First International Comprehensive Competition Conference on Engineering Sciences in Iran
September 2016
Anzali - Iran
اولین مسابقه کنفرانس بین المللی جامع علوم مهندسی در ایران
WWW.IRAN3C.COM

بهبود عملکرد محرکه های القایی با الگوریتم Q-Learning

صادق حصاری[1]، ساجده اربابی[2]

1. hesari.sadegh@yahoo.com
2. sajedeh.arbabi6486@gmail.com

چکیده

یادگیری تقویتی روشی است که در آن عامل با در نظر گرفتن حالت محیط، از بین همه اعمال ممکن ،یکی را انتخاب می کند و محیط در ازای انجام آن عمل، یک سیگنال عددی به نام پاداش به عامل باز می گرداند. هدف عامل این است که از طریق سعی و خطا سیاستی را بیابد که با دنبال کردن آن به بیشترین پاداش ممکن برسد. در این مقاله ، سعی داریم به عامل یاد بدهیم چگونه تلفات موتور القایی را کاهش بدهد. ایده اصلی ،استفاده از الگوریتم Q-Learning برای یافتن بهترین و بهینه ترین عمل در هر حالت از محیط می باشد. حالت های الگوریتم شامل گشتاور الکترومغناطیسی (Te) و سرعت موتور (wr) بوده و عمل ، جریان مغناطیسی imr می باشد.

کلمات کلیدی: یادگیری تقویتی ،الگوریتم Q-Learning ، موتور القایی ، کاهش تلفات.

۱- مقدمه

یادگیری تقویتی یعنی مسیر یابی از موقعیت به عمل به گونه ای که پاداش عددی آن عمل حداکثر باشد. در این روش به یادگیرنده گفته نمی شود که چه اقدامی بایستی انتخاب شود، بلکه او خود باید با امتحان کردن اقدام های ممکن, عمل با بالا ترین پاداش را کشف کند. یکی از روش های یادگیری تقویتی که پیاده سازی راحتی دارد ، روش Q-Learning است که در سال ۱۹۸۹ توسط واتکینز ارائه گردید[1] . این الگوریتم توسط عامل برای یادگیری از طریق تجربیات یا آموزش استفاده می شود،هر تکرار معادل با یک دوره آموزش است. هدف از آموزش ساخت مغز عامل است ، که توسط ماتریس Q نمایش داده می شود. آموزش بیشتر منجرو به ماتریس Q بهتری خواهد شد ، که می تواند توسط عامل برای حرکت در مسیر بهینه استفاده شود. بدین ترتیب با داشتن ماتریس Q ، عامل می تواند در عوض کاوش و جستجوی متعدد ، با رجوع به ماتریس حالات و انتخاب گزینه ماکزیمم ، بهترین حالت را انتخاب نماید[2-3]. علاقه به استفاده از روش های یادگیری تقویتی برای کاربرد های کنترلی روزمره به سرعت در حال رشد است به عنوان مثال در مراجع [4-12] دیده می شود. در این مقاله ما روی سیستم کنترل درایو موتور القایی متمرکز شده ایم. برای طراحی این سیستم ، این طور فرض شده که عامل از سیستم درایو هیچ اطلاعی ندارد. در مقابل آن ، اینجا ، فرض شده که عامل ، قادر به جمع آوری حالت ها و عمل های سیستم طی رفتار واقعی موتور می باشد. یادگیری از طریق تعامل با سیستم واقعی دارای مزیت هایی همچون :

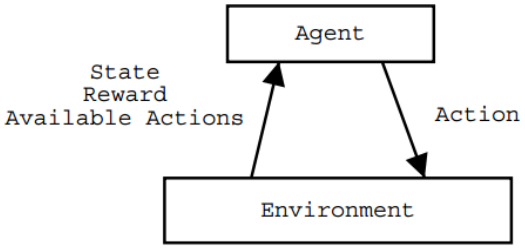
۱. احتیاجی به دانش اولیه درباره سیاست یادگیرنده ندارد. این مزیت سبب شده تا ما مقید به انجام یکسری از قانون کنترلی اولیه در مسائل کنترل کننده نباشیم. ۲. رویکرد یادگیری تقویتی در شرایطی که هیچ ایده ای در مورد قانون کنترل

کاری نیست ، نیز قابل اجراست [13] . ۳. قابلیت دیگر یادگیری تقویتی ، عدم نیاز به داده های آموزشی بوده که نسبت به مراجع [3,7,10,14] از برتری چشم گیری برخوردار است. در واقع به عامل گفته نمی شود که عمل صحیح در هر وضعیت چیست ، و فقط با استفاده از یک معیار اسکالر که سیگنال تقویتی نامیده می شود ، خوب یا بد بودن عمل به عامل نشان داده می شود. عامل موظف است با در دسترس داشتن این اطلاعات ، یاد بگیرد که بهترین عمل کدام است. این ویژگی یکی از نقطه قوتهای خاص الگوریتم یادگیری تقویتی است.

