دانشگاه صنعتی اصفهان

دانشکده برق و کامپیوتر

عنوان سمینار و زمینه اصلی:

یادگیری کنترل ربات با مشاهده ی چندین نمایش

استاد درس:

دکتر پالهنگ

دانشجو:

داریوش حسن پور

هوش مصنوعی (۹۳۰۸۱۶۴)

۱۳۹۳

۱ - چکیده

کنترل ربات یکی از مسائل مهمی است که امروزه در طراحی و ساخت ربات‌های خاص‌منظوره مورد بحث قرار میگیرد. به علت وجود متغییرهای پنهانی که در محیط پیرامون ما که انسان‌ها درک کامل و صحیحی نسبت به آنها ندارد کنترل ربات به مسائل کوچک و تک‌منظوره‌ای محدود میشود و برای کنترل‌های پیچیده بسیار دشوار میباشد. در این نوشتار به توضیح روشی میپردازیم که با استفاده از نمونه‌های اجرایی و مشاهدات انجام شده توسط انسان الگوریتمی ارائه شده است که امکان یادگیری الگوی کنترلی ربات با استفاده از نمونه‌های اجرا شده قبلی توسط انسان را دارد.

فهرست مطالب

فهرست اشکال

فهرست روابط

۲ – مقدمه

کنترل ربات یکی از مسائل مهمی است که امروزه در طراحی و ساخت ربات‌های خاص‌منظوره مورد بحث قرار میگیرد. به علت وجود متغییرهای پنهانی که در محیط پیرامون ما که انسان‌ها درک کامل و صحیحی نسبت به آنها ندارد کنترل ربات به مسائل کوچک و تک‌منظوره‌ای محدود میشود و برای کنترل‌های پیچیده بسیار دشوار میباشد. در این نوشتار به توضیح روشی میپردازیم که با استفاده از نمونه‌های اجرایی و مشاهدات انجام شده توسط انسان الگوریتمی ارائه شده است که امکان یادگیری الگوی کنترلی ربات با استفاده از نمونه‌های اجرا شده قبلی توسط انسان را دارد. با این حال که الگوریتم‌هایی مانند یادگیری تقویتی و شبکه‌های عصبی مصنوعی در گذشته برای کنترل ربات‌ها موفق عمل کرده‌اند ولی آنها از قابلیت‌های محدودی برای کنترل به علت ماهیت‌شان برخوردار بودند در این نوشتار الگوریتمی که ارائه می‌شود امکان یادگیری کنترل(مسیر) مورد دلخواه و هم‌ترازی‌های زمانی نمونه‌های اجرا شده را به ما می‌دهد. در قسمت ۳ این نوشتار با به شرح مختصری از آنچه‌که در گذشته در برای کنترل بال‌گرد انجام شده می‌پردازیم؛ در قسمت ۴ به شرح الگوریتم ارائه شده می‌پردازیم و در قسمت ۵ نتایج حاصله از الگوریتم را مورد بررسی قرار داده و در قسمت ۶ نتیجه‌گیری کرده و در انتها منابع مورد استفاده را نام برده‌ایم.

۳ - کارهای انجام شده در گذشته

در گذشته برای کنترل هوشمند ربات‌های تحقیقات زیادی انجام شده است که در این نوشتار به معرفی تعدادی از تحقیقات دکتر اندرو ان.جی[۱] استاد دانشگاه استنفورد و دانشجویان ایشان می‌پردازیم.

ربات بال‌گرد یکی از ربات‌های می‌باشد که دارای روابط کنترلی احتمالی, غیرخطی و پویا می‌باشد و که به همین دلیل کنترل ربات‌های بال‌گرد به یکی از زمینه‌های چالش برانگیز برای کنترل ربات‌های تبدیل شده است. بال‌گردها در سرعت‌های بالا استواری خوبی از خود نشان می‌دهند ولی در سرعت‌های پایین بسیار ناپایدار بوده‌اند و همچنین پرواز بال‌گردها به صورت وارون برای خلبانان خبره انسانی سخت بوده برای همین موضوع کنترل پرواز وارن در سرعت پایین که برای فرد خبره‌ی انسانی بسیار سخت بوده و نمی‌تواند برای مدت طولانی پایداری ربات را حفظ نماید؛ به موضوع جالب در کنترل بال‌گردها تبدیل شده است.

