Kinematics, Dynamics and Control for Robots

Assignment 2

Maksim Filipenko
m.filipenko@innopolis.ru

I. Dynamic of 2-link manipulator (a)

Matlab code - a2_a.m.

Dynamic of the system was implemented according to the lecture node.

$$\theta(t) = \left(\theta_1(t), \theta_2(t)\right)$$

$$\tau(t) = \left(\tau_1(t), \tau_2(t)\right)$$

$$\theta(t) = dynamic\left(\tau(t), \theta(t_0), \dot{\theta}(t_0)\right)$$

Animation of the system was designed to visualize system states.

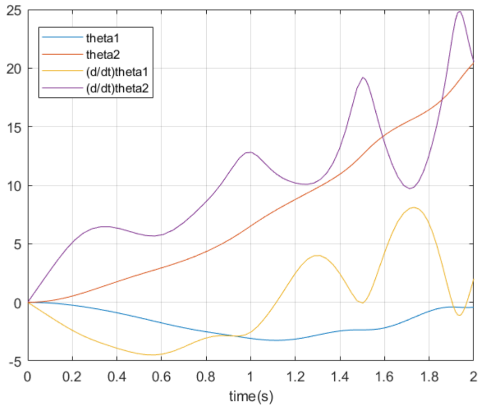

Fig. 1. Solving the dynamic problem for the system. System configuration is $\tau(t) = (10, 10), \theta(t_0) = (0, 0), \dot{\theta}(t_0) = (0, 0), \Delta t = [0, 2]$.

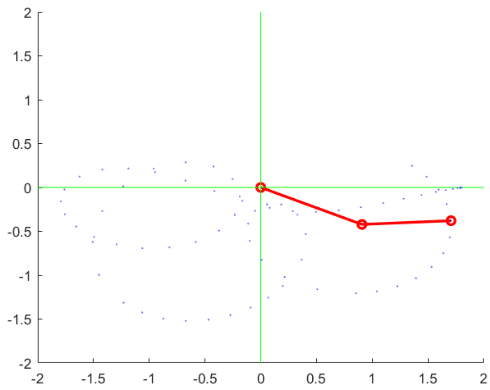

Fig. 2. Visualization of the two-link articulated robot manipulator. Blue dots correspond trajectory of the end-effector.

II. Straight motion from A to B (b)

Matlab code - a2_b_c_d_f.m

The motion of end-effector was parametrized by time based on constraints on position and speed.

Trajectories of the joint angles were found solving the inverse kinematic problem.

$$\theta(t) = inveseKinematic(X(t))$$

We have two solutions for the system. It is possible to achieve desired motion use any of this. Take first.

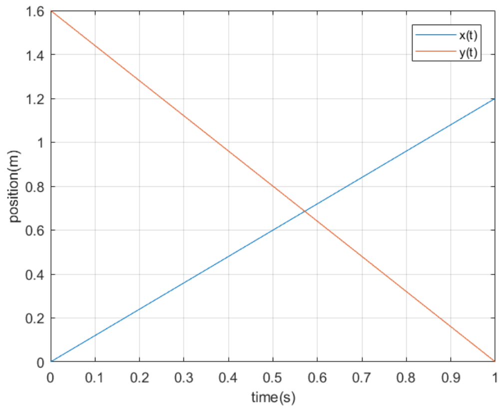

Fig. 3. The desired trajectory of end-effector $X(t)=(x(t), y(t))$.

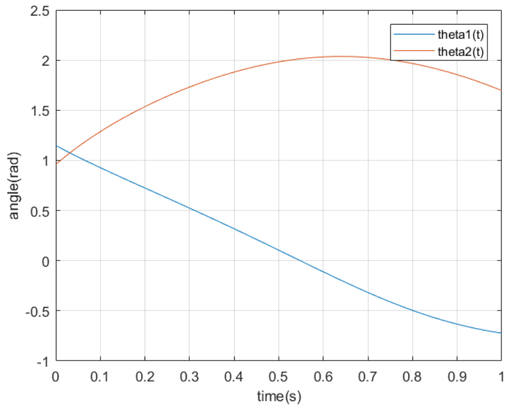

Fig. 4. The trajectory of the joint angles for the desired trajectory of end-effector.

