به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

بهینه سازی و یادگیری توزیع شده

پروژه 2

نام و نام خانوادگی	نیما زمان پور
شماره دانشجویی	810198407
تاریخ ارسال گزارش	دی 1402

فهرست

شکل‌ها

مدل سازی

در این پروژه قصد داریم یک مسئله **consensus formation** را حل کنیم. در این مسئله ورودی هر عامل، شتاب در راستای **x,y** است و عامل ها باید با توافق بر سر یک نقطه، تشکیل یک شکل منظم دهند. همچنین در طول مسیر نباید نزدیک مانع‌ها شوند. همچنین نیروی ورودی باید بهینه باشد. پس داریم:

$$horizon: length\ of\ route$$

$$X_i = \left[x_i, y_i, V_{x_i}, V_{y_i}, a_{xi}, a_{yi}\right] \rightarrow (horizon * 6)$$

$$f(X,V) = \sum_{i=1}^{n_{agent}} f_{path_i}(X_i, V) + f_{cost_i}(X_i)$$

$$s.t.: x_{i,j+1} = x_{i,j} + v_{x(i,j+1)}dt$$
$$y_{i,j+1} = y_{i,j} + v_{y(i,j+1)}dt$$

$$v_{i,j+1} = v_{i,j} + a_{x(i,j+1)}dt$$

$$v_{i,j+1} = v_{i,j} + a_{x(i,j+1)}dt$$

$$\forall i, j, obstacle: \left(x_{i,j} - x_{obs}\right)^2 + \left(y_{i,j} - y_{obs}\right)^2 \geq r_{obs}^2$$

برای طبیعی تر کردن چرخش عامل ها، فاصله بین عامل ها و عامل اول یکسان است. ولی با جهت سرعت عامل اول می‌چرخد می‌کند. این چرخش بصورت یک ماتریس دوران در $keepDist[i]$ ضرب می‌شود.

$$\forall i \in Horizon : \theta_i = arctan\left(\frac{v_{y,i}}{v_{x,i}}\right)$$

$$keepDist[i][j] = initialKeepDist[i] * rotationMatrix(\theta - \frac{\pi}{2})$$

برای به توافق رسیدن بر سر نقطه اشتراک، از یک گراف برای اجماع بر سر مقادیر مشترک استفاده می‌کنیم. همانطور که در درس توضیح داده شد، گراف باید periodically Strongly Connected و نیز Double stochastic باشد. در حین حل مسئله یک بار بهینه سازی، و یک بار ترکیب اطلاعات انجام می‌شود.

$$A: strongly\ connected\ and\ doubly\ stochastic\ matrix$$

$$V: consensus\ data = AX[last\ position\ and\ speed\ variable]$$

حال هر کدام از توابع هدف را تعریف می‌کنیم:

$$f_{path}(X_i) = \sum 10 * (X[i][last\ position\ and\ speed\ variable] - V[i] - keep\ dist[i])^2$$

$$f_{cost}(X) = \left(\|a_x\|^2 + \|a_y\|^2\right)/10$$

$$f_{avoid\ obstacle} = \exp\left(-(x_{i,j} - x_{obs})^2 + (y_{i,j} - y_{obs})^2 - r_{obs}^2\right)$$

یک راه‌حل ساده برای از بین بردن قید غیر محدب دوری از موانع، تعریف کردن تابع هزینه نزدیکی به مانع است. در این تابع، بصورت نمایی با نزدیک شدن به مانع هزینه زیاد و با دور شدن از آن کم می‌شود. بدین ترتیب به راحتی موانع دافعه مورد نظر را تولید می‌کنند.

برای شهود بهتر نتایج، گیف نیز ضمیمه کد ها شده است.

روش Centralized

الف) توضیحات

در این روش هر عامل horizon*6 متغیر و horizon*4 قید (مربوط به معادله سرعت و شتاب) دارد.

لاگرانژین را می‌نویسیم:

$$L(X, \lambda, V) = f(X, V) + \mu D(X)$$

که در آن μ ضریب لاگرانژ قید تساوی است.

