اختصاصی مهندسی کنترل:
Telegram: @controlengineers
Web: controlengineers.ir

به نام خدا

کنترل تحمل پذیر خطا در سیستم های چند عامله

مجتبی نادری سورکی

کاری از مجموعه "اختصاصی مهندسی کنترل"

Telegram: @controlengineers
Web: controlengineers.ir

فهرست

- مقدمه
- روش های FDI
- روش های بازیابی خطا (Fault Recovery Methods)
- FDIR در سیستم های چند عامله
- مثال ها

اختصاصی مهندسی کنترل
controlengineers
@controlengineers

سیستم های چند عاملی

یک سیستم چند عامله (MAS) یک پیاده سازی با تاکید بر همکاری بین عامل هایی است که با یکدیگر برای رسیدن به همگرایی روی وظایفی مشخص هماهنگ شده اند

کاربرد ها

- کاربردهای نظارت بر محیط زیست
- حفظ آرایش‌بندی برای دفاع
- عملیات جستجو و نجات،
- عملیات اکتشاف معدن
- جاسوسی و شناسایی در کاربرهای نظامی،
- نقشه برداری محیط‌های خطرناک ویا ناشناخته، اکتشافات سیاره‌ای،
- جابه‌جایی اشیای سنگین،

مثال های عامل ها: وسایل نقلیه هوایی (UAV)، زمینی (UGV) یا زیر دریایی (UUV)

اختصاصی مهندسی کنترل
controlengineers
@controlengineers

سیستم های چند عاملی

وظایفی هست که یک عامل به تنهایی و به دلیل محدودیت های زیر نمیتواند انجام دهد:

- محدودیت در سنسور ها
- محدودیت عملگرها
- هزینه

همکاری به عنوان یک تیم:

- افزایش عملکرد
- افزایش قابلیت تطابق، پایداری و انعطاف بیشتر
- افزایش مقاوم بودن سیستم و مقابله با خطا
- هزینه کمتر

انواع مدل عامل ها

یک انتگرالی $\dot{x} = u$

دو انتگرالی $\ddot{x} = u$

دینامیک خطی

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

x: حالت های سیستم

u: ورودی عامل

y: خروجی عامل

انواع مدل عامل ها

دینامیک غیر خطی

$$M(x)\ddot{x} + C(x,\dot{x})\dot{x} + g(x) = u$$

$M(x)$ ← ماتریس اینرسي و متقارن و مثبت معین

$C(x,\dot{x})$ ← ماتریس کوریولیس و ترم هاي جانب مرکز

$g(x)$ ← بردار گرانش

تئوری گراف

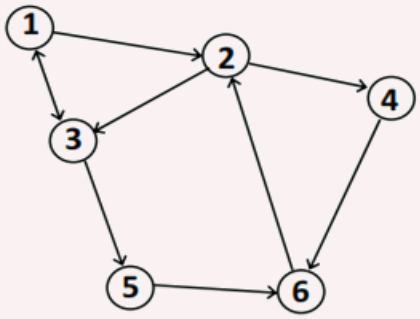
ارتباط بین عامل‌ها در یک سیستم چند عامله، با یک گراف وزن دار مشخص می‌شود.

گراف جهت‌دار G، عبارت است از یک مجموعه گره $v(G) = \{v_1, v_2, \dots v_N\}$ و یک مجموعه یال $\mathcal{E}(G) \subseteq \{(v_i, v_j) : v_i, v_j \epsilon\, v(G)\}$ که یال، در واقع یک جفت مرتب از گره‌های $v(G)$ است و نیز یک ماتریس مجاورت وزن $W = [w_{ij}] \epsilon R^{N \times N}$ با عناصر وزن $w_{ij} \geq 0$ که در آن $w_{ij} > 0$ است، اگر و تنها اگر $(v_i, v_j) \epsilon \mathcal{E}(G)$ باشد.

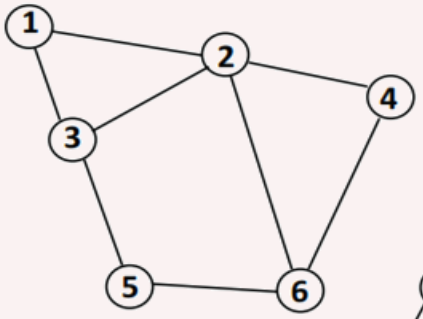
اختصاصی مهندسی کنترل
controlengineers
@controlengineers

تئوری گراف

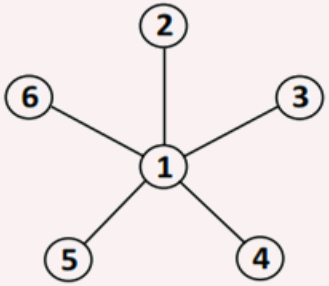
انواع مختلف گراف ها



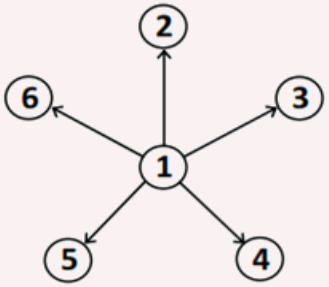
گراف جهت دار



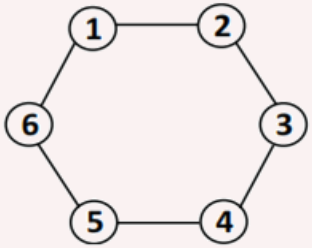
گراف غیر جهت دار



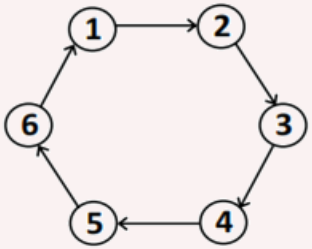
ستاره غیر جهت دار



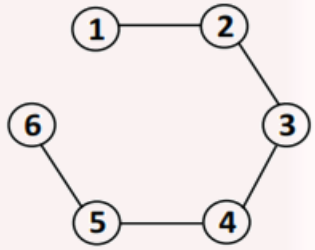
ستاره جهت دار



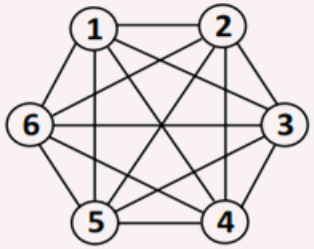
دایره غیر جهت دار (حلقه)



