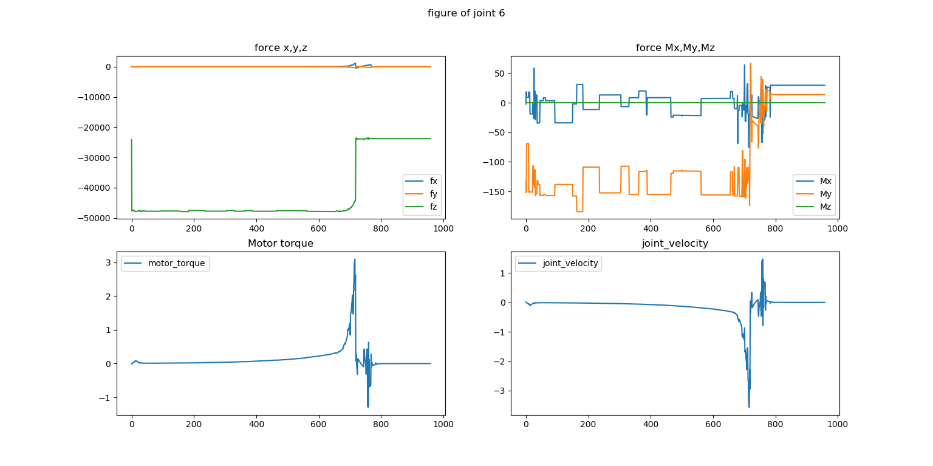
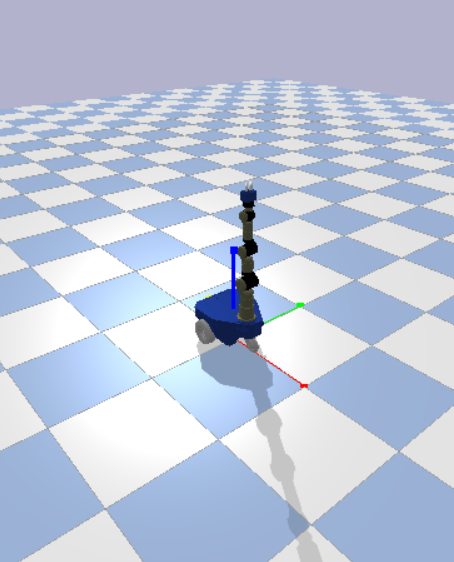
1. model problems:

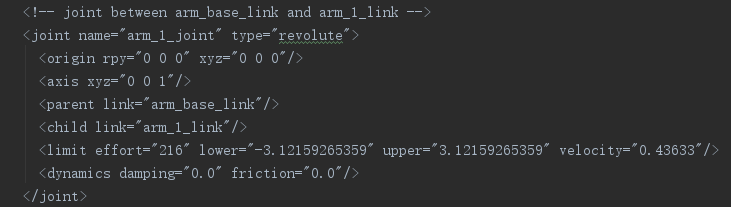
1）unsuspected high value of fz at joint 6 between arm1 and arm podest



guess, the problem is in the model

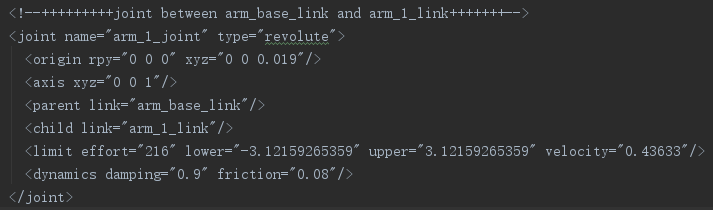
\*这里也可以发现过大的fz在一开始可能会直接导致一次抬高，猜测是mesh表面非完全平面，导致fz通过摩擦力转化为侧向力

