial robots [3]

The gradient function of the cost function (Incorporating pan and tilt angles)

$$\frac{\partial \mathcal{H}}{\partial p_i} = \sum_{k=1}^{4} \int_{Q \cap l_{ki}} (h_{N_q} - h_{N_q \setminus \{i\}}) \frac{R_i^T m_{ki}}{\| I_{2,3} R_i^T m_{ki} \|} \phi(q) dq + \int_{Q \cap B_i} \frac{2h_{N_q}^2}{a(b - \| I_{3,2} q - p_i \|)^3} \frac{(I_{3,2} q - p_i)}{\| I_{3,2} q - p_i \|} \phi(q) dq$$

$$\frac{\partial \mathcal{H}}{\partial \psi_i^s} = \sum_{k=1}^{4} \int_{Q \cap l_{ki}} (h_{N_q} - h_{N_q \setminus \{i\}}) \frac{m_{ki}^T \frac{\partial R_i}{\partial \psi_i^s} (p_i - I_{3,2} q)}{\| I_{2,3} R_i^T m_{ki} \|} \phi(q) dq \qquad s \in \{r, p, t\}$$

Where

$$\frac{\partial R_i}{\partial \psi_i^r} = R_i^t R_i^p \begin{bmatrix} -\sin\psi_i^r & \cos\psi_i^r & 0 \\ -\cos\psi_i^r & -\sin\psi_i^r & 0 \\ 0 & 0 & 0 \end{bmatrix} R^f \qquad \frac{\partial R_i}{\partial \psi_i^p} = R_i^t \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\sin\psi_i^p & \cos\psi_i^p \\ 0 & -\cos\psi_i^p & -\sin\psi_i^p \end{bmatrix} R_i^r R^f$$

$$\frac{\partial R_i}{\partial \psi_i^t} = \begin{bmatrix} -\sin\psi_i^t & 0 & -\cos\psi_i^t \\ 0 & 0 & 0 \\ \cos\psi_i^t & 0 & -\sin\psi_i^t \end{bmatrix} R_i^p R_i^r R^f$$

The learning algorithm for potential game [4]

Payoff based Inhomogeneous Partially Irrational Play (PIPIP)

Features

- Finite and a little memory
- Payoff-based
- PIPIP allows agents to make irrational decisions with a certain probability
- PIPIP assures that the actions of the group converge in probability to optimal Nash equilibria , though only convergence to pure Nash equilibria

⬇

Optimal Nash equilibria is the Nash equilibria maximizing the potential function.

The learning algorithm for potential game [4]

The set of agents $\mathcal{V} = \{1, \cdots, n\}$ The iteration $t = \{0, 1, 2, \cdots\}$

The collective action set $\mathcal{A}, a$

$\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$ $\qquad$ $\mathcal{A}_i$ $i \in \mathcal{V}$: The set of agent i's actions

$a = (a_i, a_{-i})$ $\qquad$ a_i : Agent i's action

$a_{-i} = (a_1, \cdots, a_{i-1}, a_{i+1}, \cdots, a_n)$

The set of actions $\mathcal{R}_i(a_i)$: The set of actions which agent i will be take in case he takes an action a_i

The utility function of agent i $U_i : \mathcal{A} \to \mathbb{R}$

The potential function $\phi : \mathcal{A} \to \mathbb{R}$

Constrained potential game $\Gamma = (\mathcal{V}, \mathcal{A}, \{U_i\}_{i\in\mathcal{V}}, \{\mathcal{R}_i\}_{i\in\mathcal{V}})$

The learning algorithm for potential game [4]

Definition (Constrained Potential games)

A constrained strategic game Γ is a constrained potential game if $\exists\phi \ \forall i \in \mathcal{V} \ \forall a_i \in \mathcal{A}_i \ \forall a_i' \in \mathcal{R}_i(a_i) \ \forall a_{-i} \in \Pi_{j\neq i}\mathcal{A}_j$

$$U_i(a_i', a_{-i}) - U_i(a_i, a_{-i}) = \phi(a_i', a_{-i}) - \phi(a_i, a_{-i})$$

If an changes his action,

(The change of the utility function)

= (The change of the potential function)

Definition (Constrained Nash Equilibria)

For constrained strategic game Γ, a collection of actions is a constrained pure Nash equillibrium if the following equation

$$U_i(a_i^*, a_{-i}^*) = \max_{a_i \in \mathcal{R}_i(a_i^*)} U_i(a_i, a_{-i}^*) \quad i \in \mathcal{V}$$

Any constrained potential game has at least one pure Nash equilibrium

A potential function maximizer is an optimal Nash equilibrium

There may exist undesirable pure Nash equilibria not maximizing the potential function

The learning algorithm for potential game [4]

Initialization Action *a* is chosen randomly from $\mathcal{A}$ $(a = \mathrm{rnd}(\mathcal{A}))$

$t = 0 : a_i(0)$ $U_i(0)$

$t = 1 : a_i(1) = a_i(0)$ $U_i(1) = U_i(0)$ $\forall i \in \mathcal{V}$

$\kappa \in \left(\frac{1}{C-1}, \frac{1}{2}\right]$

Step 1 Update $\varepsilon = t^{-\frac{1}{n(D+1)}}$ (Exploration rate) $C = \max_{i \in \mathcal{V}} \max_{a_i \in \mathcal{A}_i} |\mathcal{R}_i(a_i)|$

$D = \max_{i \in \mathcal{V}} D_i$ D_i: The minimal number of steps

Step 2 • $U_i(a(t-1)) \geq U_i(a(t-2))$

$$a_i^{tmp} \leftarrow \begin{cases} \mathrm{rnd}(\mathcal{R}_i(a_i(t-1)) \backslash \{a_i(t-1)\}) & \text{w.p. } \varepsilon \\ a_i(t-1) & \text{w.p. } 1-\varepsilon \end{cases}$$

• Otherwise $U_i(a(t-1)) < U_i(a(t-2))$

$$a_i^{tmp} \leftarrow \begin{cases} \mathrm{rnd}(\mathcal{R}_i(a_i(t-1)) \backslash \{a_i(t-1), a_i(t-2)\}) & \text{w.p. } \varepsilon \\ a_i(t-1) & \text{w.p. } (1-\varepsilon)(\kappa \cdot \varepsilon^{\Delta_i}) \\ a_i(t-2) & \text{w.p. } (1-\varepsilon)(1-\kappa \cdot \varepsilon^{\Delta_i}) \end{cases}$$

Step 3 Execute the action a_i^{tmp} and receive $U_i(a^{tmp})$

Step 4 $a_i(t-1) \leftarrow a_i^{tmp}$ $a_i(t-2) \leftarrow a_i(t-1)$ $U_i(a(t-1)) \leftarrow U_i(a^{tmp})$

$U_i(a(t-2)) \leftarrow U_i(a(t-1))$ $\Delta_i \leftarrow U_i(a(t-2)) - U_i(a(t-1))$

Step 5 $t \leftarrow t+1$ and go to Step1