یکی از روش‌های معمول کنترل استفاده از یادگیری تقویتی می‌باشد. که در [۱] برای کنترل بال‌گرد از ۴ دستور کنترلی استفاده کرده‌اند:

- دستورهای ۱و۲ شامل زاویه‌ی طولی(جلو-عقب) و زاویه‌ی عرضی(چپ-راست) بال‌گرد.
- دستور کنترلی ۳ سرعت پروانه‌ی اصلی بال‌گرد.
- دستور کنترلی ۴ سرعت پروانه‌ی دم بال‌گرد.

با استفاده از دستورهای کنترلی بالا مدلی برای یادگیری تقویتی برای کنترل بال‌گرد ارائه داده‌اند که توانسته است بال‌گرد را با سرعت پایین به طور وارون به پرواز درآورد.

بعد از کار ان.جی بر پرواز وارون بال‌گرد با استفاده از یادگیری تقویتی یک گام به جلو رفتند و در پی یادگیری پویای بال‌گرد بوده‌اند که در [۲] با استفاده از داده‌های جمع‌آوری شده از پرواز بال‌گرد به صورت وضعیت-عمل سعی بر یادگیری پویای بال‌گرد داشته‌اند که بر اساس مدل تصمیم‌گیری مارکوف پیاده‌سازی کرده‌اند. در [۳] نیز باردیگر مباحث مطرح شده در [۱] را بهبود بخشیدند و یادگیری تقویتی را با برنامه نویسی پویا افتراقی ادغام کرده‌اند و مدلی جهت یادگیری بهتر کنترل ربات با استفاده از یادگیری تقویتی و برنامه نویسی پویا افتراقی ارائه داده‌اند. ولی در [۴] روشی نوین نه‌تنها برای کنترل رباط بلکه برای یادگیری مسیر و مانورو ها ارائه داده اند که از روش‌های دیگر از خیلی از جهات برتری داشته است که در قسمت بعدی به بررسی روش ارائه شده می‌پردازیم.

[۱] Andrew Ng

۴ - روش یادگیری کنترل ربات با مشاهده ی چندین نمایش

اگر به تعداد M عدد نمونه نمایش داشته باشیم که هریک به اندازه‌ی N^k برای k=0..M-1 و هر شامل وضعیت های s_i^k و ورودی کنترلی u_j^k به صورت بردار حالت زیر نمایش داده شود:

$$y_j^k = \begin{bmatrix} s_j^k \\ u_j^k \end{bmatrix}, \text{ for } j = 0..N^k - 1, k = 0..M - 1$$

رابطه ۱: نحوه‌ی نمایش بردار حالت برای هر وضعیت و ورودی کنترلی

و برای مسیر مخفی به طول T رابطه ۲ را داریم به تجربه دریافتند که این مقدار اگر برابر با ۲ برابر میانگین طول نمونه های مشاهده شده باشد دارای بهینه‌ترین هم‌پوشانی زمانی می‌باشد.

$$z_t = \begin{bmatrix} s_t^\star \\ u_t^\star \end{bmatrix}, \text{ for } t = 0..T - 1.$$

رابطه ۲: نحوه ی نمایش بردار حالت برای مسیر مخفی

و برای مدل حالت اولیه در مسیر مخفی داریم

$$z_0 \sim \mathcal{N}(\mu_0, \Sigma_0)$$

رابطه ۳: مدل حالت اولیه مسیر مخفی

و مدل تقریب‌زن مدل برای حالات مسیر مخفی داریم

$$z_{t+1} = f(z_t) + \omega_t^{(z)}, \quad \omega_t^{(z)} \sim \mathcal{N}(0, \Sigma^{(z)})$$

رابطه ۴: مدل تقریب‌زن برای حالات مسیر مخفی که ω اغتشاش با میانگین صفر می‌باشد

فرض میکنیم حالات مشاهده شده در نمایش‌های انجام شده به از مدل رابطه ۵ پیروی میکنند

$$y_j^k = z_{\tau_j^k} + \omega_j^{(y)}, \quad \omega_j^{(y)} \sim \mathcal{N}(0, \Sigma^{(y)})$$