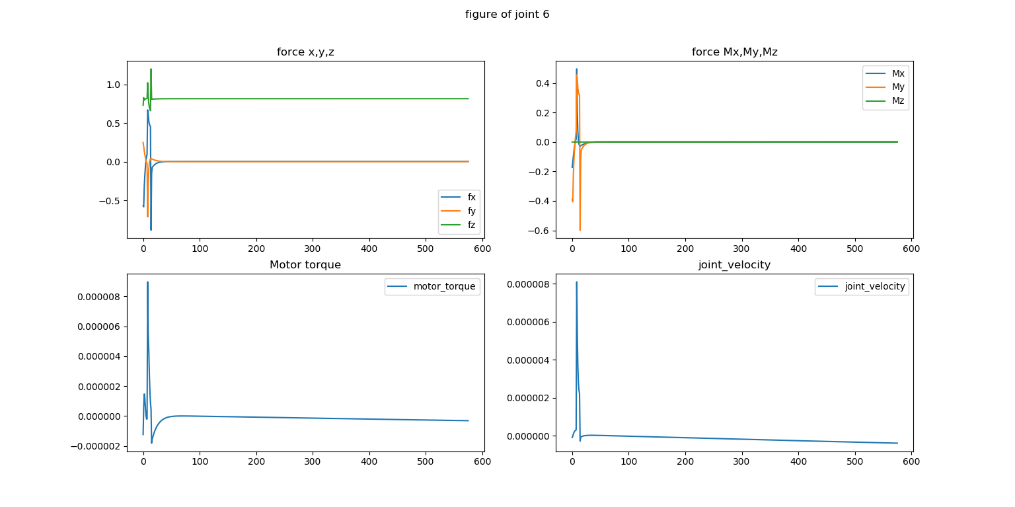
Cause: collition between arm1 and arm podest



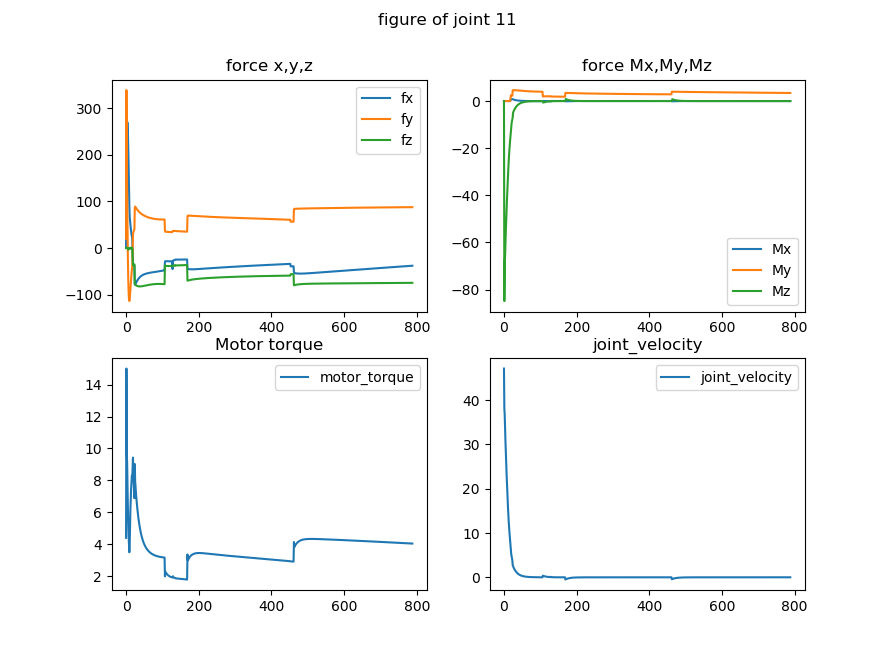
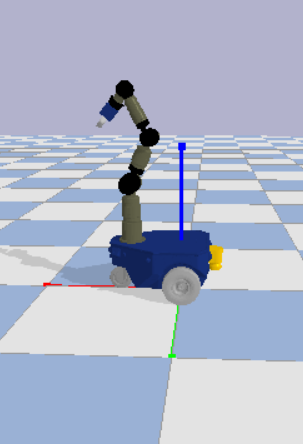
Solution: raise the Link “arm1”