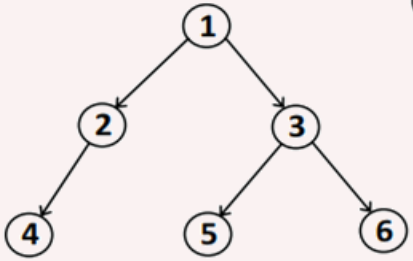
دایره جهت دار



مسیر غیر جهت دار



گراف کامل



درخت جهت دار (گراف آرایش بندی)

تعاریف خطاها

- ✓ **خطا:** انحراف یک متغیر یا پارامتر از رنج نامی آن
- ✓ **شکست سیستم:** توقف عملکرد سیستم
- ✓ **سه نوع خطا** در یک پروسه وجود دارند:
  - ✓ ۱- تغییر پارامتر
  - ✓ ۲- تغیر ساختار (سیستم یا کنترل کننده)
  - ✓ ۳- خطاهای سنسور یا عملگر
- ✓ **شناسایی خطا:** تعین اینکه کی خطا در سیستم رخ داده است
- ✓ **جداسازی خطا:** تعیین نوع خطا و مکان آن
- ✓ **بازیابی خطا:** راه حلی برای بازیابی اثرات ناشی از خطا
- ✓ **سیستم تحمل پذیر خطا:** سیستمی که عملکرد قابل قبول در حضور خطا داشته باشد

- شناسایی به موقع خطا و جداسازی آن (FDI) از شکست سیستم جلوگیری می کند.
- FDI باید سریع و دقیق باشد.
- روش های تشخیص خطا به دو کلاس کلی تقسیم می شوند:
  ۱- ایده های بر پایه مدل
  ۲- رویکرد های تاریخی
- این تقسیم بندی میتواند به صورت زیر باشد:
  ۱- روش های کیفی
  ۲- روش های کمی

روش های برپایه مدل کمی

- این ایده ها بر اساس مدل فضای حالت و مدل ورودی–خروجی پروسه هستند
- خطا به صورت یک ترم جمعی یا ضربی در نظر گرفته می شود
- ایده های پرکاربرد در این زمینه عبارتند از: رویتگرهای تشخیص، روابط همبستگی، فیلتر کالمن، فیلتر توسعه یافته کالمن، تخمین پارامترها

- همه روش های FDI بر پایه مدل، دو گام دارند:

۱- اول یک مانده که اختلاف بین سیستم واقعی و سیستم نامی را نشان میدهد، ایجاد می شوند.

۲- یک قانون تصمیم گیری برای تفسیر و شناسایی منبع این مانده ها پیاده سازی می شود.

روش های برپایه مدل کمی

- معادله تقارن (Party Equation) یک ایده مبتنی بر مدل کمی دیگر برای خطا است:
  - این روش بر پایه چک کردن ثبات مدل های سیستم با ورودی های معلوم و خروجی های سنسور اندازه گیری شده است.
  - معادلات parity با توجه به معادلات جبری که ارتباط بین بردارهای اطلاعات ورودی و خروجی را مشخص می کنند، تعریف می شوند.
  - در تشخیص خطا، بردارهای مانده فیزیبل می بایست با توجه به اینکه برای خطاهای مختلف متعامد باشند، تعریف می شوند.
  - در معادلات parity، ساختار مدل با توجه به اینکه خطای مطلوب جاسازی شده باشد، تنظیم مجدد می شود

روش های برپایه مدل کیفی

- وقتی مدل ریاضی دقیق سیستم در دسترس نباشد، رویکردهای کیفی استفاده می شوند.
- مدل های علی مبتنی بر دیاگرام، درخت های خطا، فیزیک کیفی و سلسله مراتب انتزاعی بر پایه این ایده هستند.
- اشکال عمده این روش ها، تولید راه حل های جعلی است چرا که هیچ مدل رفتاری کیفی ای نمیتواند به طور کامل و صریح یک سیستم را نمایندگی کند.

روش های برپایه مدل تاریخی

- اگر اطلاعات ذخیره شده از ورودی و خروجی موجود باشد، می توان این روش ها را اعمال کرد.
- برای استخراج اطلاعات مفید از داده ها برای تشخیص خطا از "استخراج ویژگی" استفاده می شود.
- استخراج ویژگی را میتوان در دو دسته تقسیم بندی کرد:
  ۱- استخراج ویژگی های کیفی شامل سیستم های متخصص و تحلیل روند کیفی (QTA)
  ۲- استخراج ویژگی های کمی شامل روش های آماری (تجزیه و تحلیل مولفه اصلی، حداقل مربعات جزئی و شبکه های عصبی) است.
- این روش ها بر انسجام داده های تاریخی برای استخراج ویژگی ها تکیه می کند

روش های بازیابی خطا (Fault Recovery Methods)

- دو روش کلی وجود دارد: غیرفعال و فعال:

  - کنترل فعال: هنگامی که یک خطا تشخیص داده می شود، سیستم کنترل به صورت آنلاین مجددا طراحی می شود.
  - کنترل غیر فعال: سیستم کنترل به گونه ای طراحی می شود که نسبت به وجود احتمالی خطا با دامنه مشخص، مقاوم باشد

- بر اساس نوع خطا، استراتژی بازیابی و دینامیک سیستم، روش های مختلفی برای FDIR توصیه می شود.
- هنگامی که خطای همزمان اتفاق می افتد، تشخیص و مقابله با آنها پیچیده می شود.