سایر بخش های این مقاله به صورت زیر می باشند. در بخش ۲ ، اساس یادگیری تقویتی و الگوریتم های q-learning و e-greedy به صورت اجمالی توضیح داده خواهد شد. در بخش ۳، مدل موتور القایی همراه با تلفات و براساس مدار معادل جریان مغناطیسی روتور بیان می شود. در بخش ۴ ، براساس مطالب بخش ۲و۳ ، به ارائه دیدگاه پیشنهادی ، استفاده از یادگیری تقویتی در کاهش تلفات موتور القایی خواهیم پرداخت. در بخش ۵ ، نتایج شبیه سازی که با نرم افزار matlab انجام شده ، مورد بررسی قرار گرفته است. بخش ۶ نیز ، حاوی بیان نتایج کلی مقاله است.

۲- یادگیری تقویتی

در یادگیری تقویتی ، هدف اصلی از یادگیری ، انجام دادن کاری و یا رسیدن به هدفی است ، بدون اینکه عامل یادگیرنده با اطلاعات مستقیم بیرونی تغذیه شود[9,11,15]. در این روش ، تنها مسیر اطلاع رسانی به عامل ، از طریق یک سیگنال پاداش و یا جریمه می باشد. در این حالت ، هدف عامل ، بیشینه کردن میزان پاداش دریافتی است که در بازه ای از زمان ، تغییر می کند. به این ترتیب، عامل (agent) نحوه ی عملکرد مناسب را با تمرکز بر پاداش دریافتی، یاد می گیرد. در شکل ۱ تعامل بین عامل و محیط در یادگیری تقویتی نشان داده شده است.



شکل ۱ : تعامل بین عامل و محیط

عامل یادگیرنده از طریق حس گرها ، توصیفی از حالت محیط اطرافش را به دست می آورد. هنگامی که عامل ، عملی را انجام می دهد ، پاداشی را دریافت می کند که می تواند بسته به خوبی یا بدی عمل ، پاداش مثبت و یا منفی باشد. رابطه ۱ ، یکی از شناخته ترین معادله q-learning در یادگیری تقویتی را نشان می دهد.

$$Q(s,a) \leftarrow (1-\alpha)\, Q(s,a) + \alpha\,[reward + \gamma\, Max_{a'}\, Q(s',a')] \qquad (1)$$

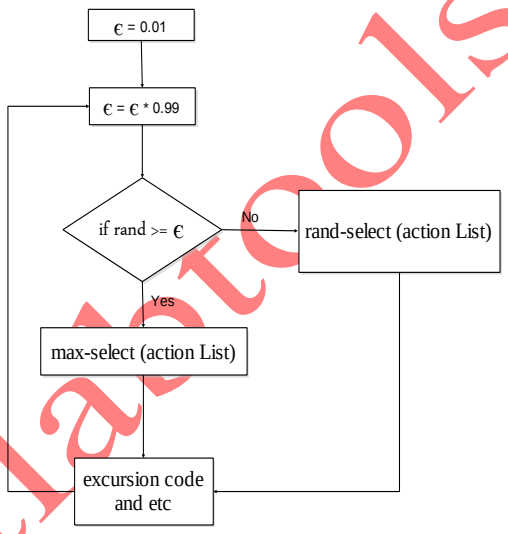
جاییکه :

α عددی بین صفر تا یک است که نرخ یادگیری (Learning Rate) نام دارد و سرعت یادگیری را تعیین می کند. بدیهی است که هرچه مقدار α بیش تر باشد، سرعت یادگیری هم بالا تر می رود ولی باید توجه داشت که مقادیر بزرگ α یادگیری را ناپایدار می کند[16-17]. در اکثر کاربرد ها معمولا" مقدار ۰,۱ برای نرخ یادگیری پیشنهاد می شود[18]. γ نیز عددی بین صفر تا یک بوده که نرخ تنزیل (Discount Factor) نام دارد و مانع از واگرایی تابع کیفیت ، حین یادگیری می

شود. مقدار γ را معمولا" برابر ۰,۹ در نظر می گیرند [18]. $Q_{max}(s', a')$ کیفیت بهینه ترین عمل در موقعیت جدید سیستم ، یعنی s' است. به عبارتی این مقدار بیشینه ی کیفیت در موقعیت s' به شمار می آید.

