

**ПОСТРОЕНИЕ 2D-МОДЕЛИ РОБОТА, ИГРАЮЩЕГО В ПИНГ-ПОНГ**

Алиусеф Алмасалмах Фади

aliyusef\_almasalmakhf@student.bmstu.ru

SPIN-код: 4501-9839

МГТУ им. Н.Э. Баумана, Москва, Российская Федерация

**Аннотация**

Исследована кинематическая модель робота, играющего в пинг-понг в плоскости. Определены условия и ограничения столкновения мяча со столом и условия окружающей среды. Представлены расчетные уравнения движения мяча, а также уравнения его столкновения с ракеткой и со столом. Приведены расчеты скорости ракетки в момент удара, необходимой для того чтобы мяч летел по выбранной траектории. Решены обратная и прямая кинематические задачи для выбранного робота. С использованием известных скорости и позиции мяча в момент удара рассчитаны зависимости движения звена робота от времени. На языке C# с помощью библиотеки OpenGL реализована 2D-модель, в которой два манипулятора играют в пинг-понг друг с другом.

**Ключевые слова**

Робот, играющий в пинг-понг, манипулятор, сопротивление воздуха, сила Магнуса, коэффициент восстановления, прямая кинематическая задача, обратная кинематическая задача, матрица Якоби, планирование траектории, OpenGL

Поступила в редакцию 12.12.2019

© МГТУ им. Н.Э. Баумана, 2020

**Введение.** В настоящее время роботы больше не являются инструментами для выполнения стереотипных и простых задач. На данной ступени развития они перешли эти границы и теперь способны решать множество сложных задач с высокой точностью и скоростью, выполнение которых затруднительно, в том числе и для людей.

Одним из примеров таких задач, для выполнения которых следует эффективно и быстро принимать решения в заданной ситуации, является игра в пинг-понг. Эта игра — большое испытание для роботов на текущей стадии их развития. Данные испытания проводятся до сих пор, что обусловлено необходимостью быстрой реакции робота и бесконечным поиском способов удара по мячу. Именно поэтому изучение подобных задач способствует развитию современного уровня робототехники и алгоритмов в целом [1, 2].

**Постановка задачи.** В этой статье будем строить 2D-модель робота, играющего в пинг-понг (в плоскости), противником которого будет являться такой же робот (рис. 1).

**Частные задачи:**

- составить уравнения, которые описывают движения мячика;
- определить точку, в которой манипулятор должен ударить по мячику;
- выбрать желаемую траекторию мячика после удара;

- вычислить скорость и ориентацию ракетки в момент удара, чтобы мячик летел по определенной траектории;
- построить нужную траекторию движения манипулятора.

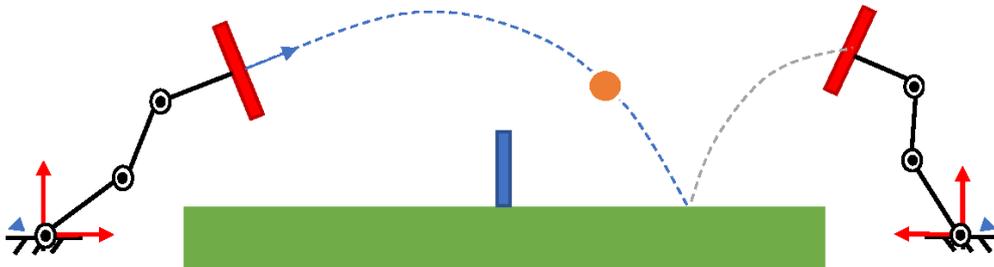


Рис. 1. Задача игры в пинг-понг

**Условия и ограничения.** Для упрощения задачи моделирования при нашей модели мы не будем учитывать следующие факторы:

- сопротивление воздуха;
- вращения мячика (и, следовательно, эффект Магнуса [3]).