در حالت متمرکز هیچ تجزیه ای صورت نمی‌گیرد و تمامی متغیر ها و ضرایب لاگرانژ درون یک تابع لاگرانژ متشکل از جمع توابع هر کدام از عامل ها قرار گرفته و از این تابع نسبت به متغیر ها گرادیان گرفته می‌شود. الگوریتم حل 2-سطحی dual بصورت زیر است:

1. $Init\ X, \mu, k = 0$
2. $For\ 100\ times\ do:$
   a. $Fuse\ data\ V(k) = AX(k)$
   b. $For\ all\ agents\ do:$
   c. $For\ 50\ times\ do:$
      i. $X^{k+1} = X^k - \alpha * \frac{\partial}{\partial X} L(X^k, \mu^k, V^k)$
   d. $\mu^{k+1} = \mu^k + \alpha * \frac{\partial}{\partial \mu} L(X^k, \mu^k, V^l)$
3. $k = k + 1$

ب) شبیه سازی

مقادیر اولیه شبیه سازی:

$$N = 4, dt = 1\,, \rho = 2$$

$$[x_0, y_0, vx_0, vy_0] = \begin{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -5 \\ -5 \end{bmatrix} & \begin{bmatrix} 20 \\ 0 \\ 0 \\ 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 20 \\ -5 \\ 0 \end{bmatrix} & \begin{bmatrix} 20 \\ 20 \\ 0 \\ 0 \end{bmatrix} \end{bmatrix}$$

$$obstacle\ position\ and\ radius = \begin{bmatrix} \begin{bmatrix} 3 \\ 15 \\ 3 \end{bmatrix} & \begin{bmatrix} 9 \\ 6 \\ 3 \end{bmatrix} \end{bmatrix}$$

$$keep\ dist\ = \begin{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} & \begin{bmatrix} -1 \\ -1 \end{bmatrix} & \begin{bmatrix} 1 \\ -1 \end{bmatrix} & \begin{bmatrix} 0 \\ -2 \end{bmatrix} \end{bmatrix}$$