رابطه ۵: مدل مفروض برای حالات مشاهده شده در نمایش‌ها که ω اغتشاش با میانگین صفر می‌باشد

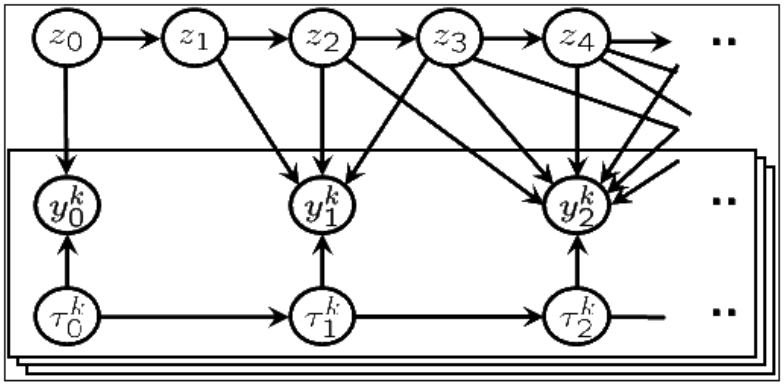
اگوریتم ارائه شده توانایی هم‌ترازی زمانی نمونه‌های مشاهده شده را دارد بنابراین متغییر به نام τ را معرفی کرده‌اند که قبلا اطلاعی راجع به این متغییر در دست نمی‌باشد و فرض می کنیم که از توزیع رابطه ۵ پیروی میکنند.

$$\mathbb{P}(\tau_{j+1}^k|\tau_j^k) = \begin{cases} d_1^k & \text{if } \tau_{j+1}^k - \tau_j^k = 1 \\ d_2^k & \text{if } \tau_{j+1}^k - \tau_j^k = 2 \\ d_3^k & \text{if } \tau_{j+1}^k - \tau_j^k = 3 \\ 0 & \text{otherwise} \end{cases}$$

$$\tau_0^k \equiv 0.$$

رابطه ۶: توزیع متغییر های هم‌ترازی زمانی که برای هم‌ترازی زمانی نمونه‌های مشاهده شده استفاده می‌گردد

در شکل ۱ مدل گرافیکی مسیر مخفی و مسیر مورد نمایش و شاخص‌های زمانی آمده است. همان‌طورکه مشاهده می‌شود هر حالت مورد نمایش در هر زمان می‌تواند به چندین حالت مخفی مرتبط باشد. حال اگر مقادیر شاخص‌های زمانی مشخص باشند مساله به یک مساله‌ی مدل مخفی مارکوف تبدیل می‌شود ولی اگر مشخص نباشند یادگیری سخت می‌شود.



شکل ۱: مدل گرافیکی از مسیر مورد نمایش و مسیر مخفی و شاخص‌های زمانی مرتبط

الگوریتم سعی دارد که با استفاده از ترکیب مدل مخفی مارکوف و الگوریتم بیشنه سازی امید ریاضی شاخص‌های تنظیم زمانی, مقادیر هم‌پراشی اغتشاش‌ها و احتمال انتقالی از وضعیتی به وضعیت دیگر را یاد بگیرد که در رابطه ۷ نشان داده شده اند.

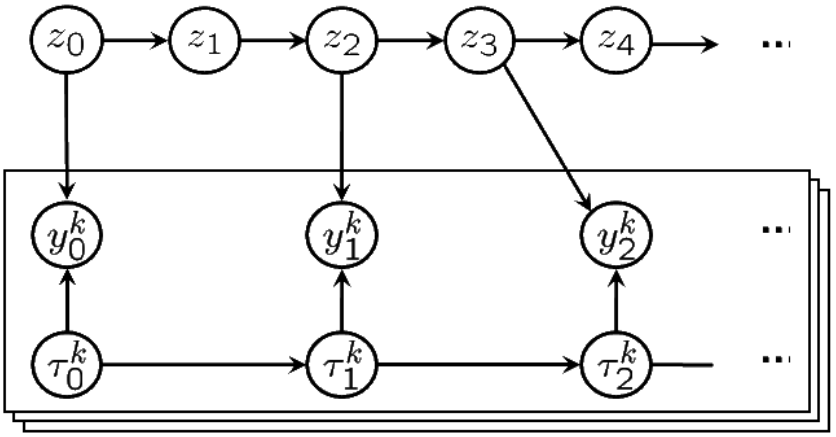
$$\max_{\boldsymbol{\tau},\Sigma^{(\cdot)},\mathbf{d}} \log \mathbb{P}(\mathbf{y},\boldsymbol{\rho},\boldsymbol{\tau}\ ;\ \Sigma^{(\cdot)},\mathbf{d})$$