**Модель робота.** Будем рассматривать движение 3R-робота в плоскости (рис. 2), поскольку требуется двигать ракетку в плоскости и менять ее ориентацию.

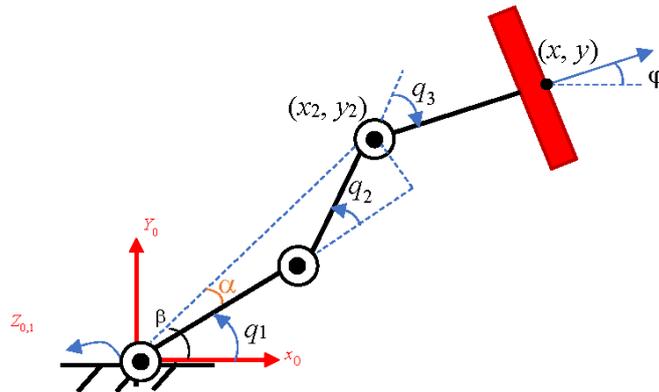


Рис. 2. Кинематическая модель робота

**Прямая кинематическая модель [4].** Эта модель необходима, чтобы знать позицию и ориентацию ракетки при текущих углах поворота звеньев. Для построения модели спроецируем звенья робота на заданные фиксированные координаты:

$$\begin{cases} l_1 c_1 + l_2 c_{12} + l_3 c_{123} = x; \\ l_1 s_1 + l_2 s_{12} + l_3 s_{123} = y; \\ q_1 + q_2 + q_3 = \varphi, \end{cases}$$

где  $l_1, l_2, l_3$  — длины звеньев робота;  $x, y$  — координаты центральной точки ракетки;  $\varphi$  — угол, показанный на рис. 2;  $c_{ij} = \cos(q_i + q_j)$ ;  $s_{ij} = \sin(q_i + q_j)$ ;  $q_1, q_2, q_3$  — углы звеньев робота (см. рис. 2).

**Обратная кинематическая модель [4].** Эта модель необходима для того, чтобы рассчитать углы суставов, при которых ракетка находилась бы в желаемой позиции и ориентации.

Чтобы вычислить углы суставов робота, запишем следующие уравнения:

$$\begin{aligned}x_2 &= x - l_3 c_\varphi; \\y_2 &= y - l_3 s_\varphi,\end{aligned}$$

где  $x_2, y_2$  — координаты центральной точки второго сустава;

$$x_2^2 + y_2^2 = l_1^2 + l_2^2 - 2l_1 l_2 c_2 \Rightarrow q_2 = \arccos\left(\frac{l_1^2 + l_2^2 - x_2^2 - y_2^2}{2l_1 l_2}\right) \in [0^\circ, 180^\circ]. \quad (1)$$

Если локоть робота находится в верхнем положении, то можно записать

$$q_2 = -\arccos\left(\frac{l_1^2 + l_2^2 - x_2^2 - y_2^2}{2l_1 l_2}\right);$$

$$\beta = \arctg 2(y_2, x_2);$$

$$\alpha = \arctg 2(l_2 s_2, l_1 + l_2 c_2);$$

$$q_1 = \beta - \alpha$$

$$\Rightarrow q_1 = \arctg 2(y_2, x_2) - \arctg 2(l_2 s_2, l_1 + l_2 c_2); \quad (2)$$

$$q_3 = \varphi - q_1 - q_2. \quad (3)$$

**Матрица Якоби [5].** Используя эту матрицу и зная линейную и угловую скорость ракетки, можно рассчитать необходимые угловые скорости суставов робота:

$$J = \begin{pmatrix} -l_1 s_1 - l_2 s_{12} - l_3 s_{123} & -l_2 s_{12} - l_3 s_{123} & -l_3 s_{123} \\ l_1 c_1 + l_2 c_{12} + l_3 c_{123} & l_2 c_{12} + l_3 c_{123} & l_3 c_{123} \\ 1 & 1 & 1 \end{pmatrix}; \quad (4)$$

$$\det J = l_1 l_2 s_2;$$

$$\dot{X} = J\dot{q} \Rightarrow \dot{q} = J^{-1}\dot{X} \quad (\text{if } \det(J) \neq 0 \Rightarrow s_2 \neq 0);$$