$$A = \frac{I_{n*n}}{2} + 1 * \frac{1}{2n}$$

ج) نتایج

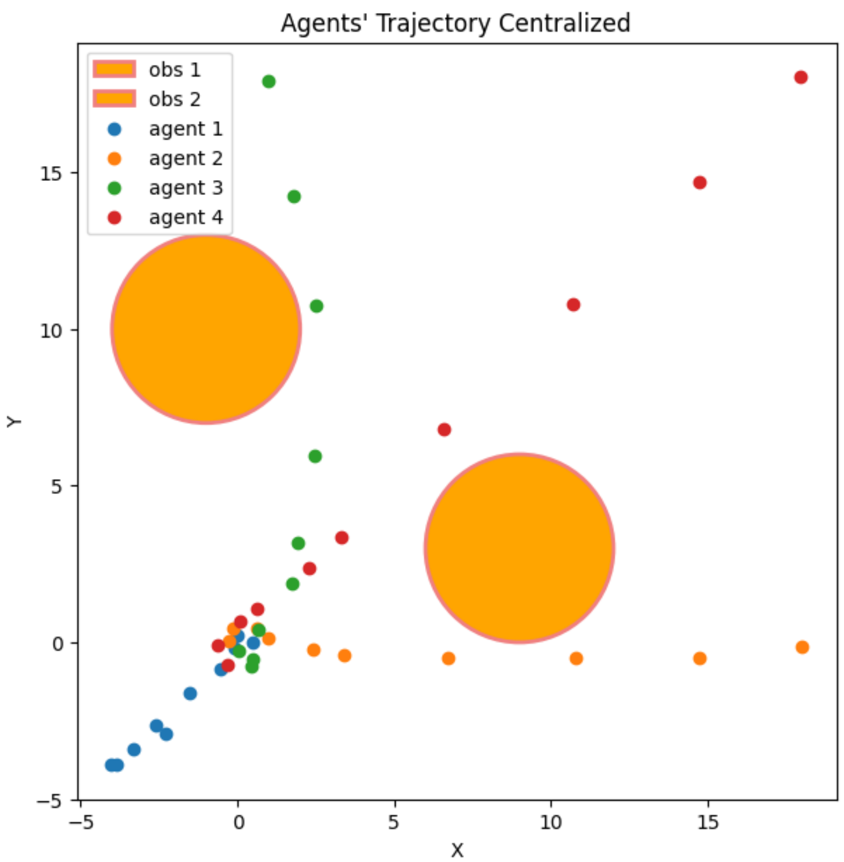

Figure 1 مسیر حرکت عامل ها

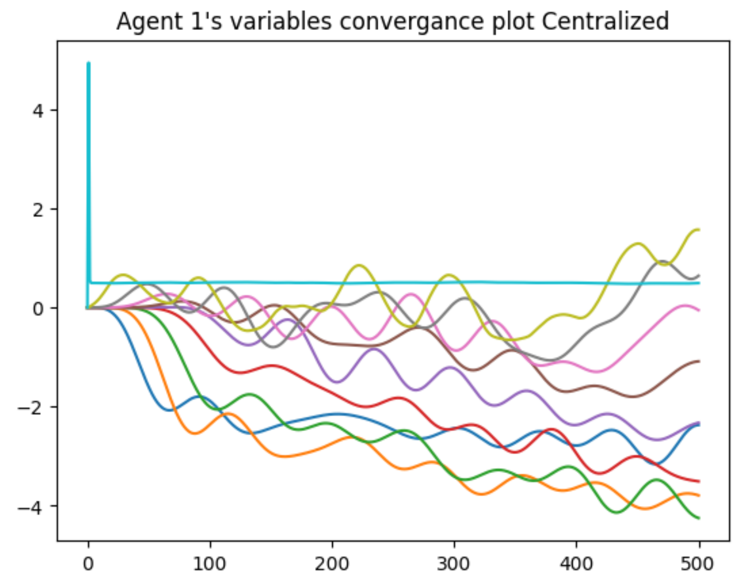

Figure 2 نمودار همگرایی بخشی از متغیر های عامل اول

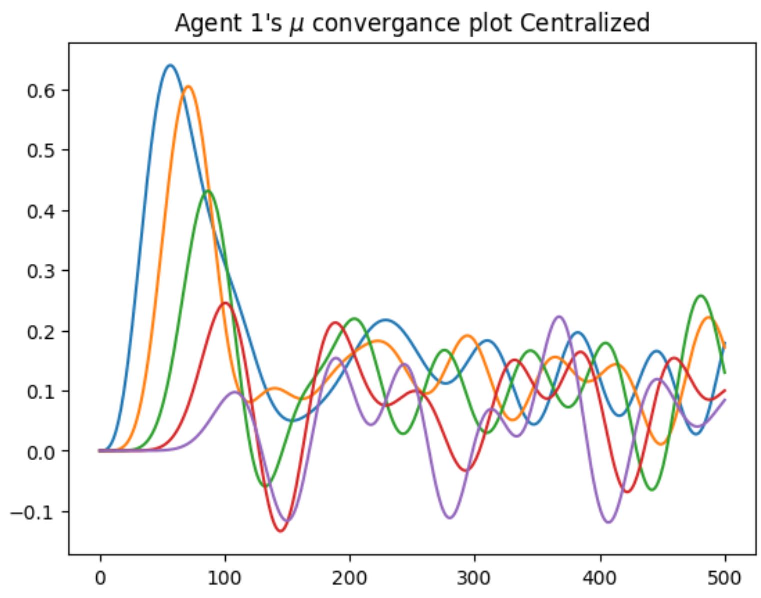

Figure 3 نمودار همگرایی بخشی از ضرایب لاگرانژ عامل اول

با نگاهی به نمودار های این روش، می‌تواند دید که عامل ها بطور قابل قبولی با دوری از موانع به اجماع رسیده‌اند اما جهت حرکت منظمی ندارند. بطور کلی این روش نیاز به زمان بیشتری برای محاسبات دارد. (بدلیل داشتن $400 = 4 * (20 + 40 + 40)$ متغیر) نمودار های متغیر ها و ضرایب لاگرانژ نیز این مسئله را تایید می‌کنند. در کل این روش 55 دقیقه به طول انجامید.

همچنین با دقت به گیف تولید شده می‌توان دریافت که عامل ها در زمان اجماع در حالت سکون هستند در حالی که در روش های دیگر در حال حرکت می‌باشند. این موضوع می‌تواند به دلیل متمرکز گرادیان گرفتن از توابع هدف باشد که موجب بهینگی بیشتر عامل ها می‌شود.