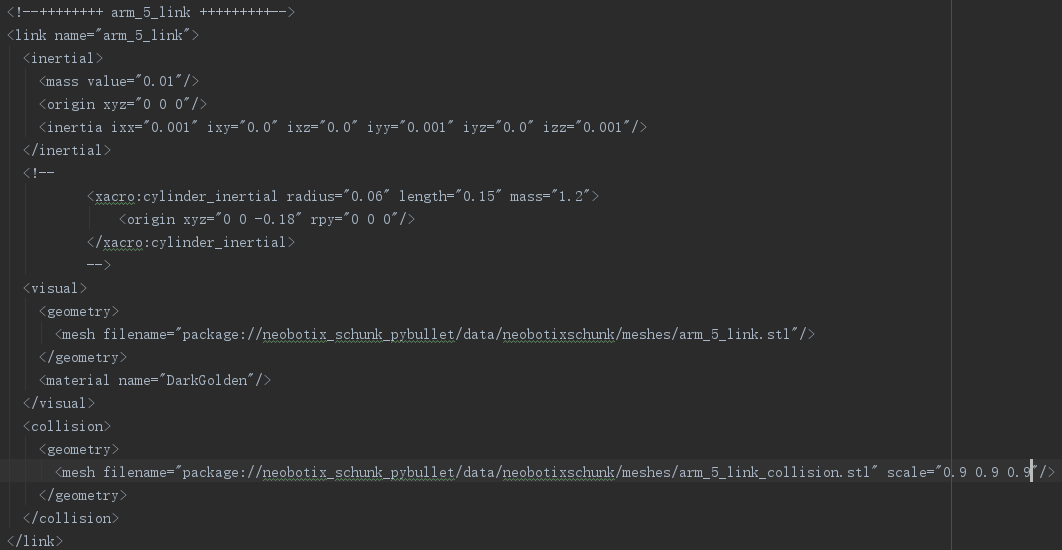
Result:





2）unsuspected high joint force when joint 11 rotates about 1.8-2.0 degree:

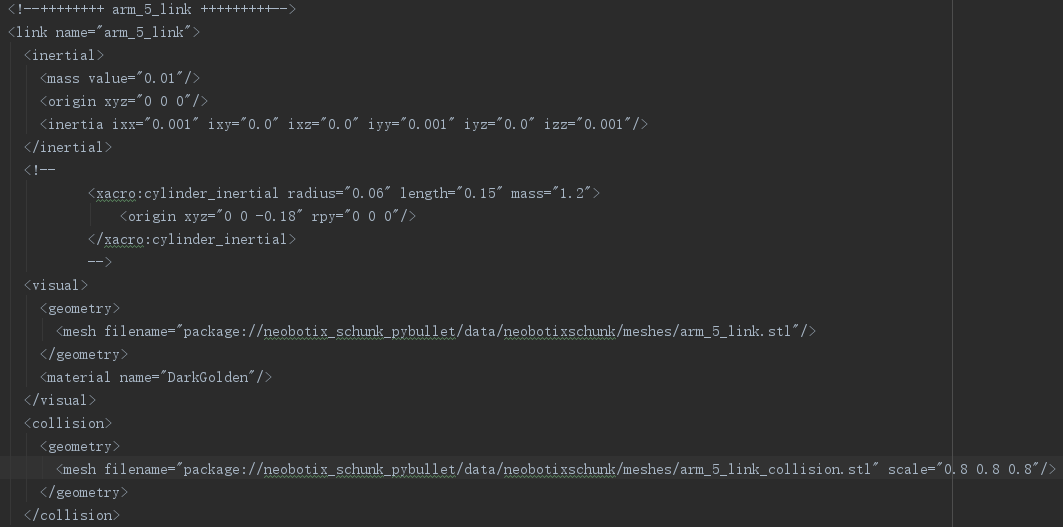


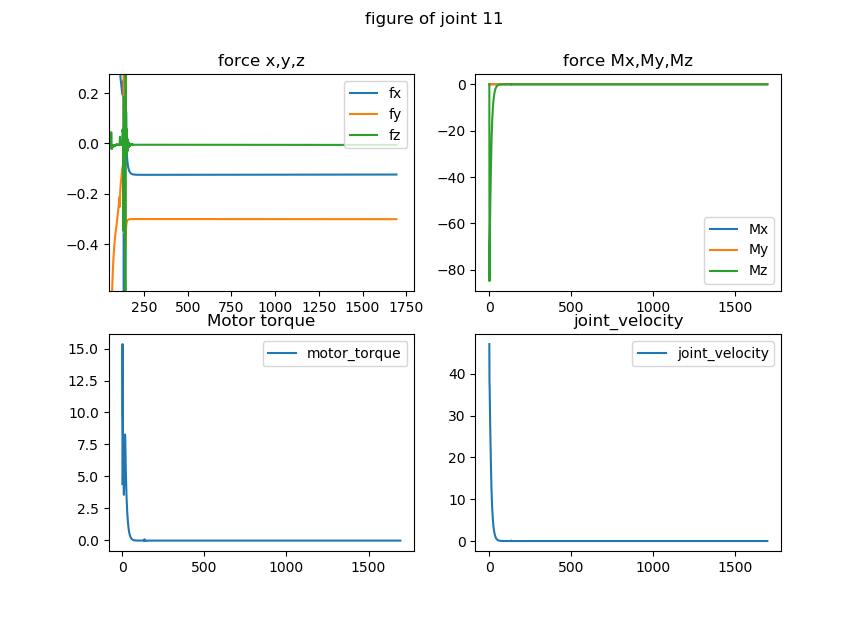
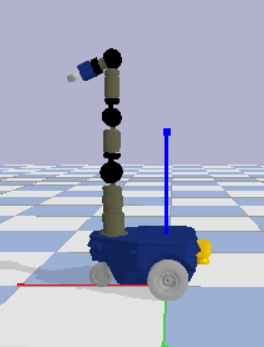


cause: collision between arm5 and arm6 or arm5 and arm7

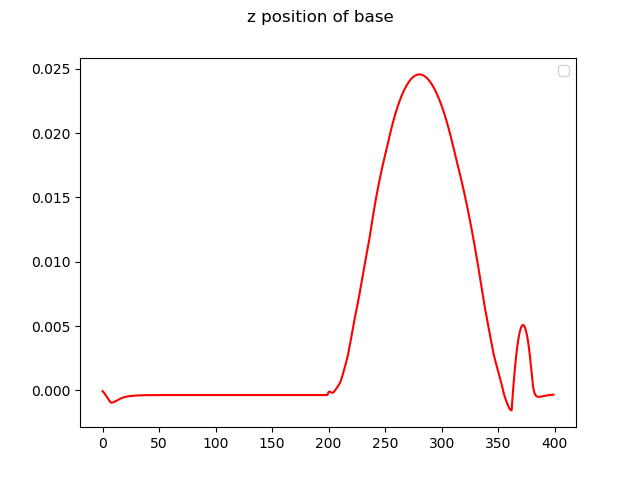
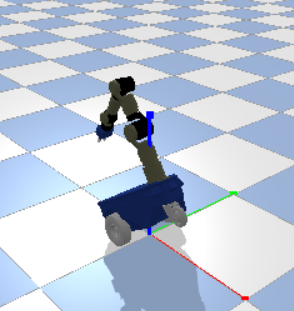
solution: decrease the collision shape of arm5

result:

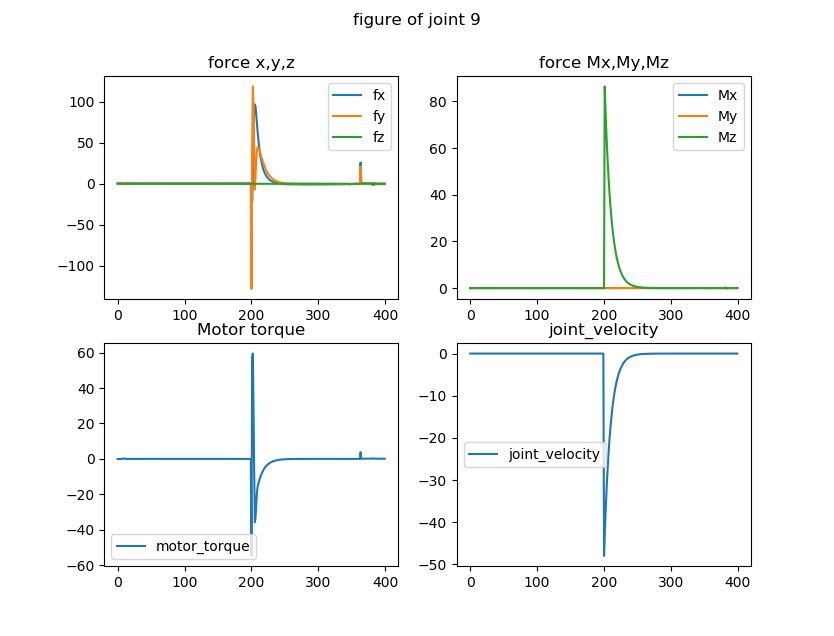




1. Raising of the base



Methode: observe the joint forces and joint speed



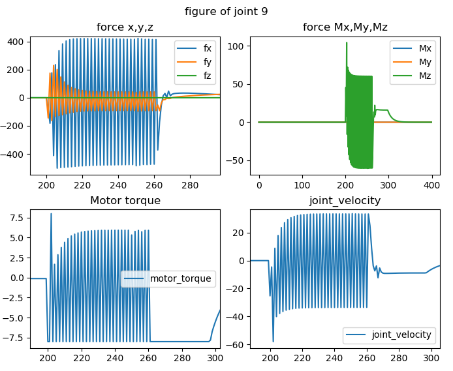
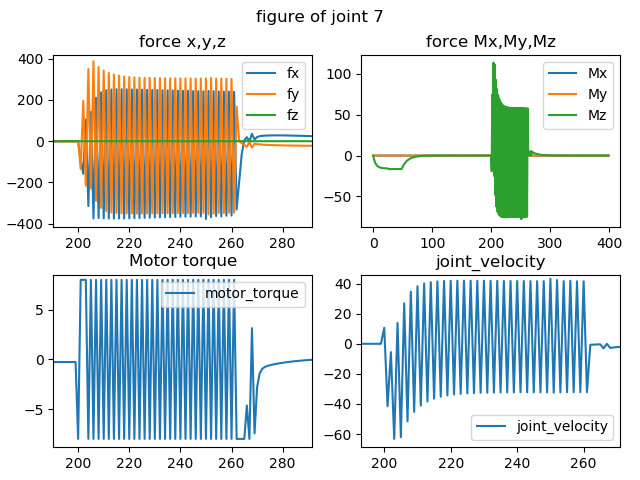
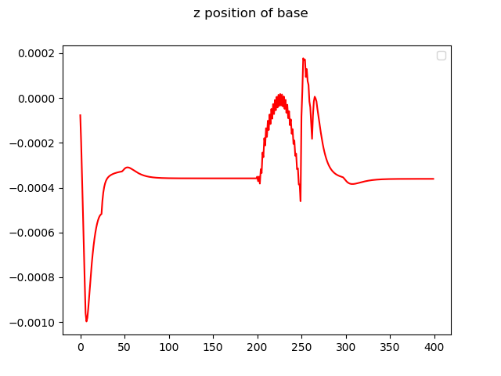
analyse: obviously, because of the unlimited motor torque and the controlling strategy of pybullet. The robot always uses the maximal torque to accelerate and then rapidly slows down, which causes a very big -fy and then a very big fy. And that also causes a huge maximal joint velocity, which lead to a huge Centrifugal force fx. These two forces fx and fy generate a huge Moment to the hinter wheels and causes the raising of base.

Solution 1: limit the maximal joint Torque

Problem: every joint should have different maximal Torque, according to the distance end effector.