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{pmatrix} = J^{-1} \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \dot{\varphi} \end{pmatrix}, \quad (5)$$

где  $\dot{q}_1, \dot{q}_2, \dot{q}_3$  — угловые скорости звеньев робота;  $v_{Rx}, v_{Ry}$  — скорости ракетки по осям  $x_0, y_0$  соответственно;  $\dot{\varphi}$  — это угловая скорость ракетки вокруг оси  $z_0$ .

**Планирование траектории ракеты.** В рамках поставленной задачи в момент времени  $t_0$  робот начинает двигаться из текущей точки  $P_0$  с начальной нулевой скоростью, робот должен прибывать в точку  $P_h$  в момент времени  $t_f$ , скорость ракеты в этой точке должна составлять  $\mathbf{v}_R$ .

Таким образом, для каждого звена можно записать следующие условия движения:

$$\begin{aligned} q_{i,0} &= q_i(t_0); \quad q_{i,f} = q_i(t_f); \\ \dot{q}_{i,0} &= 0; \quad \dot{q}_{i,f} = \dot{q}_i(t_f). \end{aligned}$$

Эти условия можно реализовать, если  $q_i(t)$  является полиномом 3-й степени [5]:

$$q_i(t) = a_i t^3 + b_i t^2 + c_i t + d_i.$$

Подставим ограничительные условия:

$$q_{i,0} = q_i(0) = d_i; \quad [\text{I}]$$

$$q_{i,f} = q_i(t_f) = a_i t_f^3 + b_i t_f^2 + c_i t_f + d_i; \quad [\text{II}]$$

$$\dot{q}_{i,0} = c_i = 0; \quad [\text{III}]$$

$$\dot{q}_{i,f} = \dot{q}_i(t_f) = 3a_i t_f^2 + 2b_i t_f + c_i; \quad [\text{IV}]$$

$$[\text{III}] \rightarrow [\text{III}]: \dot{q}_{i,f} = 3a_i t_f^2 + 2b_i t_f; \quad [\text{V}]$$

$$[\text{I}], [\text{III}] \rightarrow [\text{II}]: q_{i,f} - q_{i,0} = a_i t_f^3 + b_i t_f^2; \quad [\text{VI}]$$

$$[\text{V}] \times t_f - [\text{VI}] \times 3: \dot{q}_{i,f} t_f - 3q_{i,f} + 3q_{i,0} = -3a_i t_f^3 + 3a_i t_f^3 - 3b_i t_f^2 + 2b_i t_f^2;$$

$$\dot{q}_{i,f} t_f - 3q_{i,f} + 3q_{i,0} = -b_i t_f^2;$$

$$\Rightarrow \left. \begin{aligned} b_i &= \frac{3q_{i,f} - 3q_{i,0} - \dot{q}_{i,f} t_f}{t_f^2}; \\ a_i &= \frac{q_{i,f} - q_{i,0} - b_i t_f^2}{t_f^3}; \\ c_i &= 0; \\ d_i &= q_{i,0}. \end{aligned} \right\} \quad (6)$$

**Удар мячика о ракетку. Уравнения столкновения с ракеткой** [6]. Запишем скорости мячика относительно ракеты:

$$\begin{aligned} \mathbf{v}_{B,rel}^- &= \mathbf{v}_B^- - \mathbf{v}_R; \\ \mathbf{v}_{B,rel}^+ &= \mathbf{v}_B^+ - \mathbf{v}_R, \end{aligned} \quad (7)$$

где  $\mathbf{v}_{B,rel}^-$ ,  $\mathbf{v}_{B,rel}^+$  — скорость мячика до и после удара относительно ракеты;  $\mathbf{v}_B^-$ ,  $\mathbf{v}_B^+$  — скорость мячика до и после удара;  $\mathbf{v}_R$  — скорость ракетки.