رابطه ۷: هدف الگوریتم

که الگوریتم به صورت زیر است

- متغییرها را مقداردهی اولیه کن
- تا زمان همگرایی اجرا کن
  - با استفاده از مقادیر کنونی $\boldsymbol{\tau}$ الگوریتم بیشنه سازی امید ریاضی برای مدل مخفی مارکوف کنونی را اجرا کن.
  - با استفاده از برنامه نویسی پویا مقادیر $\boldsymbol{\tau}$ را بروز رسانی کن.

بعد از یادگیری متغییرهای رابطه ۷ شکل ۱ به صورت شکل ۲ یادگرفته خواهد شد همان‌طورکه می‌بینید فقط ارتباط یکی از حالات مخفی حفظ شده است و بقیه حذف گردیده‌اند که در نهایت مارا به یک مدل یکتا هدایت میکند.



شکل ۲: مدل گرافیکی یادگرفته شده بعد از اجرای الگوریتم بروی شکل ۱

همان‌طور که در رابطه ۷ آمده است الگوریتم ما به دنبال یادگیری τ و مقادیر پراش‌ها و پرامترهای توزیع شاخص‌های زمانی می‌باشد. برای بهینه‌سازی رابطه ۷ در بالا الگوریتم در حالت عمومی آمده است که الگوریتم دقیق برای این بهینه‌سازی در الگوریتم ۱ آمده است.

1. Initialize the parameters to hand-chosen defaults. A typical choice: $\Sigma^{(\cdot)} = I$, $d_i^k = \frac{1}{3}$, $\tau_j^k = \lceil j \frac{T-1}{N^k-1} \rceil$.
2. E-step for latent trajectory: For the current setting of $\boldsymbol{\tau}, \Sigma^{(\cdot)}$ run a (extended) Kalman smoother to find the distributions for the latent states, $\mathcal{N}(\mu_{t|T-1}, \Sigma_{t|T-1})$.
3. M-step for latent trajectory: Update the covariances $\Sigma^{(\cdot)}$ using the standard EM update.
4. E-step for the time indexing (using hard assignments): run dynamic time warping to find $\boldsymbol{\tau}$ that maximizes the joint probability $\mathbb{P}(\bar{\mathbf{z}}, \mathbf{y}, \boldsymbol{\rho}, \boldsymbol{\tau})$, where $\bar{\mathbf{z}}$ is fixed to $\mu_{t|T-1}$, namely the mode of the distribution obtained from the Kalman smoother.
5. M-step for the time indexing: estimate $\mathbf{d}$ from $\boldsymbol{\tau}$.
6. Repeat steps 2-5 until convergence.

الگوریتم ۱: الگوریتم بهینه سازی رابطه ۷

که برای گام‌های ۲ و ۳ الگوریتم به ترتیب مراحل E و M الگوریتم EM می‌باشند که به یک سیستم غیرخطی پویا با اغتشاش گوسی اعمال می‌شود. با توجه به رابطه ۴ و رابطه ۵ صافی کالمن برای گام دوم الگوریتم را اجرا میکنیم ما میتوانیم مقادیر Q و R موجود در رابطه ۴ و رابطه ۵ در گام M پیدا کنیم که به صورت میباشد.

