

Anyone got Quadcopter v2 to fly?

DLND Part 6: Deep Reinforcement Learning

Project: Teach a Quadcopter How to Fly

fzawadiak 2018-04-05 10:50:35 UTC #1

Hi, this is most frustrating project so far.

For two weeks I try to get that thing to do anything better than trashing around. I tried with regularizes, batch normalization, different network sizes, small noise, large noise, very large noise, “forced” target update on achieving better number of steps etc. Tried takeoff, then switched to hover as it should be even easier (excluding “hard” start). But in all previous projects that gave some improvement, but I could see some progress even without those changes.

Only thing I get is extreme differences in rotor speeds, which is not how you fly that thing 😊
And learning tends to vary two rotors to max and other two rotors to zero.

Anyone got version 2 to fly?

Getting loss of 0 from the beginning

racersmith 2018-04-05 13:01:45 UTC #2

I’ve been able to get it to fly in more controlled conditions. I’ve changed the action space size to 1 and then set all rotors to the same speed and had success having it learn the hover task. I’ve also had some luck with curriculum learning.

I think the railed rotor speeds is probably a large local maximum that it just has a hard time getting out of. The reward changes from our noise at that point are too small to really learn from.

fzawadiak 2018-04-05 13:34:25 UTC #3

But it is impossible to hover an quadcopter with the same rotor speeds for any longer time - it is unstable. Did you start training with one shared rotor velocity and then started to make them independent? You did a instant transition or slowly enabled additional outputs?

In my reward function I have position component and angular velocity component. I thought about angle, but in the end I decided it is redundant - it is hard to stay in one place with any angular velocity at all...

racersmith 2018-04-05 13:48:36 UTC #4

In our environment, a common rotor speed will always travel in the verticle direction only. You can test this by making some static actions in the `Basic_Agent` with a common rotor speed.

I would argue that orientation is more important that angular velocity. If I told you that your roll velocity was -0.1 rad/s, how would you want the quadcopter to respond? Always try and bring the velocity to zero?

What if you are already rolled to the right by 0.05 rad? I would argue that the -0.1 rad/s velocity, in this case, is a good thing as it is bringing you back closer to level.

I'm intercepting the action before it is passed to the simulator and adjusting it per some curriculum. I then pass this adjusted action to the simulator and back to the agent. This adjusted action is stored in memory and used for training. So, in this way you can force it to learn particular things.

fzawadiak 2018-04-05 14:51:43 UTC #5

Hmm... What rotor speed was needed to hover?

clindauerbu3 2018-04-05 22:44:00 UTC #6

around 410 to 430 is hovering, and I got it to fly, I actually graduated now yay 😊

An easy way to test is set `action_low` to the same value as `action_high` and see what it does, this will also help you test noise and know that the simulation and network is working right. Basically the old strategy of eliminating variables to debug something. Then when you get everything else working (all rotors going makes it fly up, noise is in desire range, rewards function is returning what you expect, and you have enough output tools to understand all this is happening) you can set your `action_low` and `action_high` variables to something more broad to make sure your AI is doing it's job and actually learning and start adjusting learning rate and tau etc.

racersmith 2018-04-05 23:12:31 UTC #7

@clindauerbu3 Congratulations!

yaohua.zhanghs9zq 2018-04-07 15:14:07 UTC #8

Hi @clindauerbu31d , I set `action_low` and `action_high` the same and I find my noise is very small. For example, my action high = action low == 300 and the actions with noise is between 299 and 301.

My question is whether my noise is good? Thanks!

```
[ 300.16152647]
[ 300.16152647]
[ 300.16152647]
[ 300.16152647]
lower_bounds
lower_bounds
Episode =      5total_reward -71.9266707807 total_actions  14
[[ 299.90166693]
 [ 299.90166693]
 [ 299.90166693]
 [ 299.90166693]
 [ 300.05503482]
 [ 300.05503482]
 [ 300.05503482]
 [ 300.05503482]
 [ 300.25804806]
 [ 300.25804806]
 [ 300.25804806]
 [ 300.25804806]
 [ 300.25804806]
```

fzawadiak 2018-04-08 12:47:37 UTC #9

My drone, when all rotors had the same speed, was slowly going down at 405rpm and slowly going up at 406rpm. So 1rpm noise is quite a lot. I tried with "flight instructor" code that was clipping actions to a small

range, that was increasing with episode number. When range got to 2rpm it was already starting to tumble fast.

[clindauerbu3](#) 2018-04-10 23:24:15 UTC #10

My noise was smaller than that I found that 1 revolution per second variance could actually make things unstable too. You can actually zero out the noise for testing by setting all the noise variables to 0. Then when you see things work gradually increase noise to help it explore more. I'm not sure this would be an advisable strat with bigger tasks but it did help me.

[yaohua.zhanghs9zq](#) 2018-04-11 04:33:51 UTC #11

Thanks! I've passed this project.

[permanaharianja](#) 2018-04-11 13:21:46 UTC #12

I found this final project is really unclear, could you please give me the outline of where to start, what is expected reward and maybe problems that one may have?

Any information is really appreciated.
Anggi Permana Harianja

[No progress on Quadcopter Hover task or Mountain Car](#)

[yaohua.zhanghs9zq](#) 2018-04-11 15:47:48 UTC #13

There are many helpful suggestions from mentors, but I think my experience can give you an idea about what you will encounter along the way.

Firstly, It's very important to understand what's DDPG.

I tried the sample code written in Keras. When I read the sample code, I started to learn what's the architecture of this project, but the point is the sample code in Keras is only a reference. it doesn't give the right hyper-parameters and network architecture. Anyway, I still recommend to put together codes in Keras as it shows what DDPG is. I also suggest you to read the book of Reinforcement Learning and do all the mini project in the course.

As I am not very familiar with Keras, I am not confident my network and its hyperparameters are correct. Later, when I found an article about sample code in TF learn, I decided to change to TF learn. TF learn provides low level api so that I can manage many details. Additionally, the sample code in TF learn is complete. It took me a while to change my code from Keras to TF learn. But when I trained my model, it still didn't learn.

I tried almost everything and it was really frustrating. My original task was hovering, but I was told taking off is easier, so I changed my task. The easier task was still too hard to me. When I found an article in this forum that provided some advise, I always couldn't replicate the success learning. I thought it was very difficult to train the model and almost gave up.

My agent started to fly after I found a new reward function, but I think the whole system is an integrated puzzle. One cohort told in this forum that she/he used PoE(process of elimination) to debug. I totally agree

with it.

hidehiro98 2018-04-25 03:11:58 UTC #14

My reviewer sent this article to me and this helps a lot for understanding what DDPG is.

<http://pemami4911.github.io/blog/2016/08/21/ddpg-rl.html>