روش Distributed ADMM

الف) توضیحات

در این روش بر خلاف روش قبلی آپدیت بصورت ترتیبی در یک سطح انجام می‌شود. همچنین یک عبارت درجه 2 از قید ها به لاگرانژین افزوده می‌شود. که به آن Augmented Lagrangian می‌گویند. روش بدست آوردن و تجزیه کردن دقیقا همانند روش قبل است.

$$AL_i(X_i, \mu_i, \lambda_i) = L_i(X_i, \mu_i, \lambda_i) + \frac{\rho}{2}\left(\left|\left|D_i(X_i)\right|\right|^2\right)$$

پس روش آپدیت بصورت زیر است:

1. $Initialize\ X, \mu\ , k = 0$
2. $For\ 3000\ times\ do:$
   a. $fuse\ data: V(k) = AX(k)$
   b. $for\ all\ agents\ do:$
   c. $X_i^{k+1} = X_i^k - \alpha * \frac{\partial}{\partial X_i} AL_i(X, \mu^k, V^l)$
   d. $\mu_i^{k+1} = \mu_i^k + \alpha * \rho * \frac{\partial}{\partial \mu_i} L_i(X^k, \mu^k, V^k)$
3. $k\ =\ k + 1$

ب) شبیه سازی

مقادیر اولیه شبیه سازی:

$$N = 4, dt = 1\ , \rho = 2$$

$$[x_0, y_0, vx_0, vy_0] = \begin{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -5 \\ -5 \end{bmatrix} & \begin{bmatrix} 20 \\ 0 \\ 0 \\ 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 20 \\ -5 \\ 0 \end{bmatrix} & \begin{bmatrix} 20 \\ 20 \\ 0 \\ 0 \end{bmatrix} \end{bmatrix}$$

$$obstacle\ position\ and\ radius = \begin{bmatrix} \begin{bmatrix} 3 \\ 15 \\ 3 \end{bmatrix} & \begin{bmatrix} 9 \\ 6 \\ 3 \end{bmatrix} \end{bmatrix}$$

$$keep\ dist\ = \begin{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} & \begin{bmatrix} -1 \\ -1 \end{bmatrix} & \begin{bmatrix} 1 \\ -1 \end{bmatrix} & \begin{bmatrix} 0 \\ -2 \end{bmatrix} \end{bmatrix}$$