FDI در سیستم های چند عامله

- خطاهای اصلی در سیستم های چند عامله عبارتند از:
- خطا در عملگرها: اینکه عملگر کارایی خود را از دست داده و به خوبی عمل نمی کند.
  - به عنوان مثال سرعت یک UAV به مقدار مورد نظر نمی رسد و در نتیجه آرایش بندی مطلوب از بین می رود.
- خطا در سنسورها: وقتی یک عامل نمیتواند اطلاعات را از محیط اندازه گیری و دریافت کند.
  - به عنوان مثال وقتی یک UAV نمی تواند مکان خود و یا یک مانع را به درستی تعیین کند.
- خطا در ارتباطات: وقتی خطایی در سیستم ارتباطی ایجاد می شود؛ یعنی عامل ها نمی توانند از همسایگان خود اطلاعات دقیق دریافت کنند.
- با تبادل اطلاعات و همکاری مشارکتی در یک سیستم چند عامله میتوان به شناسایی خطا کمک کرد.

بازیابی خطا در سیستم های چند عامله

- در مأموریت های مختلف، استراتژی متفاوت برای بازیابی در نظر گرفته می شود:
- در آرایش بندی: اگر یکی از وسایل نقلیه بهره وری در عملگر را کاهش دهد و کندتر از آنچه انتظار می رود، حرکت کند :
  - وسایل نقلیه دیگر می توانند سرعت خود را برای حفظ شکل گیری کاهش دهند
  - یا، عامل دارای خطا را نادیده بگیرند و تغییر شکل دهند.
- در پوشش: اگر یکی از وسایل نقلیه بهره وری در سنسور را از دست داد، نمی تواند اطلاعات را به درستی تشخیص دهد:
  - استراتژی پوشش باید تغییر داده شود تا محیط را بدون نیاز به آن وسیله نقلیه بررسی کنند.
  - یا...؟
- در ملاقات و اجماع: اگر یکی از وسایل نقلیه دارای خطای محرک باشد:
  - عامل ها باید روی مقداری اجماع کنند که عامل دارای خطا بتواند به آن برسد.
  - یا عامل دارای خطا را نادیده بگیرند و ادامه دهند

بازیابی خطا در سیستم های چند عامله: مثال ۱

- ۴ ماهواره را با مدل دینامیکی انتگرال دوتایی در نظر بگیرید:

$$\ddot{x}_i = b_{xi} u_{xi} + d_x,$$
$$\ddot{y}_i = b_{yi} u_{yi} + d_y,$$
$$\ddot{z}_i = b_{zi} u_{zi} + d_z$$

- مکان ماهواره ها
- عملگرهای ماهواره ها
- بردارهای نیروی ورودی ماهواره
- اغتشاش محیطی

- $[x_i \ y_i \ z_i]$:
- $[b_{xi} \ b_{yi} \ b_{zi}]$:
- $[u_{xi} \ u_{yi} \ u_{zi}]$:
- $[d_x \ d_y \ d_z]$:

بازیابی خطا در سیستم های چند عامله: مثال ۱

- هدف: دستیابی به یک آرایش بندی (وضعیت یکسان) در صفحه x-y برای ماهواره ها
- برای ستیابی به یک آرایش بندی، حالت های نسبی مطلوب به صورت زیر در نظر گرفته می شوند:

$$x_{12}^d,\ x_{23}^d,\ x_{34}^d$$

- بردارهای خطا به صورت زیر تعریف می شوند:

$$e_{x12} = x_{12} - x_{12}^d,$$
$$e_{x23} = x_{23} - x_{23}^d,$$
$$e_{x34} = x_{34} - x_{34}^d$$

- می توان همین آرایش بندی را در صفحه z-y نیز در نظر گرفت

بازیابی خطا در سیستم های چند عامله: مثال ۱

$$p_{xij} = b_{xj}u_{xj} - b_{xi}u_{xi} = \ddot{x}_{ij}^{d} - \lambda_1 \dot{e}_{xij} - \lambda_0 e_{xij}$$

$$\begin{bmatrix} -b_{x1} & b_{x2} & 0 & 0 \\ 0 & -b_{x2} & b_{x3} & 0 \\ 0 & 0 & -b_{x3} & b_{x4} \end{bmatrix} \begin{bmatrix} u_{x1} \\ u_{x2} \\ u_{x3} \\ u_{x4} \end{bmatrix} = \begin{bmatrix} p_{x12} \\ p_{x23} \\ p_{x34} \end{bmatrix} \qquad (1)$$

- بازیابابی خطا در دو سطح انجام می شود:
- در بازیابی خطای سطح پایین، تابع هزینه زیر در نظر گرفته می شود:

$$\min_{u_i} \quad \sum_{i=1}^{4} \frac{\alpha_{xi}^2}{2} u_{xi}^2 \quad \text{subject to (1)}$$

$$\lambda_*, \alpha_{xi} \in \mathbb{R}_+.$$

بازیابی خطا در سیستم های چند عامله: مثال ۱

- با حل مساله بهینه سازی قبل از طریق لاگرانژین داریم:

$$\begin{bmatrix} u_{x1} \\ u_{x2} \\ u_{x3} \\ u_{x4} \end{bmatrix} = \begin{bmatrix} -b_{x1} & b_{x2} & 0 & 0 \\ 0 & -b_{x2} & b_{x3} & 0 \\ 0 & 0 & -b_{x3} & b_{x4} \\ \frac{\alpha_{x1}}{b_{x1}} & \frac{\alpha_{x2}}{b_{x2}} & \frac{\alpha_{x3}}{b_{x3}} & \frac{\alpha_{x4}}{b_{x4}} \end{bmatrix}^{-1} \begin{bmatrix} p_{x12} \\ p_{x23} \\ p_{x34} \\ 0 \end{bmatrix}$$

- در نتیجه تلاش ورودی بدست آمده، نتیجه می دهد:

$$\ddot{e}_{xij} = -\lambda_1 \dot{e}_{xij} - \lambda_0 e_{xij}$$

که پایداری سیستم چند عامله را تضمین می کند.

- لذا، با توجه به اینکه نیروی کنترل مینیمم مد نظر بوده، اثر خطای از بین رفتن کارایی در عملگرها تحمل پذیر شده است.

بازیابی خطا در سیستم های چند عامله: مثال ۱

- در بازیابی خطا در سطح آرایش بندی، هدف تخمین خطای بایاس روبرو است: b_{xi}
- اگر $\hat{b}_{xi}$ را به عنوان تخمین بایاس در نظر بگیریم:

$$\begin{cases} \hat{b}_{xi} - b_{xi} = \epsilon \\ \epsilon \in \mathbb{R}_+ \end{cases}$$

فرض می شود که عامل دارای خطا تشخیص داده شده است.