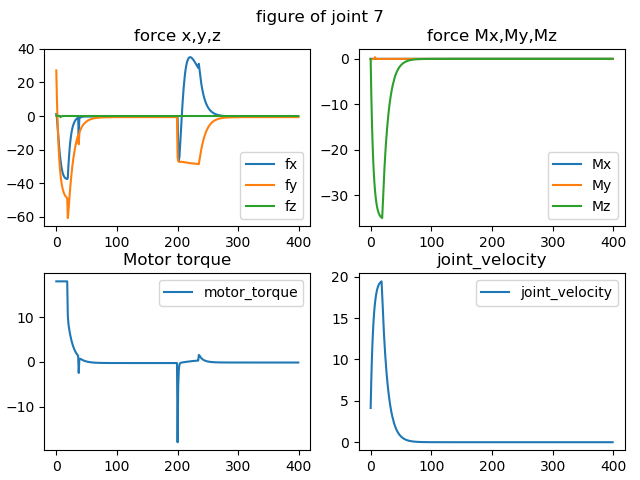
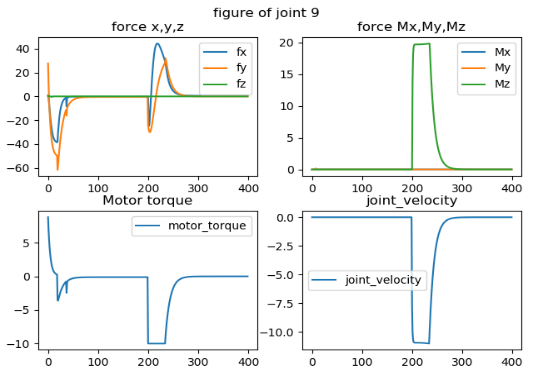
For example: set joint 7 and 9 have the maximal Torque == 8

here we can see, when joint 9 begin to use negative torque to reach angel=-2.0, joint 7 uses a negative torque at mean time to stay at same position. But because the moment to joint 7 is bigger than joint 7, joint 7 also starts to move. And because of damping and friction, the torques begin oscillate until both arrive target position.



solution: set a bigger limit of torque of joint7 than joint 9

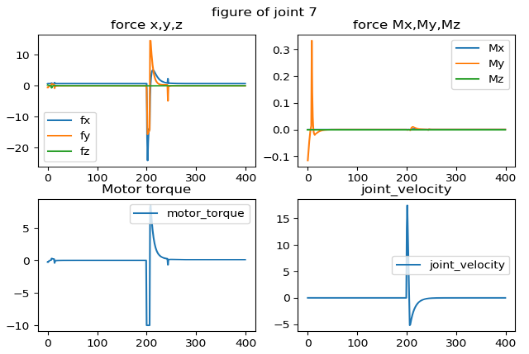
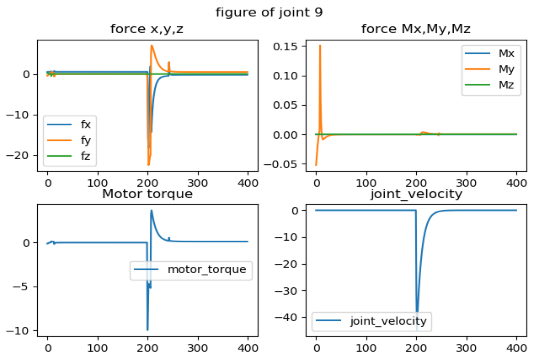
result (joint 9—torque 10, joint 7 –torque18):