۲-۱- الگوریتم *e-greedy*

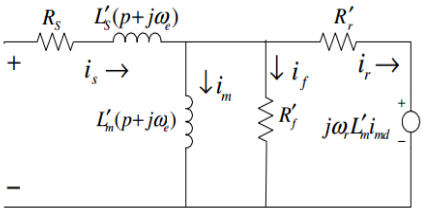
یکی از روش های انتخاب ، روشی است که ، به روش انتخاب حریصانه یا greedy مشهور است. این روش توصیه می کند ، در هر حالت ، عملی انتخاب شود که مقدار تابع ارزش بیشینه شود. روش دیگری که به نام ε–greedy مشهور است ، یک عدد کوچک ε در بازه [0,1] تعیین می شود که میزان احتمال انتخاب شدن یک عمل به شکل تصادفی را بیان می کند[17]. فرضا" اگر مقدار ε را برابر ۰,۲ در نظر بگیریم، ۰,۲ احتمال دارد که عمل به شکل تصادفی انتخاب شود و ۰,۸ احتمال دارد که عملی انتخاب شود که بیش ترین کیفیت را نسبت با سایر اعمال دارد. همچنین می توان ε را به شکل پویا تغییر داد. بدین شکل که در شروع یادگیری مقدار آن زیاد باشد که سبب می شود احتمال انتخاب تصادفی نیز زیاد باشد و با جلو رفتن یادگیری مقدار آن کاهش یابد که باعث می شود احتمال انتخاب براساس کیفیت افزایش یابد[19]. الگوریتم e-greedy به کار رفته در این مقاله در شکل ۲ دیده می شود.



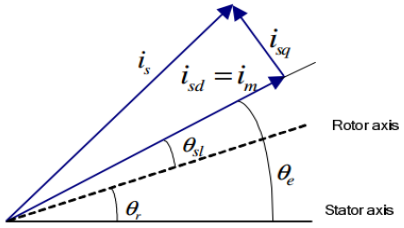
شکل ۲ : الگوریتم e-greedy

۳- مدل موتور القایی

در این مقاله ما از مدار معادلی که به جریان مغناطیسی روتور اشاره شده ، استفاده می نمائیم. یک مقاومت تلفات آهن Rf در موازی با اندوکتانس مغناطیسی در فریم مرجع شار روتور،اضافه شده که در شکل(۳) نشان داده شده است [20-23].

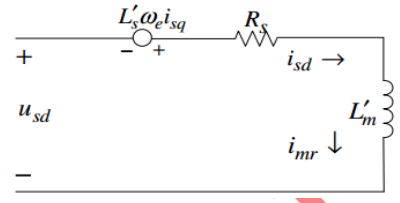


شکل (۳):مدار معادل موتور القایی شامل مقاومت تلفات آهن

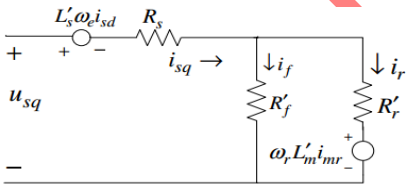


شکل(۴):دیاگرام فازوری مدارمعادل وتعریف زوایای میدان روتور

در حالت دائمی ،اندوکتانس نشتی روی موتور وجود ندارد و مدار معادل موتور مشابه شکل(۵) می شود.



الف)محورمعادل محور d



ب)مدار معادل محور q

شکل(۵):مدار معادل موتور در حالت دائمی

برای گسترش مدل تلفات،یک روش معمول و ساده در تحقیقات گذشته صورت گرفته است[24].

$$P_{total} = R_d i_{mr}^2 + R_q \frac{T_e^2}{K_t^2 i_{mr}^2} \qquad (2)$$

$$R_d = R_s + \frac{L'^2_m}{R'_f + R'_r}\omega_r^2 \quad ,R_q = R_s + \frac{R'_f R'_r}{R'_f + R'_r}.$$

همانطور که از رابطه (۲) دیده می شود ، تلفات کل موتور القایی به جریان imr ، Te و Wr نسبت داده شده است.این رابطه بیان می کند که با کنترل جریان imr موتور ، می توان تلفات موتور را کنترل نمود.