Для упрощения опустим индексы и введем обозначения  $\mathbf{v}_{B,rel}^- = \mathbf{v}^-$ ;  $\mathbf{v}_{B,rel}^+ = \mathbf{v}^+$ .

Будем считать, что мячик теряет часть скорости по направлению  $\mathbf{n}$  после отражения. Это можно учесть с помощью коэффициента восстановления  $k$  [7] и относительных скоростей (см. рис. 3):

$$v_n^+ = -kv_n^-, \quad (8)$$

где  $v_n^+$  — проекция  $\mathbf{v}^+$  на  $\mathbf{n}$ ;  $v_n^-$  — проекция  $\mathbf{v}^-$  на  $\mathbf{n}$ .

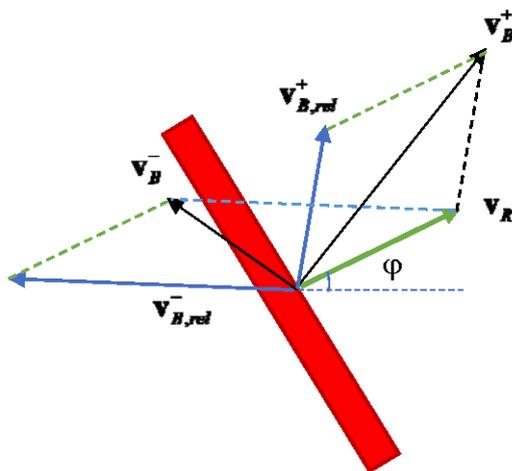


Рис. 3. Скорости мячика относительно ракетки

Предполагаем, что скорость по оси  $\mathbf{t}$  не меняется (рис. 4):

$$v_t^+ = v_t^-, \quad (9)$$

где  $v_t^+$  — проекция  $\mathbf{v}^+$  на  $\mathbf{t}$ ;  $v_t^-$  — проекция  $\mathbf{v}^-$  на  $\mathbf{t}$ .

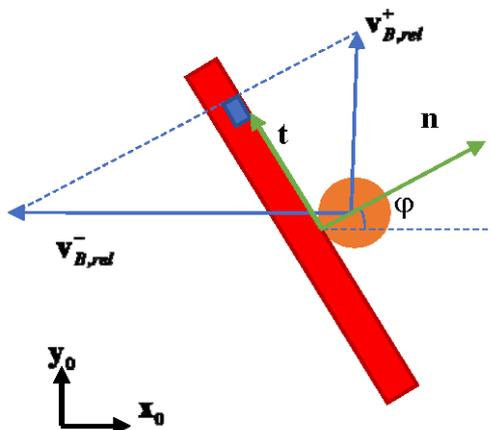


Рис. 4. Скорости мячика относительно ракетки

Из (9) можно вывести следующее:

$$\mathbf{v}^+ = \mathbf{v}^- + \lambda \mathbf{n}.$$

Проецируя это уравнение на вектор  $\mathbf{n}$ , получим:

$$\begin{aligned} \mathbf{v}^+ \cdot \mathbf{n} &= (\mathbf{v}^- + \lambda \mathbf{n}) \cdot \mathbf{n} \\ \Rightarrow -k v_n^- &= v_n^- + \lambda \\ \Rightarrow \lambda &= -(k+1)v_n^- \\ \Rightarrow \boxed{\mathbf{v}^+ = \mathbf{v}^- - (1+k)v_n^- \mathbf{n}}. \end{aligned}$$

Предположим, что скорость  $\mathbf{v}_R$  параллельна вектору нормали к поверхности ракетки  $\mathbf{n}$ .

