Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Факультет систем управления и робототехники

Отчет по лабораторной работе №5

«Управление роботом с дифференциальным приводом»

по дисциплине «Введение в профессиональную деятельность»

Выполнили: студенты группы R3136

Мартынова А.А.

Мангутов Н.Р.

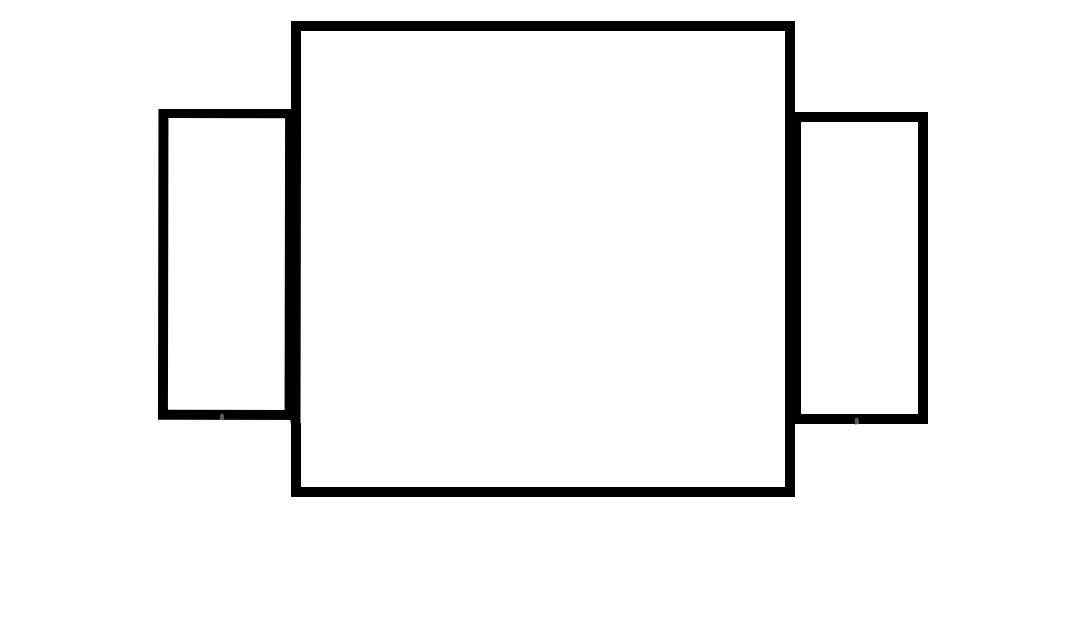
Шахтаров И.Р.

Корякина Е.Н.

Преподаватель: Перегудин А.А.

Санкт-Петербург

2019

1. Цель работы  
   Получить опыт построения математической модели робота, освоить алгоритм движения робота с дифференциальным приводом к заданной точке.
2. Ход работы  
   К сожалению, фотографий робота не сохранилось по техническим причинам, но ниже приведен его примерный чертеж.  
     
   
3. Программа для EV3  
     
   #!/usr/bin/python3

from ev3dev.ev3 import \*

from array import \*

import time

import math

mLeft = LargeMotor('outC')

mRight = LargeMotor('outD')

coord\_x = [1,1,-1,-1,1]

coord\_y = [1,-1,-1,1,1]

err\_d = 0.005

Rad = 0.0275

Brad = 0.17

x = 0

y = 0

angle = 0

start\_time = 0

cur\_time = 0

prev\_left\_position = 0

prev\_right\_position = 0

to\_rad = 3.14/180

to\_grad = 180/3.14

Ks = 250

Kr = 70

i = 0

fh = open('data.txt', 'w')

fh.write('0' + ' 0' + '\n')

def run\_motors(x1, y1):

global x, y, angle, prev\_left\_position, prev\_right\_position

start\_time = time()

while math.sqrt(math.pow((x1 - x),2) + math.pow((y1 - y),2)) > END\_DIST:

cur\_time = time()

lpsi = (prev\_left\_position - mLeft)\*to\_rad

rpsi = (prev\_right\_position - mRight)\*to\_rad

theta = (lpsi - rpsi)\*Rad/Brad

x = x + cos(angle) \* (lpsi + rpsi) / 2 \* Rad

y = y + sin(angle) \* (lpsi + rpsi) / 2 \* Rad

angle = angle + theta

fh.write((str)x + ‘ ’ + (str)y)