۴- الگوریتم پیشنهادی

در این روش حالتهای Te,wr را در هر گام زمانی برای موتور مشخص می کنیم و عمل imr متناسب با هر حالت را نیز معین می نماییم. سپس برای هر imr موتور ، در هر حالت یک پاداش یا تنبیه در نظر می گیریم و بر اساس رابطه ۱ به هر زوج مرتب (حالت ها Te, wr و عملimr) یک مقدار Q اختصاص داده می دهیم. موتور در مرحله یادگیری جدول Q ها را پر می

کند و در هر مرحله عمل از این جدول استفاده می کند ، یعنی در گذر از هر حالت به حالت دیگر ، جریانimr ای را انتخاب می کند که بیشترین مقدار Q را داشته باشد. الگوریتم پیشنهادی این مقاله به صورت زیر می باشد :

S=Te & wr a = imr

۱. ابتدا مقدار Q(s,a) را به ازای تمام حالات و عمل برابر صفر در نظر می گیریم.
۲. حالت فعلی سیستم (Te,wr) را به دست می آوریم.
۳. براساس الگوریتمε-greedy [17]، یک عمل (imr) را انتخاب می کنیم.
۴. عمل imr را روی موتور اعمال می کنیم و منتظر امتیاز عمل خود (r) می شویم.
۵. حالت جدید سیستم را که پس انجام عمل سیستم به آن می رود (imr') به دست می آوریم.
۶. براساس رابطه ی ۱ مقدار Q(s, a) را به روز می کنیم.

میزان امتیاز بعد از انجام عمل imr در رابطه (۲) مورد بررسی قرار می گیرد . در این مقاله پاداش دریافتی به صورت جدول ۱ در نظر گرفته شده است:

جدول ۱ : پاداش دریافتی بعد از انجام هر عمل

```
If ploss == 0     ,  reward = 100
elseif (ploss < = 10 &  ploss > 0) ,    reward =  75
elseif (ploss >10 & ploss <=30)    ,    reward = 50
elseif (ploss > 30 & ploss<=50)    ,    reward=25
else          ,  reward=0
end
```

عامل سعی میکند طوری رفتار کند که تابع پاداش را ماکزیمم نماید. برای گشتاور الکترومغناطیسی ، سرعت روتور و جریان مغناطیسی روتور داریم :

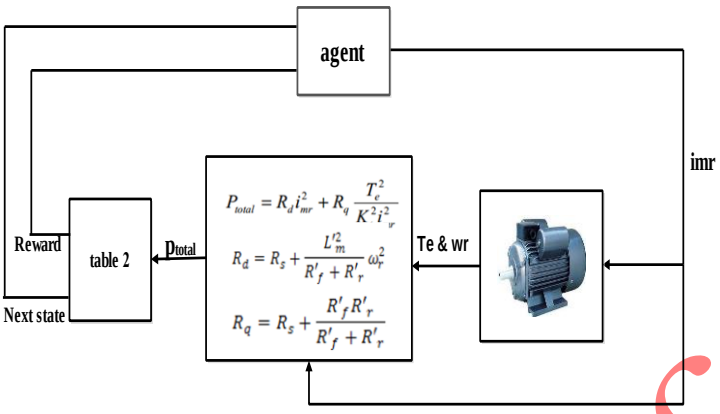
Te = 0 to 2 Nm wr = 150 imr = 0 to 5 amper

براساس موتوری که در ضمیمه ۱ ارائه شده است ، مقدار مناسب جریان مغناطیسی imr براساس مرجع [24] تنظیم شده است.