- سطح آرایش بندیی، می بایست یک ضریب α_{xi} به اندازه کافی بزرگ را متناسب با عملگر تخمین زده شده $\hat{b}_{xi} \neq b_{xi}$ انتخاب کند.

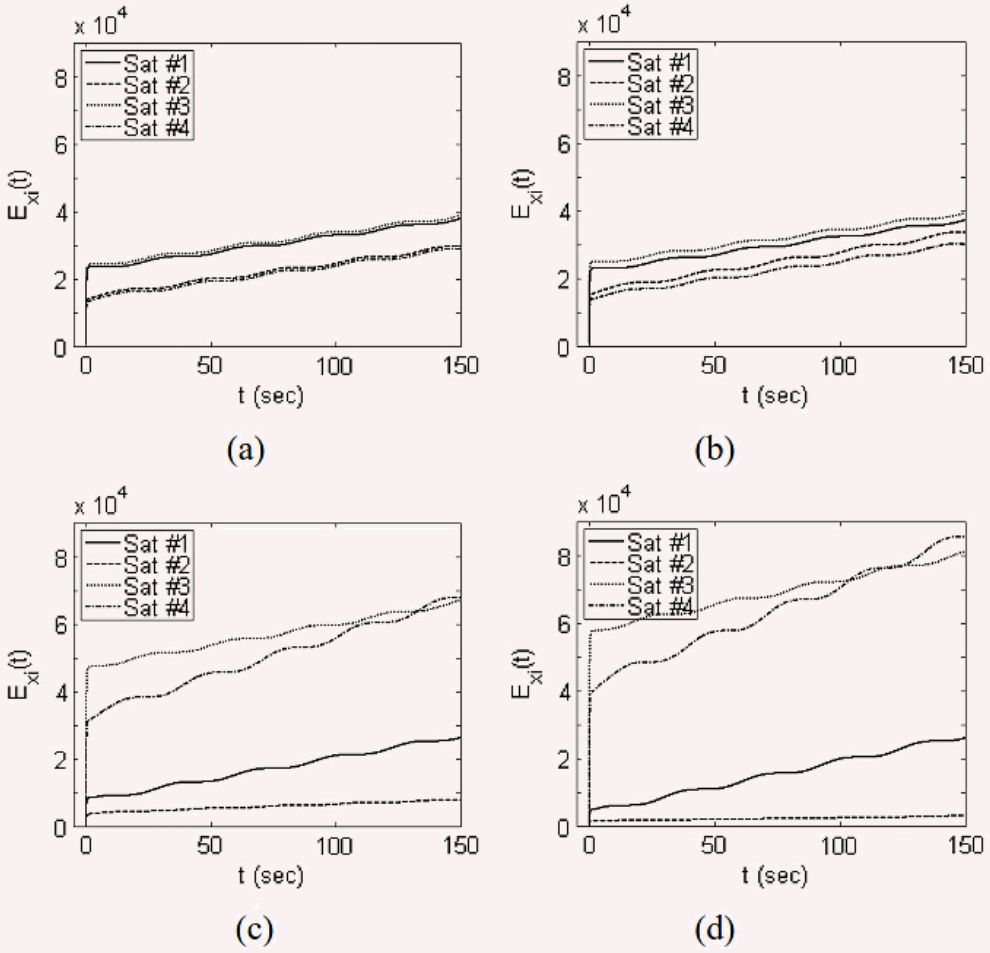
بازیابی خطا در سیستم های چند عامله: مثال ۱

- فرض کنید ماهواره ها می بایست در یک آرایش بندی مربعی با ضلع ۲۰۰ متر قرار بگیرند
- هدف، یک مانور چرخش بر خلاف ساعت در صفحه x-y با فرکانس ۰.۱ رادیان بر مجذور ثانیه است.
- اغتشاش محیطی برابر است با: $10^{-5}\mathrm{N}$
- ۲۰٪ از دست دادن کارایی و ۱۰٪ بایاس روی b_{x2} در نظر گرفته می شود.
- برای تحلیل عملکرد سیستم چند عامله، تلاش ورودی در جهت محور x به صورت زیر در نظر گرفته می شود:

$$E_{xi}(t) = \int_{\tau=0}^{t} u_{xi}^2(\tau)\mathrm{d}\tau, i = 1, 2, 3, 4$$

اختصاصی مهندسی کنترل
controlengineers
@controlengineers

بازیابی خطا در سیستم های چند عامله: مثال ۱



بازیابی خطا در سیستم های چند عامله: مثال ۱

- شکل a : حالت بدون خطا
- شکل b: حالت خطا در بازیابی خطای سطح پایین $(\alpha = 1)$

  ماهواره دارای خطا نسبتا بازیابی می شود.
- شکل c: حالت خطا با بازیابی خطای سطح آرایش بندی $(\alpha = 6)$
- شکل d: حالت خطا با بازیابی خطای سطح آرایش بندی$(\alpha = 10)$
- تلاش ورودی جمعی محور X برای دومین ماهواره به صورت مشخصی کاهش می یابد.
- در این شرایط، سایر ماهواره ها تلاش کنترلی بیشتری بکار می برند تا عدم کارایی ماهواره دوم را جبران کنند.