Us = Ks \* math.sqrt(math.pow((x1 - x),2) + math.pow((y1 - y),2))

u = math.atan2(y1 - y, x1 - x) - angle

if (math.fabs(u) > pi):

u = u - copysign(2\*pi, u)

Ur = Kr \* u

prev\_left\_position = mLeft.position

prev\_right\_position = mRight.position

if math.fabs(Ur)>20: Ur = copysign(Ur,20)

if math.fabs(Us)>80: Us = copysign(Us,80)

mRight.run\_direct(duty\_cycle\_sp = Us+Ur)

mLeft.run\_direct(duty\_cycle\_sp = Us-Ur)

for i in range(0, len(coord\_x)-1):

run\_motors(coord\_x[i], coord\_y[i])

mRight.stop(stop\_action = 'brake')

mLeft.stop(stop\_action = 'brake')

time.sleep(1)

fh.close()  
  
4. Графики

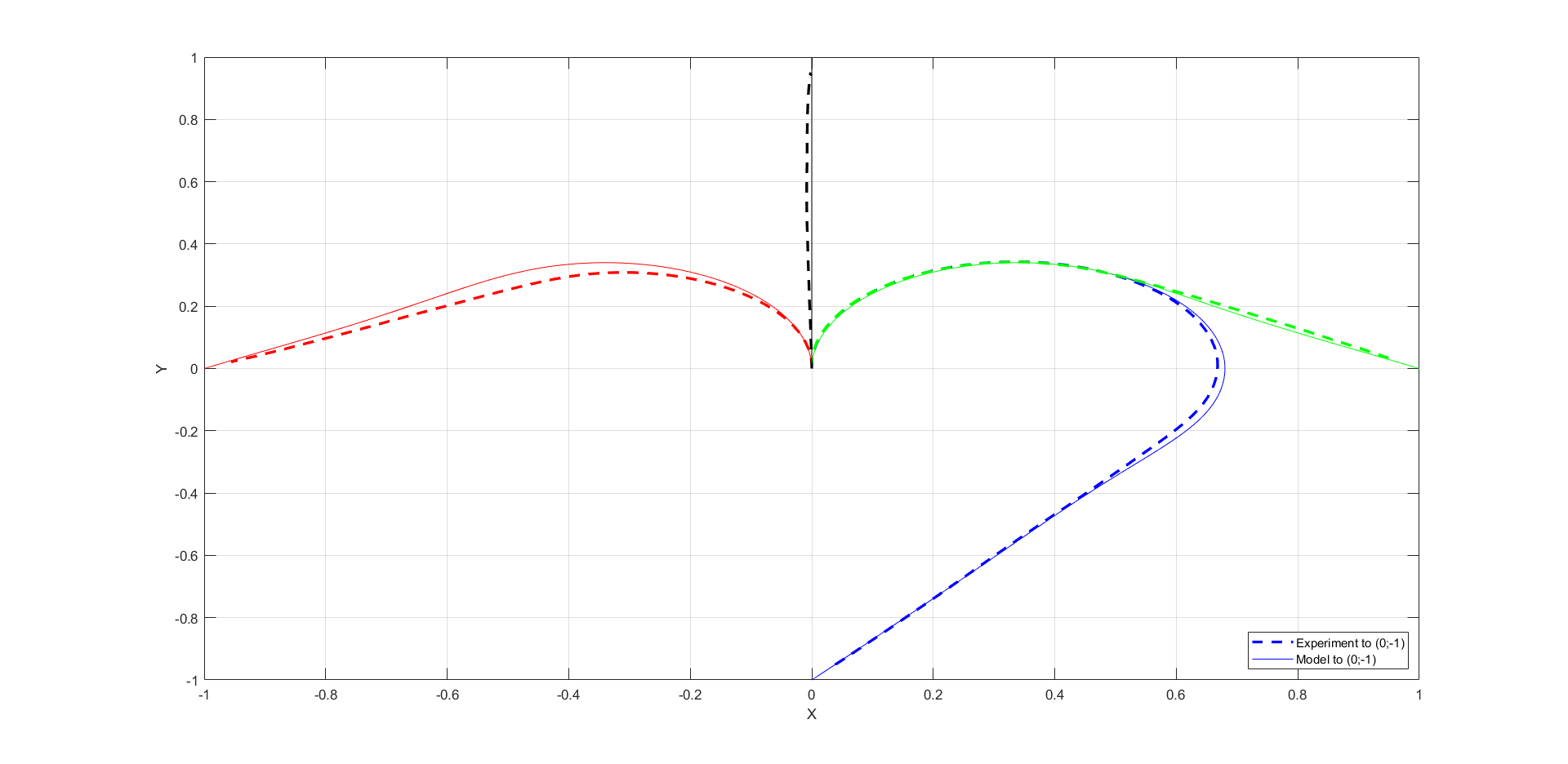


Рисунок 1. Графики траектории движения робота до заданных точек

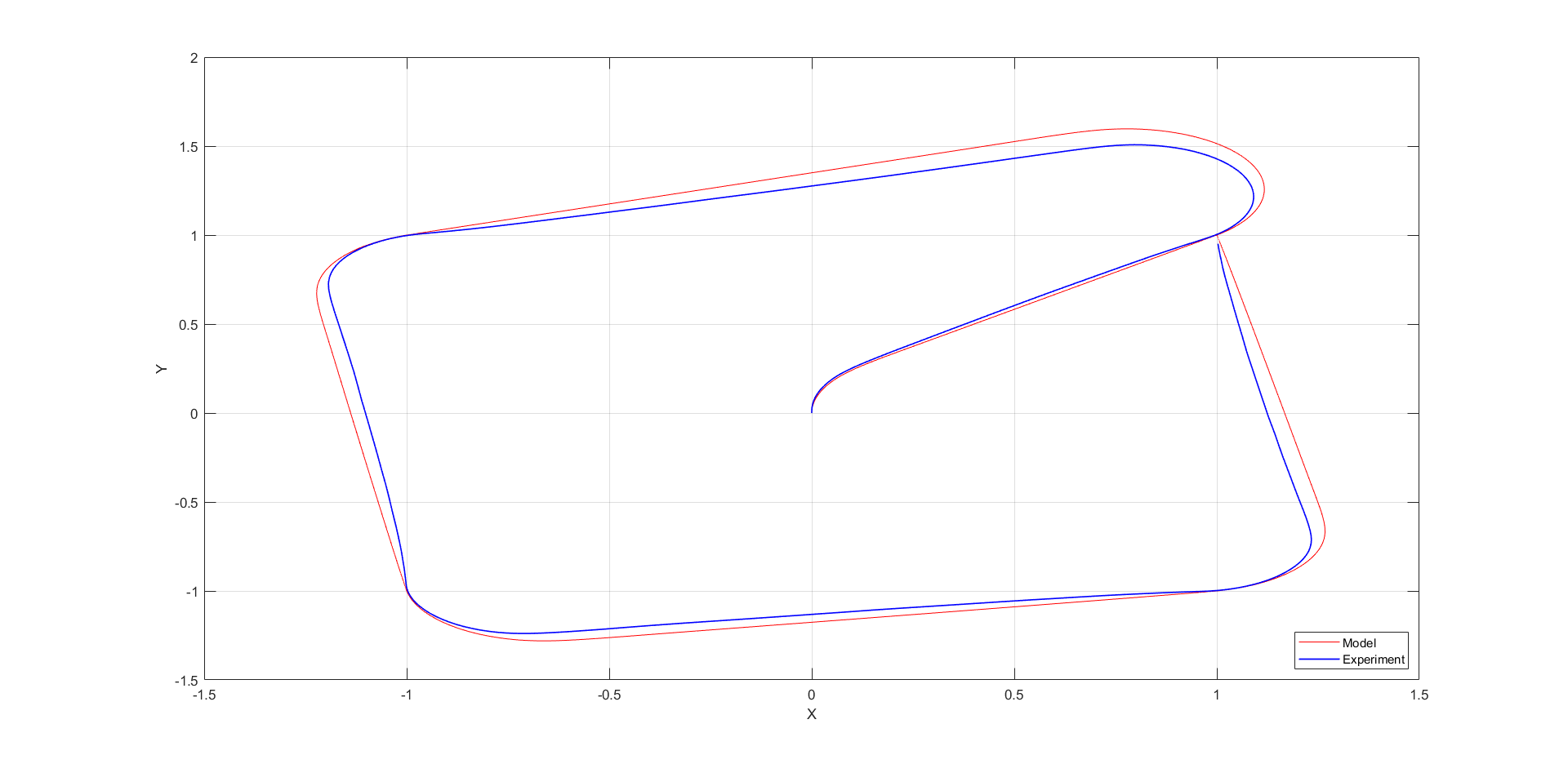
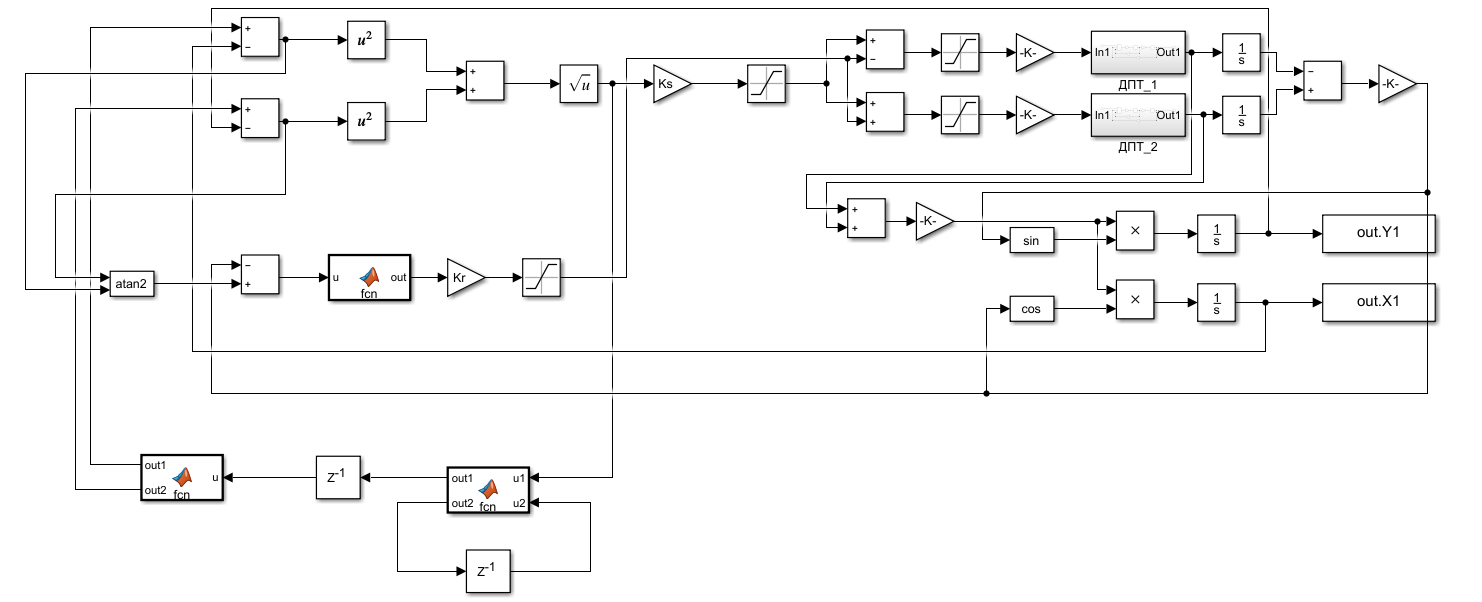


Рисунок 2. Траектория движения робота по квадрату

5. Модель в Simulink



6. Выводы