۵- نتایج شبیه سازی

پیاده سازی این مقاله در ۳ سناریو انجام شده است. در سناریو اول ، مقدار گشتاور مرجع اعمالی بین ۰ تا ۲ نیوتن متر ، سرعت ۱۰۰ رادیان بر ثانیه و جریان imr بین ۰ تا ۵ آمپر در نظر گرفته شده است (این مقدار جریان براساس مدل موتور انتخاب شده است[۲۴]) . در سناریو اول گشتاور الکترومغناطیس خروجی را به ۱۰۰۰ قسمت مجزا تقسیم نموده و جریان imr نیز به ۵۰ قسمت ، تقسیم می نمائیم. در سناریو دوم ، با الگوریتم Qlearning جدول Q را تکمیل می نماییم. سطرهای این ماتریس حالت ها و ستون های این ماتریس ، عمل ممکن (id) می باشد. در این سناریو ، عامل هیچ اطلاعی در مورد موتور ندارد ، در واقع به ازای هر جریان imr ،گشتاور خروجی و سرعت موتور محاسبه شده و در رابطه تلفات (۲) اعمال می شود (موتور در این مقاله ، محیط مساله می باشد). محیط به ازای اعمال هر جریان imr ، یک پاداش به عامل می دهد. عامل بر

اساس این پاداش نتیجه می گیرد که آیا این imr مناسب بوده یا خیر ؟. میزان این پاداش در جداول ۲ نشان داده شده است. سناریوی دوم، در شکل (۶) خلاصه شده است. سرعت موتور، ثابت برابر با مقدار سرعت حالت پایدار در نظر گرفته شده است.



شکل ۶ : سناریوی دوم (تکمیل جدول Q)

در جداول زیر برخی از نتایج مربوط به سناریوی دوم ارائه شده است. ماتریسQ در اینجا یک ماتریس ۱۰۰۰×۵۰ بدست آمده است. یعنی دارای ۱۰۰۰ حالت و ۵۰ عمل می باشد. در هر یک از جداول،عامل یادگیرنده ۲۵۰۰۰ بار از بین ۵۰ عمل موجود، عملی را انتخاب می نماید که دارای ماکزیمم مقدار Q باشد. در مقاله ε=0.1 در نظر گرفته شده است (یعنی ε=exploit , 1- Ɛ=explore).

جدول ۲-۱. محاسبه Q برای حالت Te=0 , Wr=100

action / states	imr=0	imr=0.2	imr=1	imr=2.4	imr=3	imr=4.3	imr=5
Te=0 & Wr=100	Q= 49.99	Q= 199.52	Q= 198.57	Q= 130.96	Q= 171.91	Q= 116.58	Q= 112.49

جدول ۲-۲. محاسبه Q برای حالت Te=0.8 , Wr=100

action / states	imr=0	imr=0.2	imr=1	imr=2	imr=3	imr=4.3	imr=5
Te=0.8 & Wr=100	Q= 99.00	Q= 123.95	Q= 167.91	Q= 199.98	Q= 174.60	Q= 149.62	Q= 149.41

جدول ۲-۳. محاسبه Q برای حالت Te=1.2 , Wr=100

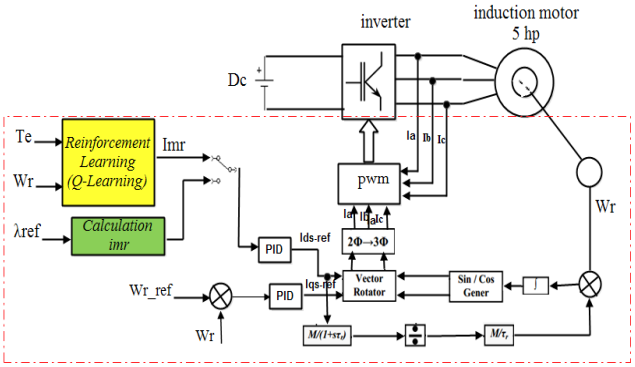
action / states	imr=0	imr=0.2	imr=1	imr=2	imr=2.5	imr=4.3	imr=5
Te=1.2 & Wr=100	Q= 124.38	Q= 69.76	Q= 132.79	Q= 148.40	Q= 149.60	Q= 122.43	Q= 122.31

جدول ۲-۴. محاسبه Q برای حالت Te=2 , Wr=100

states \ action	imr=0	imr=0.2	imr=1	imr=2.4	imr=3	imr=3.8	imr=5
Te=2 & Wr=100	Q= 69.67	Q= 79.44	Q= 138.07	Q= 143.29	Q= 145.91	Q= 149.36	Q= 122.31