بازیابی خطا در سیستم های چند عامله: مثال ۲

- یک سیستم چند عامله را در $\mathbb{R}^2$ به صورت زیر در نظر بگیرید:

$$\ddot{\mathbf{p}}_k = \mathbf{u}_k, k = 1, 2, 3, 4.$$

- همه عامل هایک دایره با مرکز $[x_v \ y_v]^\top$ و شعاع α را دنبال می کنند (یک عامل مجازی)
- همه عامل ها با با شارژ یکسان q مدل می شوند که یکدیگر را دفع می کنند.
- قانون کنترل $[f_{xk}, f_{yk}]^\top$ برای k امین عامل به صورت زیر است:

$$f_{xk} = F_{xk} - k_{sk}\Big((x_k - x_v)\big((x_k - x_v)^2 + (y_k - y_v)^2 - \alpha^2\big)\Big),$$

$$f_{yk} = F_{yk} \underbrace{-k_{sk}\Big((y_k - y_v)\big((x_k - x_v)^2 + (y_k - y_v)^2 - \alpha^2\big)\Big)}_{\text{attraction forces toward the circle}}$$

$k_{sk} \in \mathbb{R}_+$

بازیابی خطا در سیستم های چند عامله: مثال ۲

- بعلاوه، نیروهای دافعه عبارتند از:

$$F_{xk} = k_r \sum_{i=1, i\neq k}^{N} \frac{q^2}{r_{ki}^2} \cos(\theta_{ki}),$$

$$F_{yk} = k_r \sum_{i=1, i\neq k}^{N} \frac{q^2}{r_{ki}^2} \sin(\theta_{ki})$$

- که در آن داریم: $k_r \in \mathbb{R}_+,\ r_{ki} = \|\mathbf{p}_k - \mathbf{p}_i\|,$

$$\cos(\theta_{ki}) = \frac{x_k - x_i}{r_{ki}},\quad \sin(\theta_{ki}) = \frac{y_k - y_i}{r_{ki}}.$$

- هر عامل نقطه تعادل و مکان خود را برای قرار گیری در یک آرایش بندی چند ضلعی منظم به صورت اتووماتیک پیدا می کند.
- وقتی تعداد عامل ها زیاد می شود این امر مفید خواهد بود.

بازیابی خطا در سیستم های چند عامله: مثال ۲

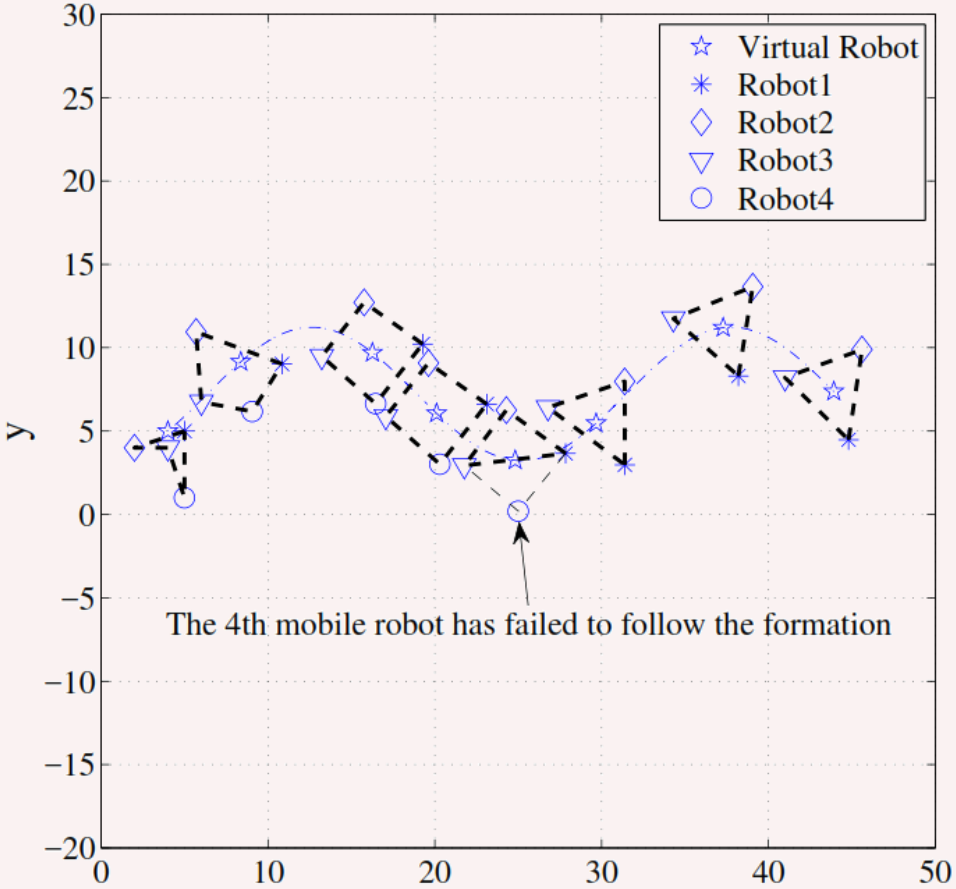
- برای شبیه سازی فرض می کنیم که: $k_{sk} = 100, k = 1, 2, 3, 4, \quad k_r = 10, \alpha = 3, q = 10$
- فرض می کنیم نیروی ورودی $\mathbf{u}_k$ به صورت روبرو محدود باشد: $[-20\ 20] \times [-20\ 20]$
- مکان اولیه عامل ها به صورت زیر فرض می شود: $(5, 5), (2, 4), (4, 4), (5, 1).$
- مکان اولیه عامل مجازی به صورت روبرو فرض می شود: $(4, 5)$

$$\mathbf{u}_v = -10\dot{\mathbf{p}}_v + [8, 8\sin(\tfrac{x_v}{4})]^\top$$

- فرض می کنیم که وسیله چهارم در $x_4 = 25$ شناور است.

اختصاصی مهندسی کنترل
controlengineers
@controlengineers

بازیابی خطا در سیستم های چند عامله: مثال ۲



بازیابی خطا در سیستم های چند عامله: مثال ۳

- یک سیستم چند عامله را با دینامیک خطی زیر در نظر بگیرید:

$$\dot{x}^{(i)} = A^{(i)}x^{(i)} + \Gamma^{(i)}e + F_1^{(i)}\mu_1^{(i)} + F_2^{(i)}\mu_2^{(i)}$$
$$y^{(i)} = C^{(i)}x^{(i)} + v^{(i)}$$

- که در آن:

حالت عامل ها: $x^{(i)} \in \mathbb{R}^n$:

ورودی عامل ها: $e \in \mathbb{R}^m$:

خروجی عامل ها: $y^{(i)} \in \mathbb{R}^p$:

نویز اندازه گیری: $v^{(i)} \in \mathbb{R}^p$:

خطای هدف: $\mu_1^{(i)} \in \mathbb{R}$:

خطای دوم: $\mu_2^{(i)} \in \mathbb{R}$:

و سایر ماتریس ها دارای ابعاد مناسب هستند.