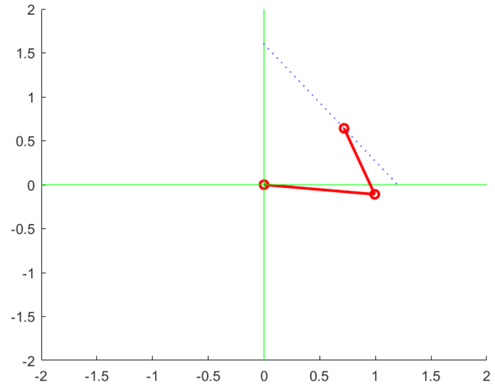

Fig. 5. Visualization of the two-link articulated robot manipulator. Blue dots correspond trajectory of the end-effector. Desired motion of robot (without dynamic).

It requires first and second derivative of the join angle to solve the inverse dynamic problem according to the dynamic model. The numerical derivative was applied (one/two times) to find this.

$$\dot{f}(t_i) = \frac{f(t_i) - f(t_{i-1})}{\vartriangle t} + O(\vartriangle t)$$
$$\vartriangle t = t_i - t_{i-1}$$

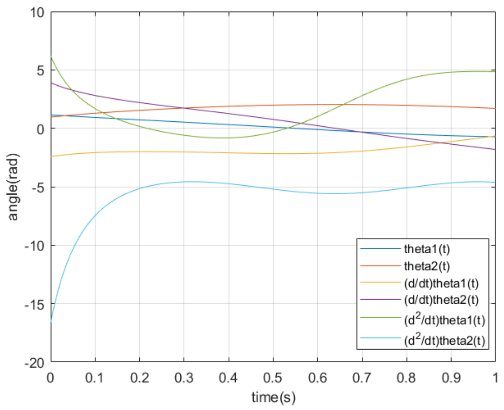

Fig. 6. Trajectory and derivatives of the joint angles for desired trajectory of end-effector

After this, we can solve the inverse dynamic problem. Solution corresponds to the torques commands to achieve desired trajectory of the end-effector.

$$\tau = inverseDynamic(\ddot{\theta}(t), \dot{\theta}(t), \theta(t))$$

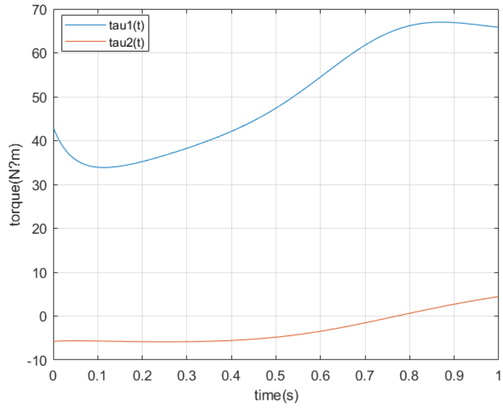

Fig. 7. Computed torques trajectory to achieve desired trajectory of the end-effector.

III. Straight Motion from A to B. (C)

Matlab code - a2_b_c_d_f.m

Solve dynamic problem which was found in subtask a with computed torques in subtask b. We can observe small error (compare with maximum value, about 0.01). It can be improved using numerical derivatives more order precision or using another time grid.

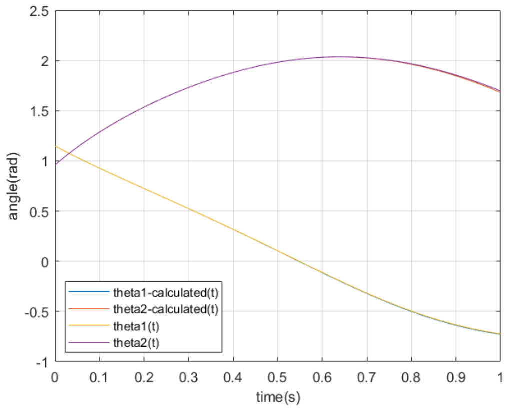

Fig. 8. Computed and desired trajectory of the joint angles.

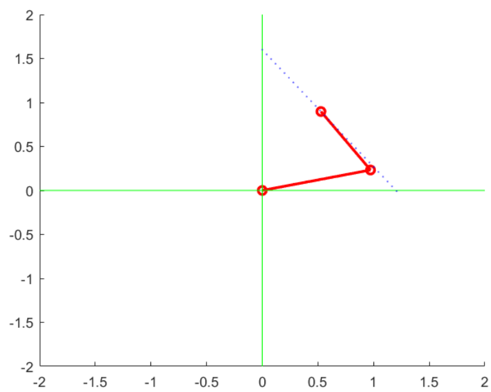

Fig. 9. Visualization of the two-link articulated robot manipulator. Blue dots correspond trajectory of the end-effector. Joint angles were taken from the dynamic model.