$$A = \frac{I_{n*n}}{2} + 1 * \frac{1}{2n}$$

ج) نتایج

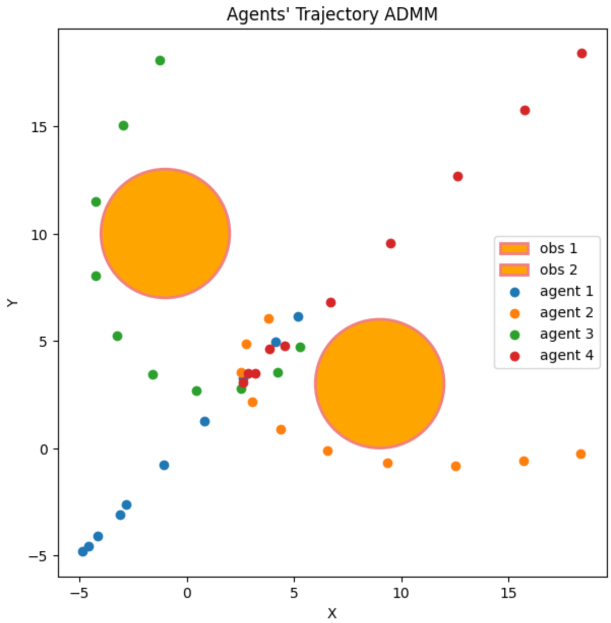

Figure 4 مسیر حرکت عامل ها

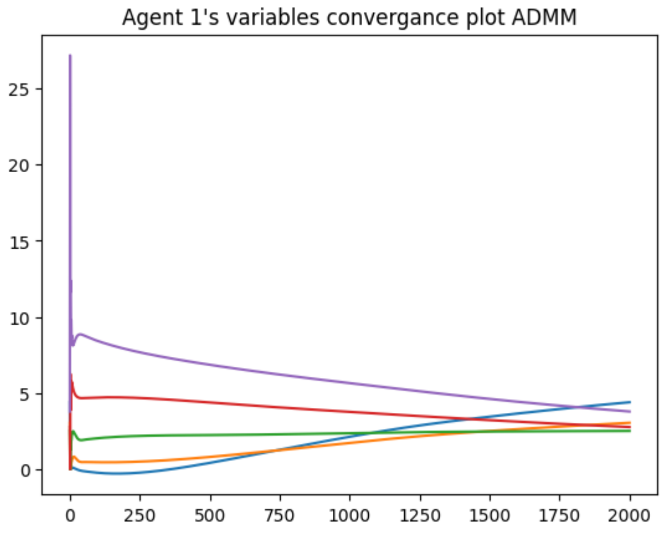

Figure 5 نمودار همگرایی بخشی از متغیر های عامل اول

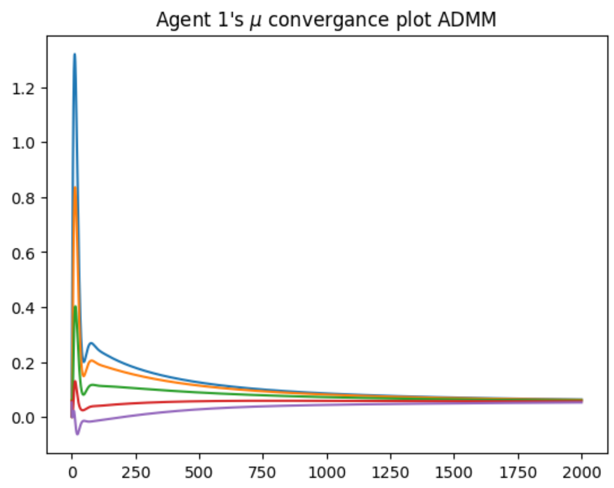

Figure 6 نمودار همگرایی بخشی از ضرایب لاگرانژ عامل اول

با نگاهی به نمودار ها می‌توان دید که عامل ها به سمت هم حرکت کرده و با دوری از موانع در نهایت به نقطه ای که برای همه آن ها زحمت کنترلی کمتری دارد به اجماع رسیده‌اند. همچنین متغیر ها و ضرایب لاگرانژ نیز به خوبی میل کرده اند. در کل روش ADMM بسیار خوب عمل کرده و در مدت زمان کوتاه 3 دقیقه به اجماع رسیده است.

روش Penalty Dual

الف) توضیحات

این روش نیز تقریبا مشابه روش dual است. با این تفاوت که قیود ناتساوی یکسو سازی می‌شوند و قیود تساوی قدر مطلق گرفته می‌شوند. پس لاگرانژین جدید بصورت:

$$L_i(X_i,\mu_i,\lambda_i) = \mathrm{F_i}(\mathrm{X_i},\mathrm{V}) + \mu_i|\mathrm{D}(\mathrm{X_i})| + \lambda_i \max(0, g(X_i))$$

است. از آنجایی که توابع هدف و قید ها جدا از هم هستند. بقیه روش مشابه روش centralized می‌شود.

پس روش آپدیت بصورت زیر است:

1. $Init\ X,\mu,k=0$
2. $For\ 100\ times\ do:$
   a. $Fuse\ data\ V(k) = AX(k)$
   b. $For\ all\ agents\ do:$
   c. $For\ 50\ times\ do:$
      i. $X_i^{k+1} = X_i^k - \alpha * \frac{\partial}{\partial X_i} L_i(X,\mu^k,V^k)$
   d. $\mu_i^{k+1} = \mu_i^k + \alpha * \frac{\partial}{\partial \mu_i} L_i(X^k,\mu,V^k)$
3. $k = k+1$

ب) شبیه سازی

مقادیر اولیه شبیه سازی:

$$N = 4, dt = 1\,, \rho = 2$$

$$[x_0, y_0, vx_0, vy_0] = \begin{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -5 \\ -5 \end{bmatrix} & \begin{bmatrix} 20 \\ 0 \\ 0 \\ 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 20 \\ -5 \\ 0 \end{bmatrix} & \begin{bmatrix} 20 \\ 20 \\ 0 \\ 0 \end{bmatrix} \end{bmatrix}$$

$$obstacle\ position\ and\ radius = \begin{bmatrix} \begin{bmatrix} 3 \\ 15 \\ 3 \end{bmatrix} & \begin{bmatrix} 9 \\ 6 \\ 3 \end{bmatrix} \end{bmatrix}$$

$$keep\ dist\ = \begin{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} & \begin{bmatrix} -1 \\ -1 \end{bmatrix} & \begin{bmatrix} 1 \\ -1 \end{bmatrix} & \begin{bmatrix} 0 \\ -2 \end{bmatrix} \end{bmatrix}$$