بازیابی خطا در سیستم های چند عامله: مثال ۳

- عامل iام قادر به تخمین $\hat{x}^{(i)}$ از بردار حالت محلی $x^{(i)}$ با استفاده از تخمین گر محلی زیر است:

$$\dot{\hat{x}}^{(i)} = A^{(i)}\hat{x}^{(i)} + L^{(i)}(y^{(i)} - C^{(i)}\hat{x}^{(i)})$$

- که در آن $L^{(i)}$ گین فیلتری حالت ماندگار است.

بازیابی خطا در سیستم های چند عامله: مثال ۳

- سیگنال های مانده می توانند برای شناسایی خطاها مورد استفاده قرار بگیرند.
- هر مانده، به گونه ای طراحی می شود که نسبت به یک مجموعه از خطاها حساس باشد..
- مانده محلی متناظر عبارت است از:

$$r^{(i)} = H^{(i)}(y^{(i)} - C^{(i)}\hat{x}^{(i)})$$

- که در آن $H^{(i)}$ یک ماتریس طراحی مناسب است.
- با استفاده از ایده کنترل بهینه، $L^{(i)}$ و $H^{(i)}$ با توجه به اینکه مانده به صورت اولیه تحت تاثیر خطای هدف $\mu_1^{(i)}$ و حداقل تحت خطای دوم $\mu_2^{(i)}$ و نویز ها باشد بدست می آیند.

بازیابی خطا در سیستم های چند عامله: مثال ۳

- تخمینگر های محلی بدلیل در دسترس نبودن اندازه گیری ها و نیز مناسب نبودن مدل های محلی ناکارآمد هستند.
- هدف، فرموله کردن یک استراتژی تخمین بر پایه اجماع یا همکاری مشارکتی است.
- هر عامل مانده ها را به گونه ای که بیشتر نسبت به خطای هدف و کمتر نسبت به خطای دوم حساس باشند، تولید می کند:

$$\dot{\xi}^i = A_i\xi^i + \sum_{j=1,j\neq i}^{3} K_{ij}(\xi^j - \xi^i) + L_i(y^{(i)} - C_i\xi^i)$$

- که در ان ξ^i مقدار تخمین زده شده حالت عامل iام است.
- مقادیر $A_i,\ C_i,\ L_i$ از مقادیر $A^{(i)},\ C^{(i)},\ L^{(i)}$ و با استفاده از یک ایده بهینه بدست می آیند.

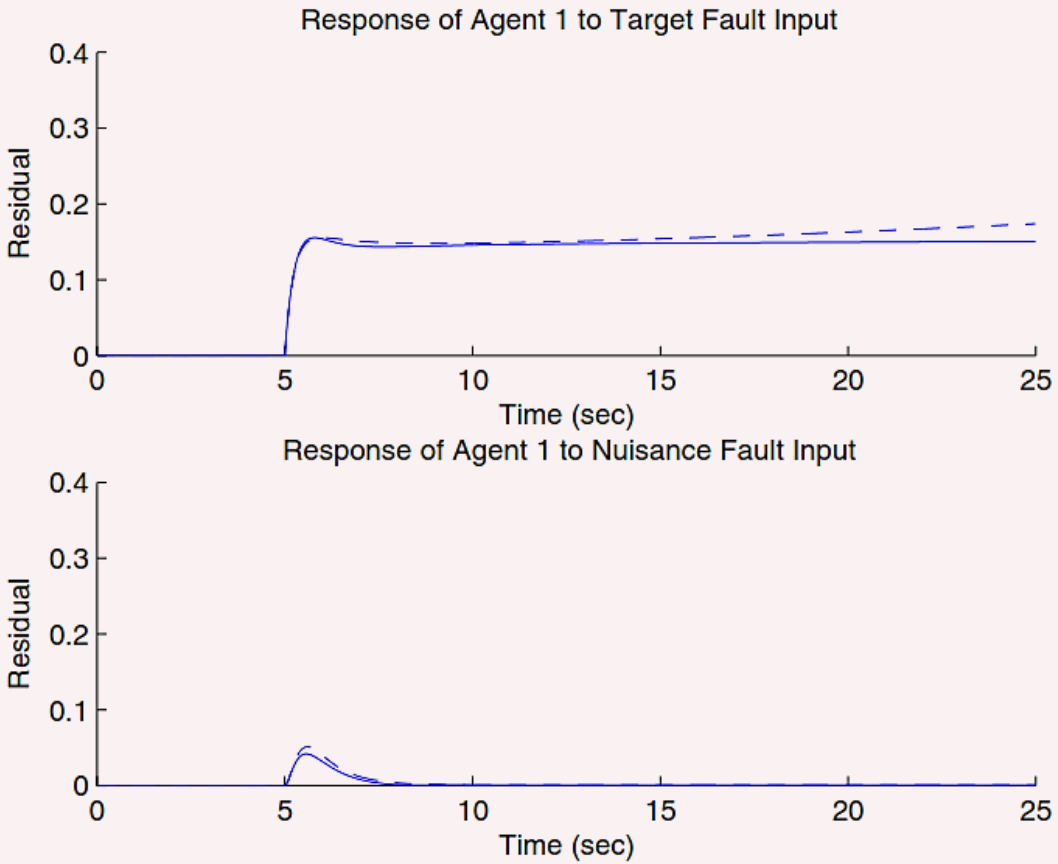
بازیابی خطا در سیستم های چند عامله: مثال ۳

- برای شبیه سازی:

- سه عامل را در نظر بگیرید.
- خطاهای هدف، توابع پله با دامنه ۱ هستند که در زمان t=5 رخ می دهند.
- با توجه به شکل در صفحات بعد می توان گفت که مانده های استراتژی بر پایه اجماع (خطوط عادی) بیشتر به خطاها حساس هستند.