IV. TORQUE LIMIT (D)

Matlab code - a2_b_c_d_f.m

We can observe a significant deviation from the desired trajectory after applying maximum torque limit. It is mean that given torques are not enough to hold system on desired trajectory due to gravitation force. This example illustrates the situation when the system has some malfunction of motors.

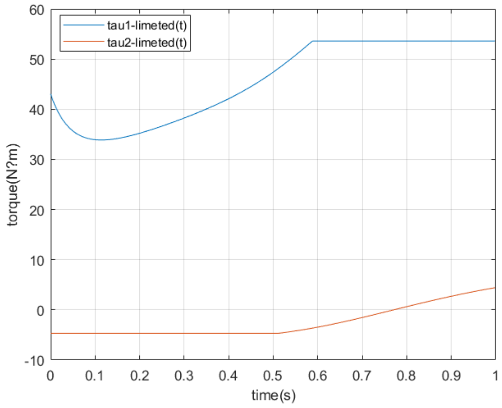

Fig. 10. Computed torques trajectory to achieve desired trajectory of the end-effector with limited maximum value.

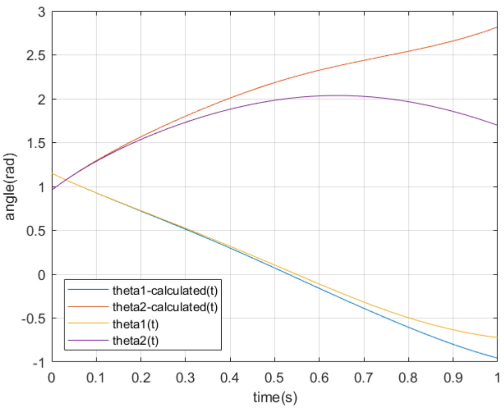

Fig. 11. Computed and desired trajectory of the joint angles.

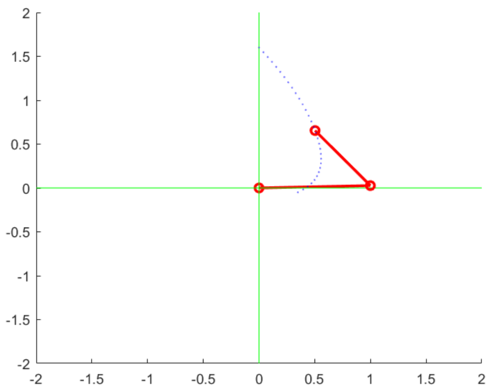

Fig. 12. Visualization of the two-link articulated robot manipulator. Blue dots correspond trajectory of the end-effector. Joint angles were taken from the dynamic model.

V. MOTION WITH ACCELERATION FROM A TO B (E)

Matlab code - a2_e_g.m.

We take parametrization of the line from A to B by distance traveled. We found desired trajectory of end-effector in Cartesian coordinates and after this in joint space.

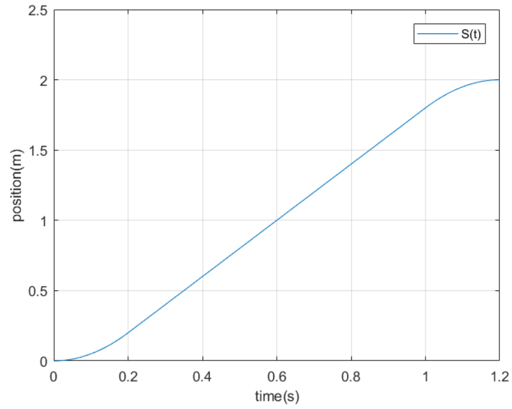

Fig. 13. Distance traveled from point A in direction of point B.

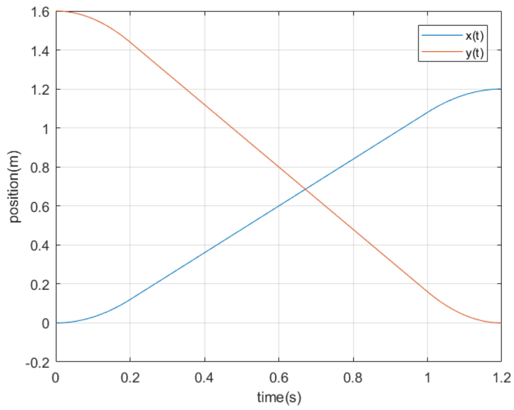

Fig. 14. Desired trajectory of end-effector $X(t) = \left(x(S(t)), y(S(t))\right)$.