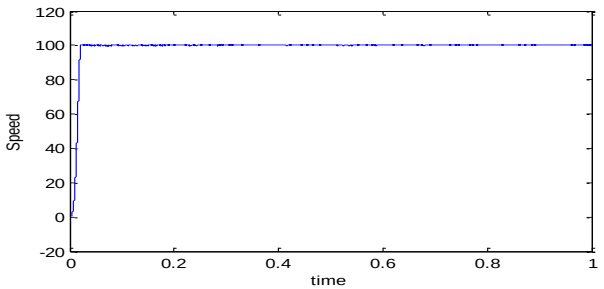
در جدول ۲-۱ ، برای حالتی که Te=0 & Wr=100 است ، بهترین جریانی که می توان برای آن به کار برد ، جریان imr=0.2 می باشد. در این حالت مقدار تلفات کل موتور برابر ۰,۰۷۳۵ بدست آمده است. Q مناسب با رنگ طوسی تمایز داده شده است. در جدول ۲-۲ ، برای حالت Te=0.8 & Wr=100 ، مناسب ترین جریان برابر imr=2 می باشد. در این حالت تلفات کل برابر ۹,۵۵۱۰ می باشد. در جدول ۲-۳ ، در حالت Te=1.2 & Wr=100 ، جریان imr=2.5 به عنوان بهترین جریان انتخاب شده است. تلفات در این حالت برابر ۱۴,۶۵۳۴ بدست آمده است. در جدول ۲-۴ نیز در حالت Te=2 & Wr=100 ، جریان imr=3.8 توسط الگوریتم q-learning انتخاب شده است. در این حالت تلفات برابر ۳۰,۳۳۷۲ می باشد.

در سناریو سوم ، با استفاده از نتایج بدست آمده از جدول Q سناریو دوم ، مدل ساختاری شکل (۷) را کامل می کنیم. در این ساختار، مقدار گشتاور و سرعت موتور هر لحظه از خروجی فیدبک گرفته شده و به ازای هر سرعت و گشتاور ، الگوریتم Q-learning ، یک جریان imr به موتور اعمال می کند. اعمال چنین جریانی به موتور ، کاهش تلفات و بهبود بازده را نتیجه خواهد داد. نتایج شبیه سازی این رویکرد در شکل های ۸ تا ۱۰ نشان داده شده است.



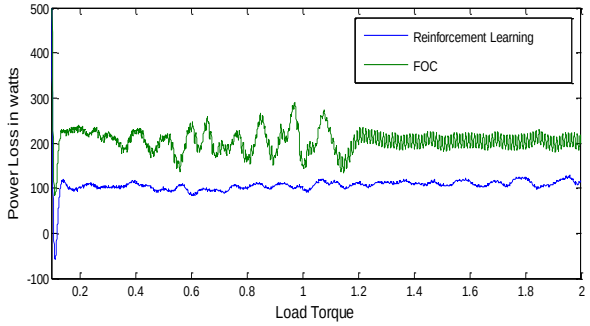
شکل ۷ : ساختار پیشنهادی (سناریوی سوم)

شکل ۸ ، سرعت خروجی موتور را نشان می دهد. از شکل می بینیم که سرعت خیلی سریع ، کمتر از ۰,۰۲ ثانیه به مقدار مرجع خود یعنی 100rad/s رسیده است.



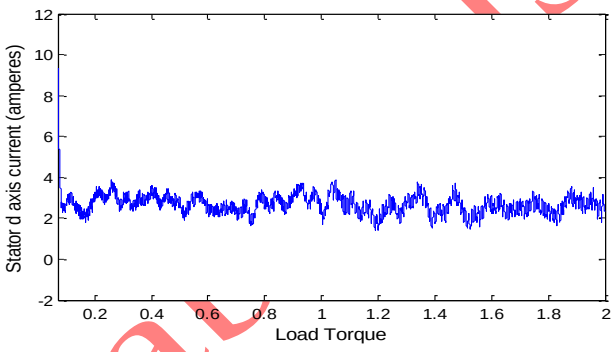
شکل ۸ : سرعت موتور

تلفات موتور با اعمال روش RL و بدون اعمال RL (FOC) در شکل ۹ نشان داده شده است.