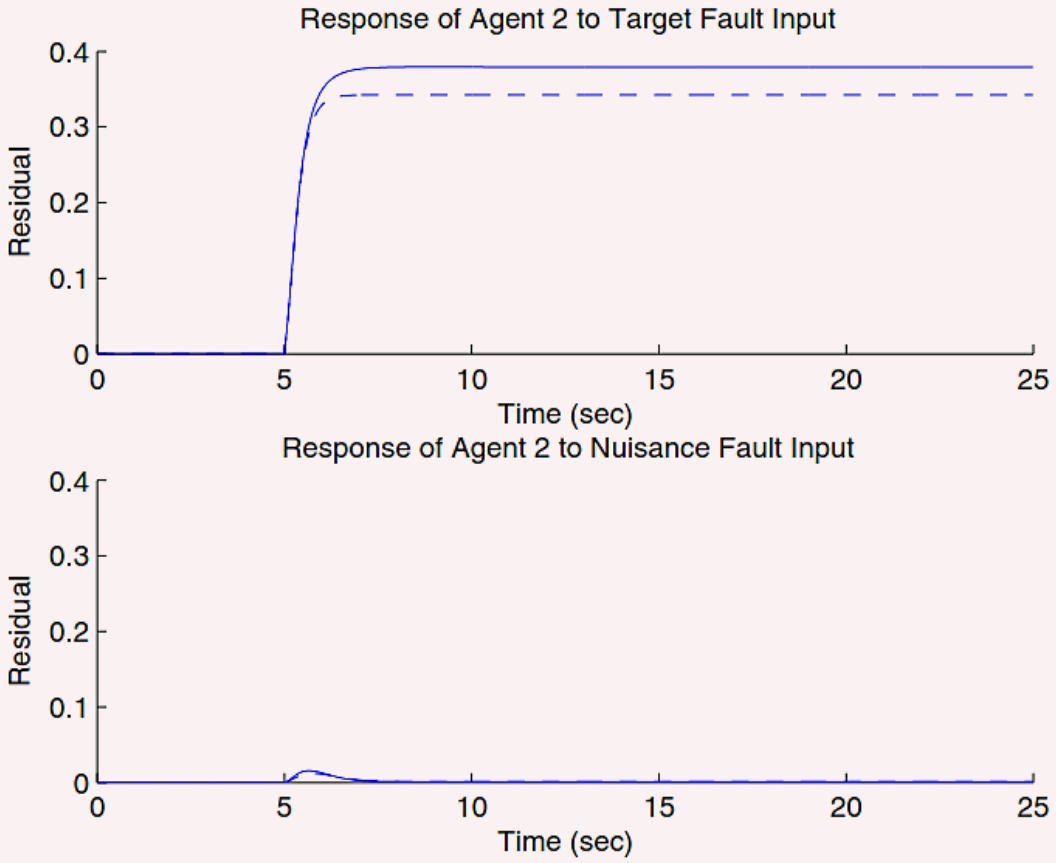
اختصاصی مهندسی کنترل
controlengineers
@controlengineers

بازیابی خطا در سیستم های چند عامله: مثال ۳



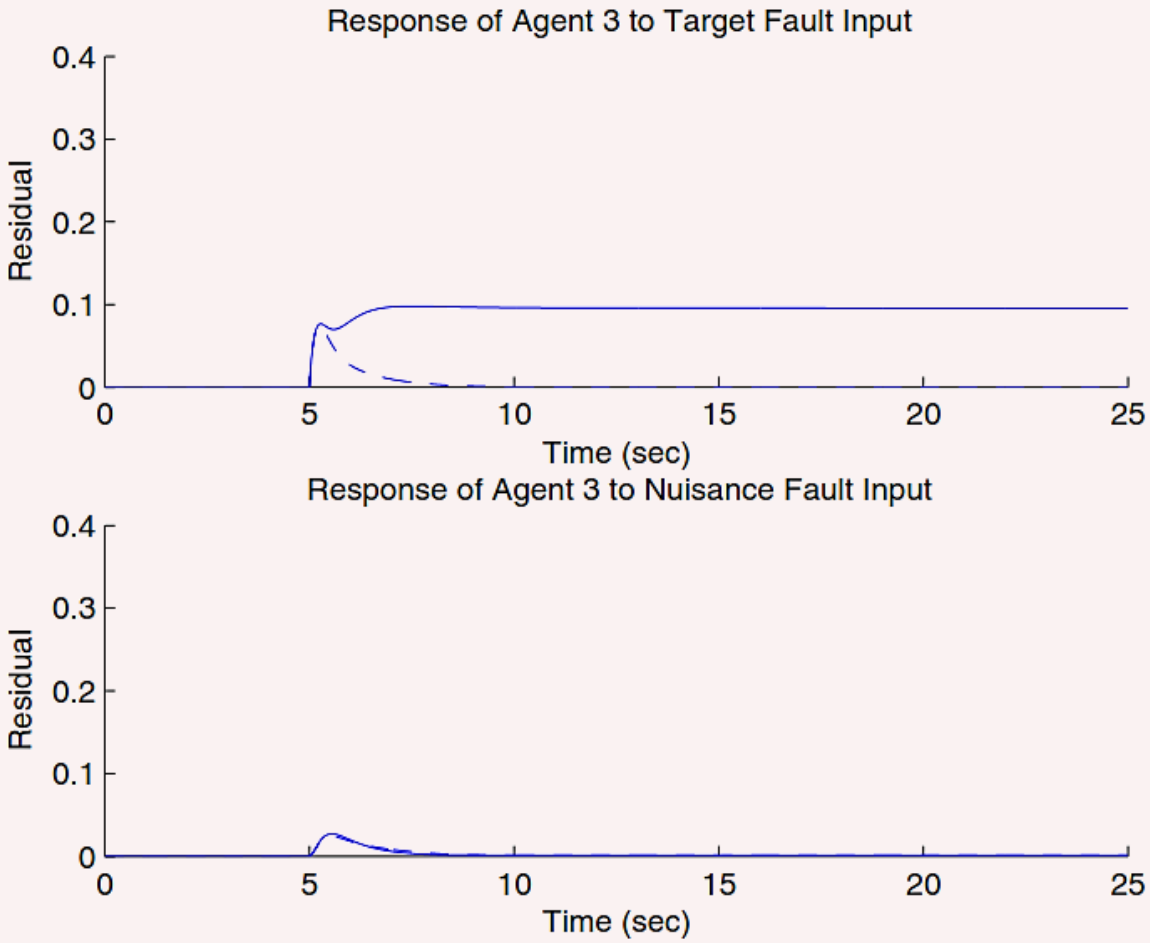
اختصاصی مهندسی کنترل
controlengineers
@controlengineers