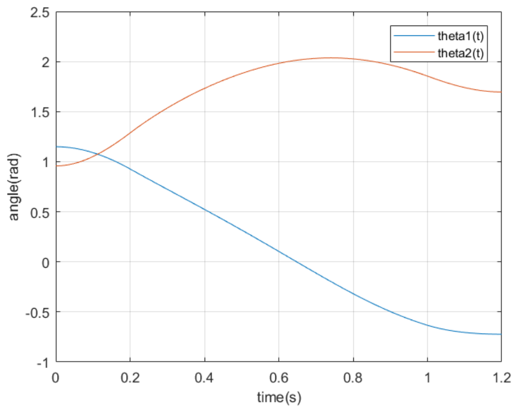

Fig. 15. The trajectory of the joint angles for the desired trajectory of end-effector.

Derivatives were computed numerically. Derivatives have breakpoints because of the piecewise prescribed law of motion.

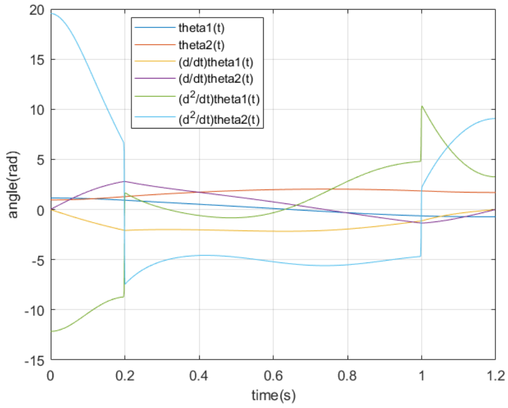

Fig. 16. Trajectory and derivatives of the joint angles for desired trajectory of end-effector

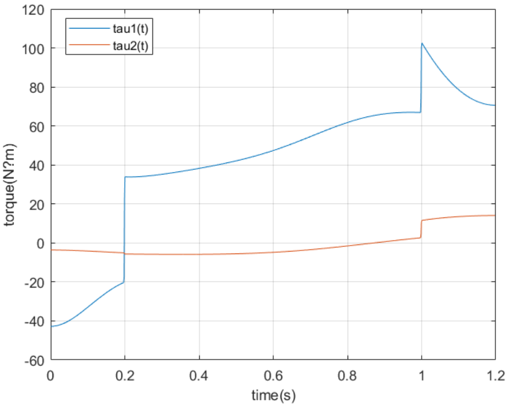

Fig. 17. Computed torques trajectory to achieve desired trajectory of the end-effector.

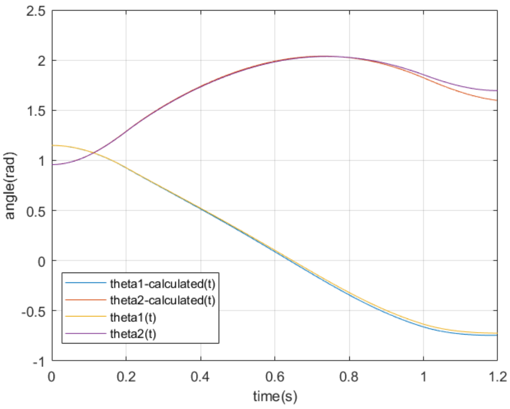

Fig. 18. Computed and desired trajectory of the joint angles.

We can observe more significant error (compare with motion with constant speed, about 0.01). Part of the error was generated by the numerical approach, which we use. However, another part probably connected with high nonlinearity and breakpoints of the torques trajectory. At this example, we have the maximum value of torques about 100, which is more than it was in motion with constant speed (maximum was about 70).

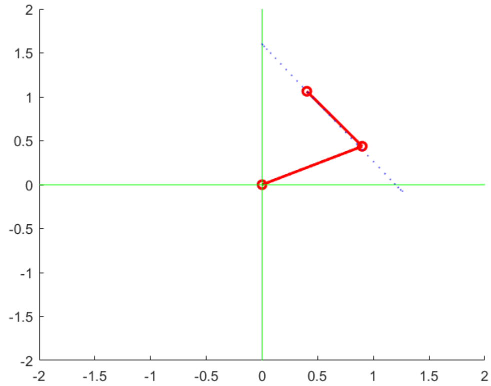

Fig. 19. Visualization of the two-link articulated robot manipulator. Blue dots correspond trajectory of the end-effector. Joint angles were taken from the dynamic model.

VI. PD CONTROL (F)

The goal is found appropriate constant based on the transient process with an optimum value of rising time and overshoot.