شکل ۹ : تلفات در دو حالت FOC و RL

از نتایج شکل ۹ می توان دید ، حالتی که از رویکرد یادگیری تقویتی بهره برده ، از میزان تلفات کمتری برخوردار است. توانسته ایم تلفات را در حدود ۱۰۰ وات در گشتاور بار بین ۰ تا ۲ نیوتن متر کاهش دهیم. در حالت پایدار (دائمی موتور) ، جریان مغناطیسی روتور برابر با جریان محور d استاتور می باشد(imr = id) . متوسط جریان محور d استاتور در شکل ۱۰ دیده می شود. این جریان بین ۲ تا ۴ آمپر تغییر می کند.



شکل ۱۰ : متوسط جریان محور d استاتور

۶- نتیجه گیری

در این مقاله ، رویکردی متفاوت روی درایو موتور القایی صورت گرفت. از روش یادگیری تقویتی برای کاهش تلفات موتور القایی استفاده نمودیم. الگوریتمی که برای این منظور به کار رفت ، استفاده از الگوریتم Q-Learning بود. موتوری که برای بهینه سازی تلفات صورت گرفت،یک موتور القایی hp۵ با گشتاور ورودی بین ۰ تا ۲ نیوتن متر در نظر گرفته شد. از آنجاییکه تلفات موتور در حالت کم باری دارای بیشترین مقدار است،لذا در این مقاله الگوریتم کاهش تلفات موتور را در بازه بین ۰ تا ۲ نیوتن متر در نظر گرفتیم. سه سناریو برای کاهش تلفات موتور در این مقاله پیشنهاد شد.در سناریوی اول مدل سازی درایو انجام گرفت. در سناریوی دوم ،رویکرد Q-Learning برای تکمیل جدول بهینه Q براساس مدل سناریوی اول صورت گرفت. و در نهایت در سناریوی سوم به صورت آنلاین این جدول Q برای کاهش تلفات موتور القایی به کار رفت. همانطور که از نتایج دیده شد ، تلفات به مقدار زیادی کاهش یافت. و به این معنی است که با توان ورودی کمتری توانسته ایم این میزان گشتاور بار ورودی را تامین نماییم.

ضمیمه ۱:

موتور ۵ اسب بخار، ۴۶۰ ولت، ۱۷۵۰ دور بر دقیقه	
مقاومت استاتور: ۱،۱۱۵ اهم	مقاومت روتور: ۱،۰۸۳ اهم
اندوکتانس استاتور: ۰،۰۰۵۹ هانری	اندوکتانس روتور:۰،۰۰۵۹ هانری
اندوکتانس متقابل: ۰،۲۰۳۷ هانری	اینرسی: ۰،۰۲
تعداد جفت قطب ها: ۲	ضریب تحریک: ۰،۰۰۵۷۵۲

مراجع

[1] Christopher john Cornish hellaby Watkins , " ***learning from delayed rewards*** " , thesis submitted for ph.d. , may 1989.

[2] Damien Ernst , Mevludin Glavic, and Louis Wehenkel ," ***Power Systems Stability Control : Reinforcement Learning Framework***" , IEEE Transactions on Power Systems, (Volume:19 , Issue: 1)

[3] Stephan ten Hagen and Ben , "Neural Q-Learning" , ***Neural Computing & Applications***, 12(2):81-88, November 2003 .

[4] Hiroshi Kawano, "***Effect of Virtual Work Braking on Distributed Multi-Robot Reinforcement Learning***" , 2013 IEEE International Conference on Systems, Man, and Cybernetics.

[5] Yusuke MAEDA and Ryohei ABURATA," Teaching and Reinforcement Learning of Robotic View-Based Manipulation", 2013 IEEE RO-MAN: The 22nd IEEE International Symposium on Robot and Human Interactive Communication Gyeongju, Korea, August 26-29, 2013.

[6] Daisuke Shinohara, Takamitsu Matsubara* and Masatsugu Kidode , "Learning Motor Skills with Non-Rigid Materials by Reinforcement Learning" , Proceedings of the 2011 IEEE International Conference on Robotics and Biomimetics December 7-11, 2011, Phuket, Thailand.

[7] Zhao Jin, Wang Jianjing, Zhang Huajun, Yang Wei , "Reinforcement Learning Based Self-Constructing Fuzzy Neural Network Controller for AC Motor Drives" , 2011 6th IEEE Conference on Industrial Electronics and Applications (ICIEA).

[8] David Claveau , "Progress Towards a Humanoid Robot that Learns to Stand" , 2012 IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL).

[9] Russ T edrake, T eresa Weirui Zhang,"Learning to Walk in 20 Minutes",In Proceedings of the Fourteenth Yale Workshop on Adaptive and Learning Systems. 2005.