$$
\begin{aligned}
\delta\mu_t &= \mu_{t+1|T-1} - f(\mu_{t|T-1}),\\
A_t &= \mathcal{D}f(\mu_{t|T-1}),\\
L_t &= \Sigma_{t|t}A_t^\top \Sigma_{t+1|t}^{-1},\\
P_t &= \Sigma_{t+1|T-1} - \Sigma_{t+1|T-1}L_t^\top A_t^\top - A_t L_t \Sigma_{t+1|T-1},\\
Q &= \frac{1}{T}\sum_{t=0}^{T-1} \delta\mu_t \delta\mu_t^\top + A_t\Sigma_{t|T-1}A_t^\top + P_t,\\
\delta y_t &= y_t - h(\mu_{t|T-1}),\\
C_t &= \mathcal{D}h(\mu_{t|T-1}),\\
R &= \frac{1}{T}\sum_{t=0}^{T-1} \delta y_t \delta y_t^\top + C_t\Sigma_{t|T-1}C_t^\top .
\end{aligned}
$$

الگوریتم ۲: نحوه ی محاسبه مقادیرQ وR موجود در رابطه ۴ و رابطه ۵ در گام M

و گام ۴ الگوریتم به صورت رابطه ۸ محاسبه میکنیم.

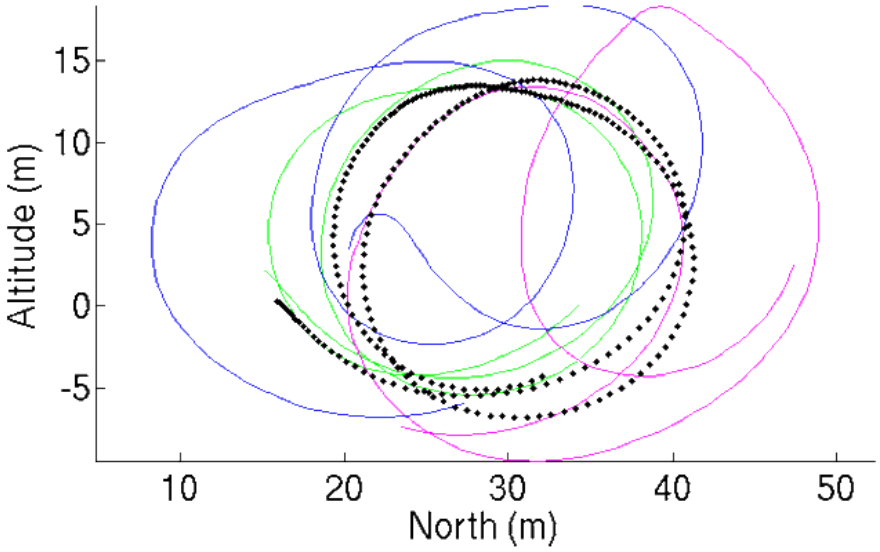
$$
\bar{\tau} = \arg\max_{\tau} \sum_{k=0}^{M-1}\sum_{j=0}^{N^k-1} \left[\ell(y_j^k|\bar{z}_{\tau_j^k}, \tau_j^k) + \ell(\tau_j^k|\tau_{j-1}^k)\right]
$$

رابطه ۸: محاسبه شاخص زمانی (اصلاح نشده)

و در گام ۵ الگوریتم به محاسبه مقادیر d می‌پردازیم توسط روش استاندارد تخمین بیشترین شباهت برای توزیع چند بعدی محاسبه می‌شوند.

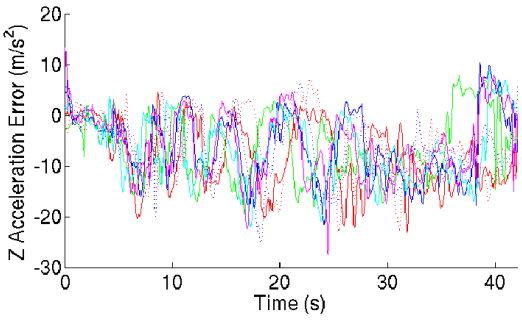
۵ - نتایج اجرای الگوریتم

همان‌طور که در شکل ۳ می‌بینیم خطوط رنگی مسیر پروازی خلبان انسانی بوده است همان‌طور که دیده میشود در هردفعه اجرا باوجود داشتن یک الگوی ثابت(مسیر پرواز دایره‌ای شکل) خلبان انسانی قادر به حفظ یک الگو و همچنین اجرای کامل و موفق نبوده است. ولی الگوریتم اجراشده برای مسیرهای مشاهده شده علاوه بر اینکه قادر به تشخیص مسیر موردنظر به درستی بوده است توانسته با توجه به قوانین پویای حاکم بر بال‌گرد اجرای کاملی هم داشته باشد.