ج) نتایج

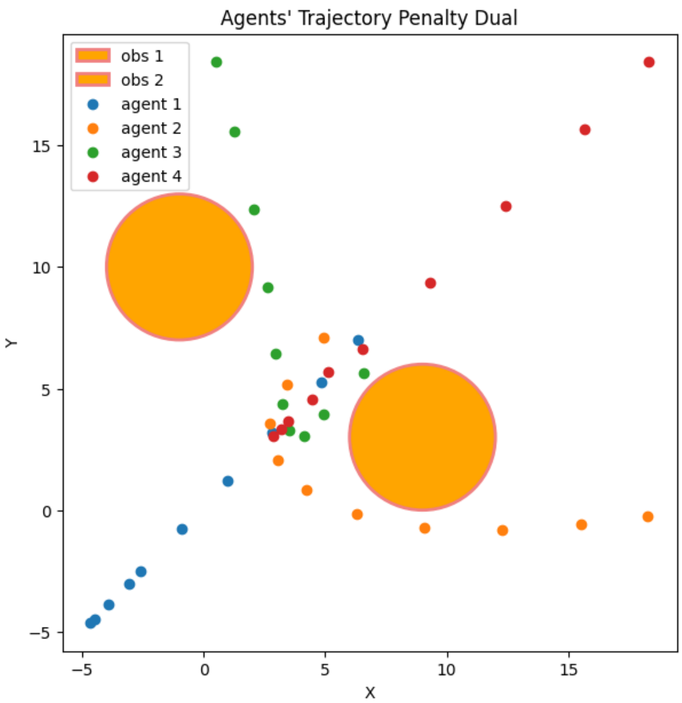

Figure 7 مسیر حرکت عامل ها

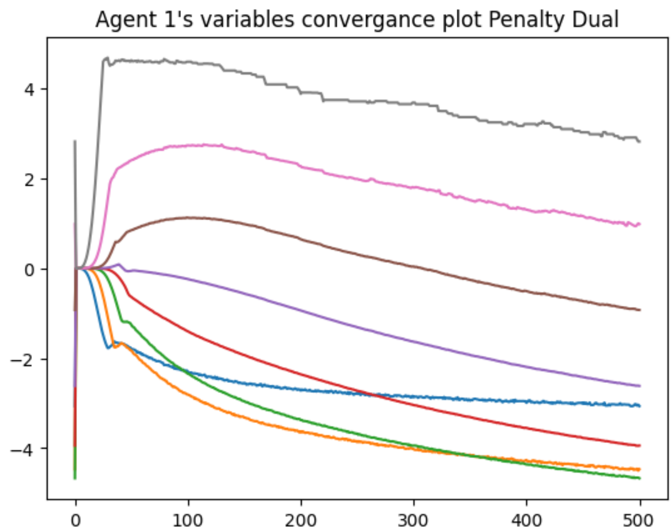

Figure 8 نمودار همگرایی بخشی از متغیر های عامل اول

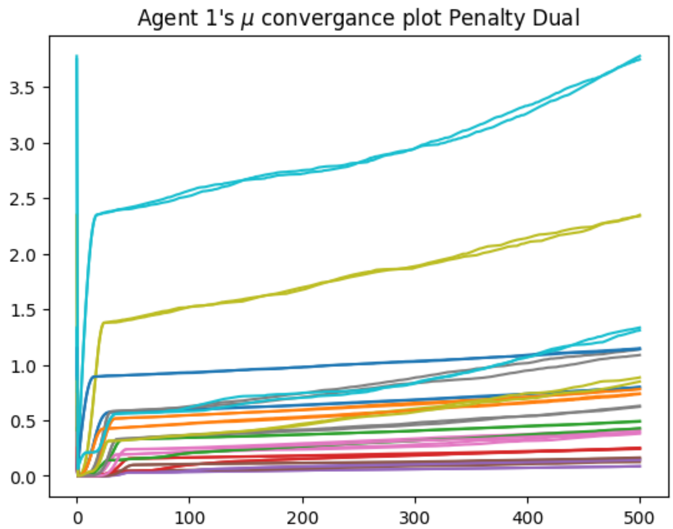

Figure 9 نمودار همگرایی بخشی از ضرایب لاگرانژ عامل اول

با نگاهی به نمودار های این روش، می‌تواند دید که عامل ها به خوبی با دوری از موانع به اجماع رسیده اند اما برعکس روش ADMM عامل سوم از بین مانع ها به نقطه اشتراک رسیده است. بطور کلی این روش با اینکه به همگرایی رسیده اما با نگاه به نمودار های متغیر ها و ضرایب لاگرانژ می‌توان دریافت که نیاز به زمان بیشتری برای محاسبات دارد. تا این نمودار ها نیز به خوبی میل کنند. در کل این روش 25 دقیقه به طول انجامید.