[10] Zhang Huajun, Zhao Jin, Wang Rui, Ma Tan , "Multi-Objective Reinforcement Learning Algorithm and Its Application in Drive System" , 34th Annual Conference of Industrial Electronics, IEEE 2008. IECON 2008.

[11] Corneliu Caileanu , "An Agent-based Approach to Induction Motor Drives Control" , International Conference on Intelligent Engineering Systems, 1997. INES '97. Proceedings., 1997 IEEE

[12] M N Howell T J Gordon and M C Best , "The Application of Continuous Action Reinforcement Learning Automata to Adaptive PID Tuning " , Learning Systems for Control (Ref. No. 2000/069), IEE Seminar , transection of IEEE , 2000.

[13] Shi-chao Wang, Zheng-xi Song, Hao Ding, Hao-bin Shi," An Improved Reinforcement Q-Learning Method with BP Neural Networks In Robot Soccer" , 2011 Fourth International Symposium on Computational Intelligence and Design , 2011 IEEE.

[14] Rong-Jong Wai ," Motion Control of Linear Induction Motor via Petri Fuzzy Neural Network", IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 54, NO. 1, FEBRUARY 2007

[15] Jason Pazis, Michail G. Lagoudakis , "Reinforcement Learning in Multidimensional Continuous Action Spaces" , Adaptive Dynamic Programming And Reinforcement Learning (ADPRL), 2011 IEEE.

[16] Wang Qiang, Zhan Zhongli," Reinforcement Learning Model,Algorithms and Its Application" , 2011 International Conference on Mechatronic Science, Electric Engineering and Computer, 2011,IEEE.

[17] FU Bo, CHEN Xin, HE Yong, WU Min , "An Efficient Reinforcement Learning Algorithm for Continuous Actions", 25th Chinese Control and Decision Conference (CCDC), IEEE,2013 .

[18] Akira Notsu, Katsuhiro Honda, Hidetomo Ichihashi, Yuki Komori," Simple Reinforcement Learning for Small-Memory Agent" , 2011 10th International Conference on Machine Learning and Applications.

[19] Richard S. Sutton and Andrew G. Barto. Reinforcement Learning: An Introduction. Bradford Books, MIT, 1998.

[20] M.Nasir Uddin,Sang Woo Nam," Adaptive backstepping online loss minimization control of an IM drive",IEEE,2007.

[21] T.R. Chelliah, J.G. Yadav," Optimal Energy Control of Induction Motor by Hybridization of Loss Model Controller Based on Particle Swarm Optimization and Search Controller ", World Congress on Nature & Biologically Inspired Computing, Pp. 1178 – 1183, 2009.

[22] Yong Tai, Zhaomian Liu," Efficiency optimization of induction Motor using genetic algorithm and hybrid genetic algorithm",2013,IEEE.

[23] M. Nasir Uddin, Sang Woo Nam," New Online Loss-Minimization-Based Control of an Induction Motor Drive", IEEE TRANSACTIONS ON POWER ELECTRONICS, VOL. 23, NO. 2, MARCH 2008.

[24] Waheeda beevi,Sukesh Kumar," loss minimization of vector controlled induction motor drive using genetic algorithm",IEEE,2012.

First International Comprehensive Competition Conference on Engineering Sciences in Iran
September 2016
Anzali - Iran
اولین مسابقه کنفرانس بین المللی جامع علوم مهندسی در ایران
WWW.IRAN3C.COM

چکیده انگلیسی

Using The Q-Learning in order to Improving the Efficiency in Drive of Induction Motors

Sadegh hesari , Sajedeh Arbabi

1. hesari.sadegh@yahoo.com
2. sajedeh.arbabi6486@gmail.com

Abstract

Reinforcement learning is a method where an agent considers the environment state, chooses one action among all the possible actions, and the environment returns a Numerical signal as a reward for that action. The agent aims at finding a policy by trial-and-error method to reach the maximum reward. In this paper, we have tried to teach the agent how to reduce the induction motor loss. The main idea is to use Q-Learning algorithm in order to find the best and optimal action in every state of the environment. The algorithm states include electromagnetic torque (Te) and motor speed (wr), and the action is imr magnetic current.

Key words: Reinforcement Learning, Q-Learning, Induction Motor,Loss Minimization.