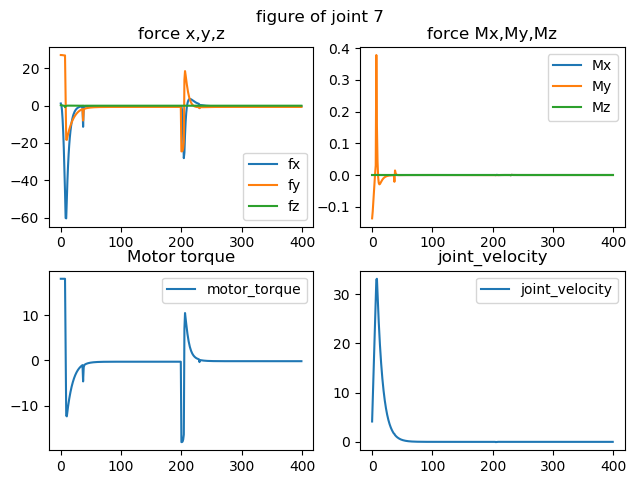
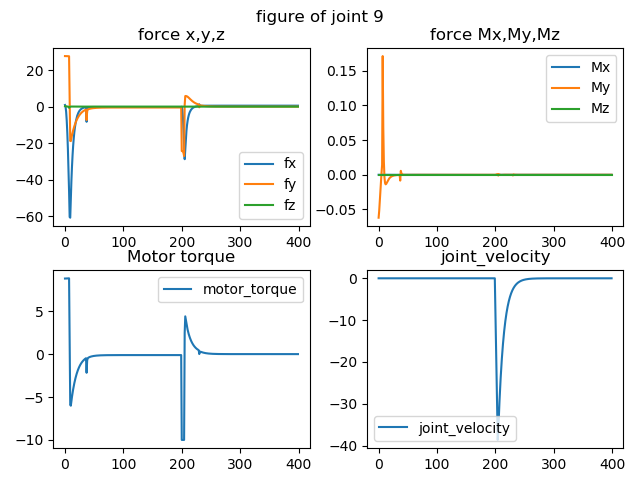
here we can see, because limit of joint7 is bigger than joint9, joint9 doesn’t start to move, only joint9 moves to target position as normal.

test without damping and friction:

1. joint9 limit 10, joint7 limit 10



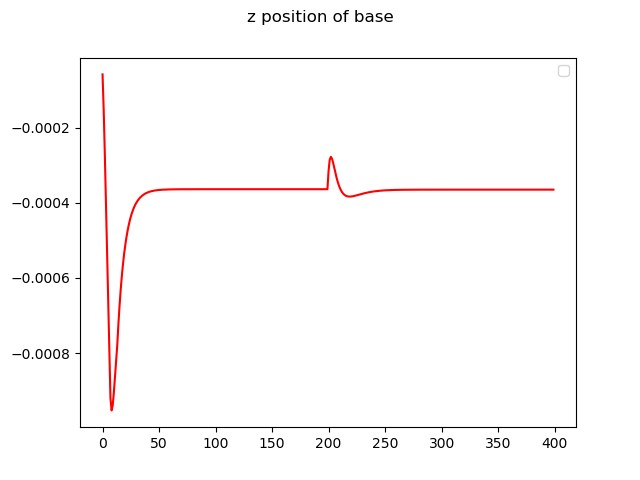
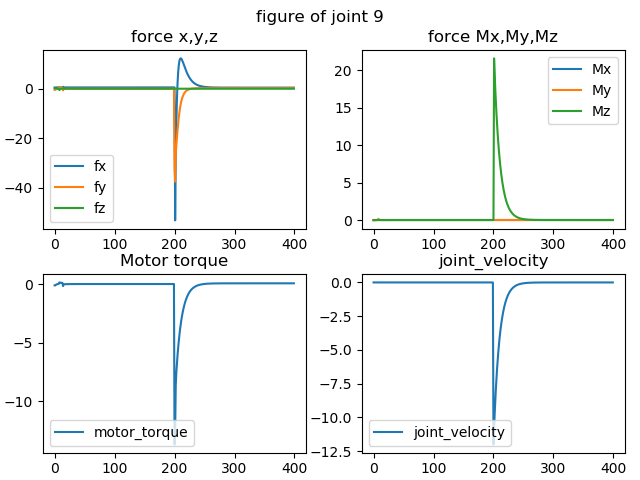
2—joint9 limit 10, joint7 limit 18



Solution 2: limit the angle increasing/difference

As we said before, this problem was also causes by the control strategy of pybullet. It always try to use a torque as big as possible to accelerate, and then slow down with another big torque. And how big the torques are depends not only on the torque limitation, but also should on the difference between target position and current position.

Test the guess: joint9 limit 50, start at 0 ,target -0.5



the maximal velocity is much smaller than before, the forces too

Discussion: If we use solution 1, we have to define individual maximal Torque for all the joints. It’s impractical in reality and not extendable (e.g. add a joint at the end, all the limitations should be redefined). Solution2 seems better, we can choose a maximal position difference for every joint at all action steps. And actually 0.5 degree in one step is not small, it’s maybe more practical than solution1. So finally, we decide to use solution2 to solve the problem of raising.