پس در کل بهترین عملکرد را روش ADMM با اختلاف زیادی نسبت به بقیه روش ها داشته است.

بررسی اثر تغییر ماتریس وزن A

در این قسمت به بررسی اثر وزن دهی ماتریس A بر روی شکل اجماع می‌پردازیم. برای مقایسه پذیری، همه تغییرات روی روش distributed ADMM انجام می‌شود.

الف)ماتریس $A: a_{ij} = \frac{1}{n}$:

در این ماتریس برخلاف ماتریس اصلی که وزن زیادی به متغیر عامل اول شخص می‌دهد، وزن ها یکنواخت توزیع شده اند.

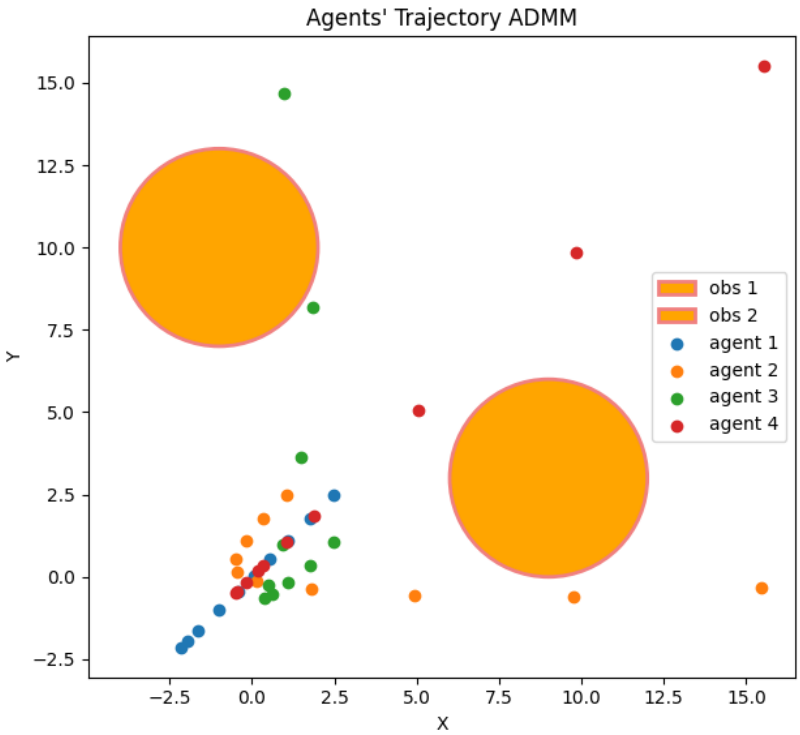

Figure 10 مسیر حرکت عامل ها

در این روش می‌بنیم که عامل ها در نزدیکی عامل 1 اجماع کرده اند. این مسئله را می‌توان اینطور توجیه کرد که عامل 1 دارای سرعت اولیه به سمت پایین چپ است و عامل ها با داشتن وزن های کمتر خود محور، به این مسئله توجه بیشتر کرده و محل اجماع بیشتر نزدیک عامل 1 شد.