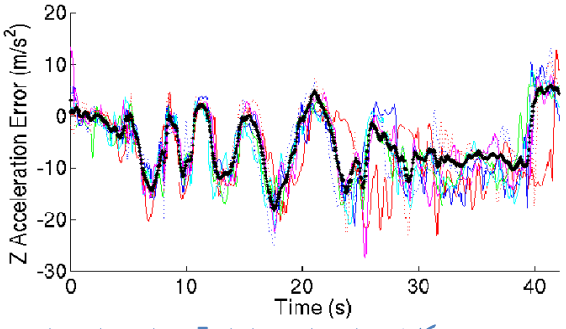


شکل ۳: خط مشکی مسیر یادگرفته شده توسط الگوریتم از مسیرهای نمایش داده شده‌ی قبلی(خطوط رنگی)

در شکل ۴ میزان تغییرات شتاب در راستای محورZ برای نمونه‌های مشاهده شده آمده است همانطور که می‌بینید یک الگوی مخفی در بین‌شان مشاهده می‌شود ولی بعد از اجرای الگوریتم و بدست آوردن شاخص هم‌ترازی زمانی و هم‌تراز کردن داده‌ها می‌بینیم که همگی در زمان های بخصوصی مقادیر بخصوصی رو اختیار می‌کنند و دلیل این امر این می‌باشد که متغییرهای پنهان بسیاری وجود دارند که انسان نمیتواند به طور دقیق مدل کند(مانند جریان هوا در اطراف ربات؛ سرعت پروانه ها و غیره.) و در هر دفعه اجرا متغییرهای پنهان تمایل دارند مقادیریکسانی با اجراهای قبلی را اختیار کنند.



شکل ۴: تغییرات شتاب در راستای Z



شکل۵: تغییرات شتاب در راستای Z- بعد ازهم ترازی زمانی

۶ - نتیجه گیری

- اگر مسیری را چندبار طی کنیم الگوی تغییرات متغییرها ثابت خواهد بود.
  - متغییرهای پنهان بسیاری وجود دارند که انسان نمیتواند به طور دقیق مدل کند.
    - جریان هوا در اطراف ربات؛ سرعت پروانه ها و غیره.
  - در هر دفعه اجرا متغییرهای پنهان تمایل دارند مقادیریکسانی با اجراهای قبلی را اختیار کنند.
- الگوریتم پیشنهادی
  - توانایی یادگیری متغییرهای پنهان را دارد.
    - همچنین شاخص هم ترازی زمانی را نیز یاد می گیرد.
  - در رباتیک برای انجام مانوروهای پیچیده و مشکل برای پیاده سازی انسانی کاربرد دارد.
    - به شرط امکان جمع آوری داده برای هر گونه رباتی کاربرد دارد.
  - برای مصارف غیررباتیک نیز کاربرد دارد.
    - برای هر مساله ای دارای متغییرهای پنهان و ناهم ترازی به شرط وجود الگوی پنهان کاربرد دارد.

۷ - منابع

[1] **Inverted autonomous helicopter flight via reinforcement learning**, Andrew Y. Ng, Adam Coates, Mark Diel, Varun Ganapathi, Jamie Schulte, Ben Tse, Eric Berger and Eric Liang. In *International Symposium on Experimental Robotics*, 2004

[2] **Learning vehicular dynamics, with application to modeling helicopters**, Pieter Abbeel, Varun Ganapathi, and Andrew Y. Ng. In *NIPS 18*, 2006.

[3] **An Application of Reinforcement Learning to Aerobatic Helicopter Flight**, Pieter Abbeel, Adam Coates,Morgan Quigley, and Andrew Y. Ng. In *NIPS 19*, 2007.

[4] **Learning for Control from Multiple Demonstrations**, Adam Coates, Pieter Abbeel, and Andrew Y. Ng. *ICML*, 2008.