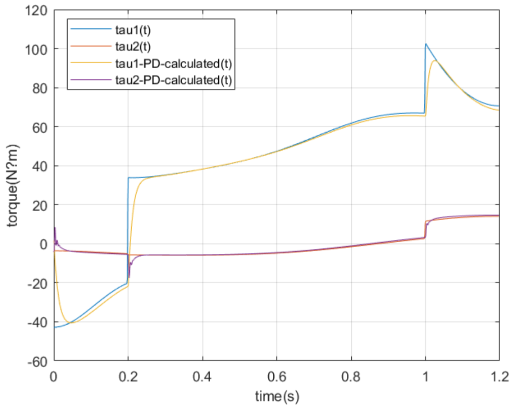

Fig. 20. Computed torques trajectory to achieve desired trajectory of the end-effector and PD controller generated torques trajectory with parameters $(K_{p1} = -100, K_{d1} = -500, K_{p1} = -100, K_{d1} = -500)$.

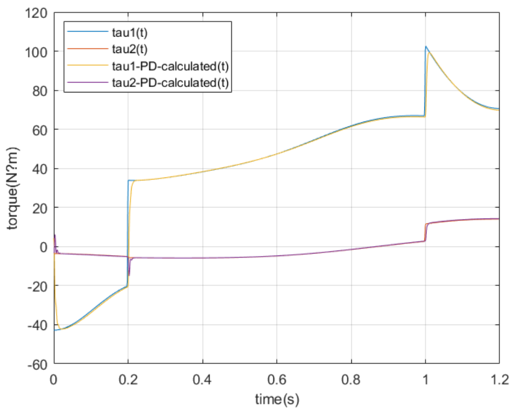

Fig. 21. Computed torques trajectory to achieve desired trajectory of the end-effector and PD controller generated torques trajectory with parameters $(K_{p1} = -200, K_{d1} = -1500, K_{p1} = -100, K_{d1} = -450)$.

VII. PD CONTROL (G)

Matlab code - a2_e_g.m.

PD control gives wagging trajectory of the joint angle around desired trajectory. PD control reduces the maximum error (compare with the case without PD control).

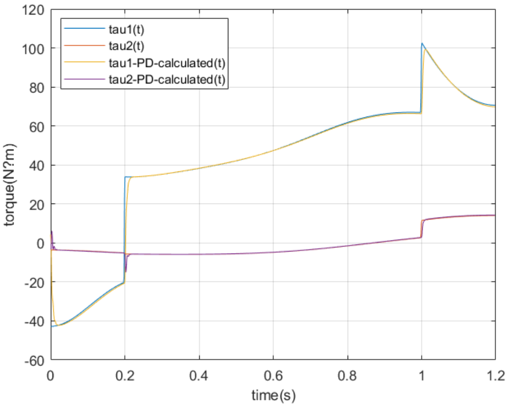

Fig. 22. Computed torques trajectory to achieve desired trajectory of the end-effector and PD controller generated torques trajectory with parameters $(K_{p1} = -200, K_{d1} = -1500, K_{p1} = -100, K_{d1} = -450)$.

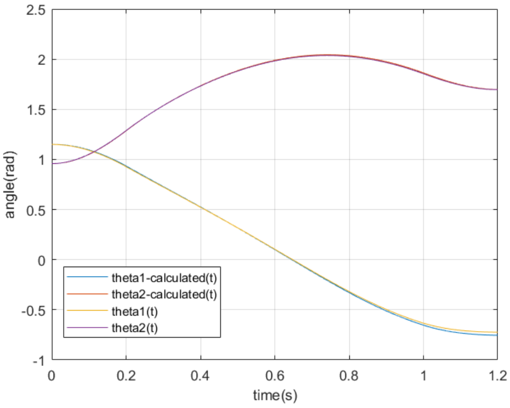

Fig. 23. Computed by PD-control and desired trajectory of the joint angles.

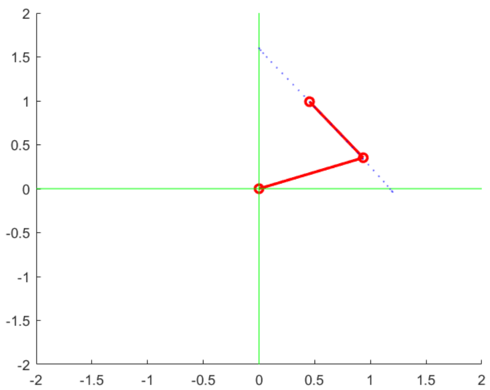

Fig. 24. Visualization of the two-link articulated robot manipulator. Blue dots correspond trajectory of the end-effector. Joint angles were taken from the dynamic model.