ب) ماتریس تصادفی

$$A = col([\,0.0, 0.9, 0.1, 0.0])$$

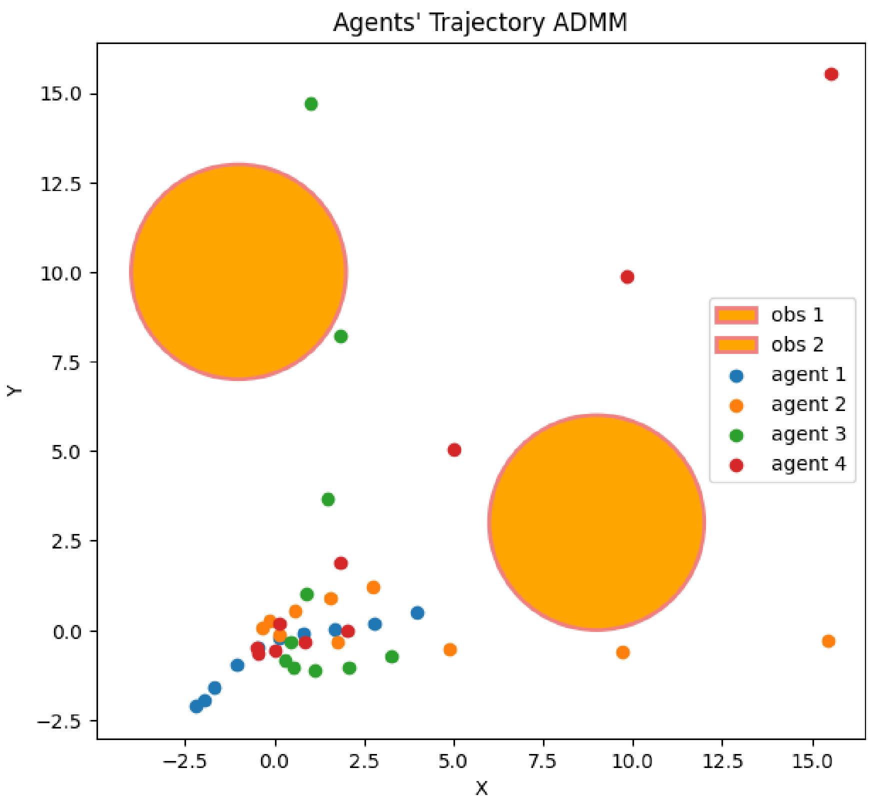

Figure 11 مسیر حرکت عامل ها

به دلیل وزن زیاد عامل دوم، با نگاه به گیف ایجاد شده می‌توان دید که محل نهایی اجماع کمی نزدیک به عامل دوم است.

ج) ماتریس A: Strongly Periodically connected

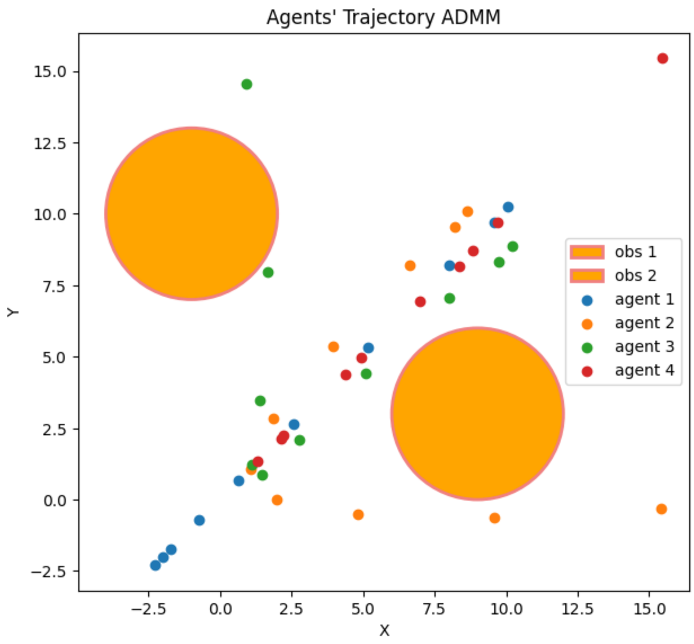

Figure 12 مسیر حرکت عامل ها

در این روش از ماتریس A در بخش های اول استفاده می‌کنیم. اما هر 100 بار اطلاعات ترکیب می‌شوند. با مشاهده گیف ایجاد شده می‌توان دریافت که این عدم اتصال در مدت طولانی خود را در غیر بهینه کردن مسیر عامل ها نشان می‌دهد. بطوری که عامل ها زمان زیادی را در مسیر رساندن متغیر های خود به محل قدیمی صرف می‌کنند اما محل اجماع به یکباره بعد از 100 بار گرادیان گیری عوض می‌شود. پس عامل ها اجبارا مسیری را اضافه می‌روند.

بطور کلی به این دلیل که در طول بهینه سازی، حدود 1000 بار میانگین گرفته می‌شود. محل های نهایی اجماع نزدیک به هم می‌شود. و با وجود تفاوت زیاد در ماتریس A تفاوت زیادی در محل اجماع دیده نمی‌شود. اما با periodically SC شدن ماتریس A عامل ها کمی غیر بهینه عمل می‌کنند.