بازیابی خطا در سیستم های چند عامله: مثال ۳



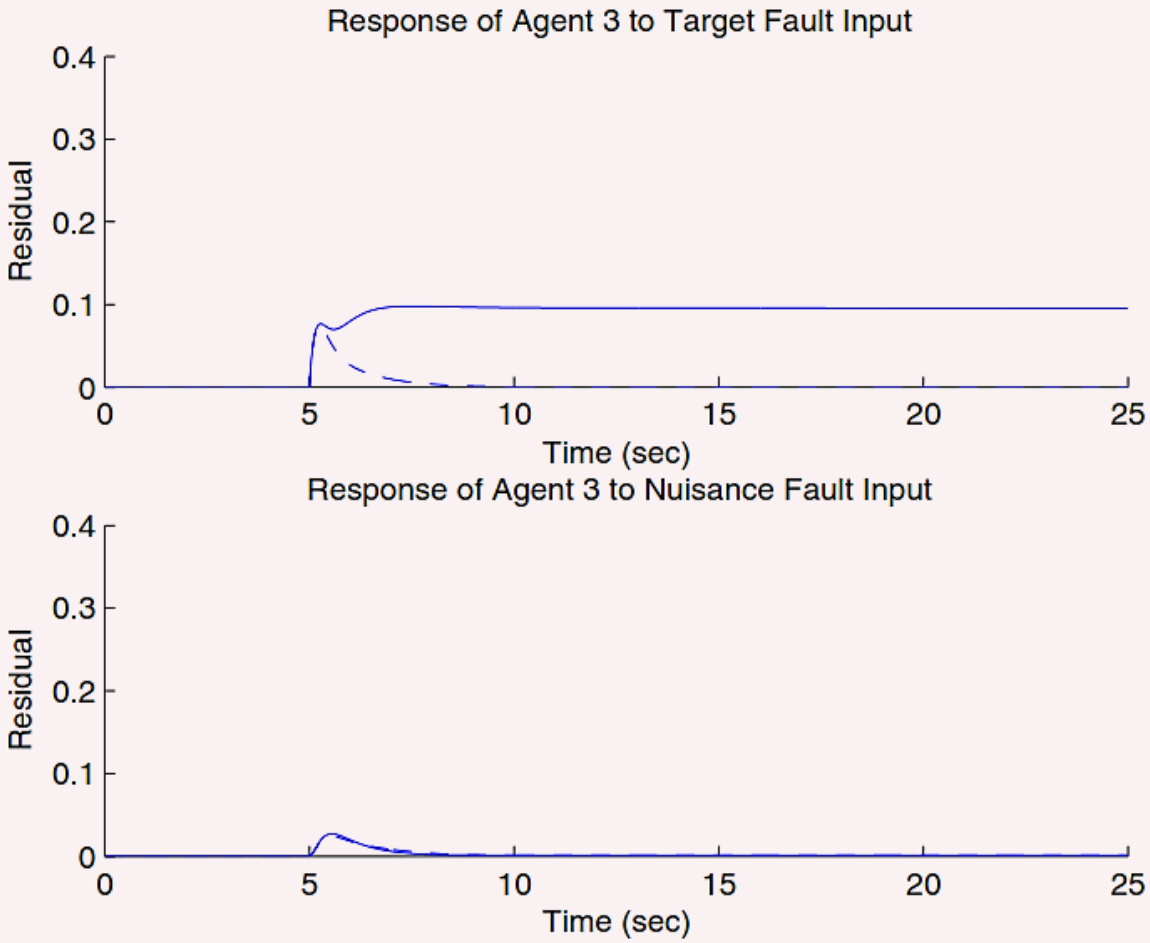
اختصاصی مهندسی کنترل
controlengineers
@controlengineers

بازیابی خطا در سیستم های چند عامله: مثال ۳



اختصاصی مهندسی کنترل
controlengineers
@controlengineers

بازیابی خطا در سیستم های چند عامله: مثال ۳



جمع بندی

- مقدمات سیستم های چند عامله و تعاریف انواع خطا و مقدمه ای بر تئوری گراف
- انواع روش های تشخیص و جداسازی و بازیابی خطا
- تشخیص و بازیابی خطا در سیستم های چند عامله
- سه مثال مختلف از تشخیص و جداسازی خطا و حفظ عملکرد در حضور خطا در سیستم های چند عامله

مراجع

- S. M. Azizi and K. Khorasani, "A hierarchical architecture for cooperative fault accommodation of formation flying satellites in deep space," (USA), pp. 4178–4183, July 2009.
- H. Rezaee and F. Abdollahi, "A decentralized cooperative control scheme with obstacle avoidance for a team of mobile robots," IEEE Transactions on Industrial Electronics, vol. 61, no. 1, pp. 347–354, Jan 2014, Doi:10.1109/TIE.2013.2245612.
- S. Stankovic and N. Ilic and Z. Djurovic and M. Stankovic and K. H. Johansson, "Consensus based overlapping decentralized fault detectionand isolation," (France), pp. 570–575, October 2010.

References

- Davoodi, M.R. and Khorasani, K. and Talebi, H.A. and Momeni, H.R., "A novel distributed robust fault detection and isolation filter design for a network of nonhomogeneous multi-agent systems," in Proceedings of the 51st IEEE Conference on Decision and Control, (USA), pp. 592–599, December 2012.
- N. Meskin, Fault Detection and Isolation in a Networked Multi-Vehicle Unmanned System. PhD thesis, Concordia University, Montreal,QC, Canada, 2008.
- S. M. Azizi, Cooperative Fault Estimation and Accommodation in Formation Flight of Unmanned Vehicles. PhD thesis, Concordia University, Montreal,QC, Canada, 2010.

با تشکر از توجه شما