**Нахождение ориентации ракетки.** Если  $\mathbf{v}_B^+$  и  $\mathbf{v}_B^-$  известны, можно определить требуемую ориентацию. Для этого используем уравнение (9) и уравнение (7) в форме

$$v_t^+ = v_t^- \text{ или } \mathbf{v}_{Bt}^+ = \mathbf{v}_{Bt}^-,$$

где  $\mathbf{t}$  — вектор, касательный к ракетке;  $\mathbf{n}$  — вектор, нормальный к ракетке;  $\mathbf{v}_{Bt}^+$ ,  $\mathbf{v}_{Bt}^-$  — проекция скоростей до и после удара на ось  $\mathbf{t}$ .

Поэтому можно определить ориентацию ракетки так, чтобы уравнение (9) было справедливым.

Предположим следующее:

$$\bar{\mathbf{n}} = \frac{\mathbf{v}_B^+ - \mathbf{v}_B^-}{\|\mathbf{v}_B^+ - \mathbf{v}_B^-\|},$$

поскольку  $\mathbf{t} \perp \mathbf{n}$ , найдем  $\mathbf{t}$  следующим образом:

$$\mathbf{t} = \underbrace{\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}}_A \cdot \mathbf{n}.$$

Докажем, что  $\mathbf{t}$  и  $\mathbf{n}$  удовлетворяют уравнению (9):

$$\begin{aligned} \mathbf{t} = A \cdot \mathbf{n} &= \frac{1}{\|\mathbf{v}_B^+ - \mathbf{v}_B^-\|} (A \cdot \mathbf{v}_B^+ - A \cdot \mathbf{v}_B^-); \\ \mathbf{v}_B^+ \cdot \mathbf{t} &= \frac{1}{\|\mathbf{v}_B^+ - \mathbf{v}_B^-\|} \left( \underbrace{\mathbf{v}_B^+ \cdot (A \cdot \mathbf{v}_B^+)}_{=0} - \mathbf{v}_B^+ \cdot A \cdot \mathbf{v}_B^- \right). \end{aligned}$$

Так как  $(A \cdot \mathbf{v}_B^+)$  перпендикулярен  $\mathbf{v}_B^+$ ,

$$\begin{aligned} \mathbf{v}_B^- \cdot \mathbf{t} &= \frac{1}{\|\mathbf{v}_B^+ - \mathbf{v}_B^-\|} \left( \mathbf{v}_B^- \cdot A \cdot \mathbf{v}_B^+ - \underbrace{\mathbf{v}_B^- \cdot A \cdot \mathbf{v}_B^-}_{=0} \right) \\ &\Rightarrow \mathbf{v}_B^+ \cdot \mathbf{t} = \mathbf{v}_B^- \cdot \mathbf{t}. \end{aligned}$$

Это то, что мы стремились доказать. Таким образом,

$$\boxed{\varphi = \arctg 2\left((\mathbf{v}_{By}^+ - \mathbf{v}_{By}^-), (\mathbf{v}_{Bx}^+ - \mathbf{v}_{Bx}^-)\right)}. \quad (1)$$

**Вычисление требуемой скорости ракетки.** Чтобы у мячика была желаемая скорость  $\mathbf{v}_B^+$ , можно выбрать  $\mathbf{v}_R$ , параллельную  $\mathbf{n}$ . На основе уравнения (8) запишем

$$\begin{aligned} \mathbf{v}^+ \cdot \mathbf{n} &= -k\mathbf{v}^- \cdot \mathbf{n}; \\ (1) \Rightarrow \mathbf{v}_B^+ \cdot \mathbf{n} - \mathbf{v}_R \cdot \mathbf{n} &= -k(\mathbf{v}_B^- \cdot \mathbf{n} - \mathbf{v}_R \cdot \mathbf{n}); \\ (1+k)\mathbf{v}_R \cdot \mathbf{n} &= \mathbf{v}_B^+ \cdot \mathbf{n} + k\mathbf{v}_B^- \cdot \mathbf{n}; \\ \mathbf{v}_R \cdot \mathbf{n} &= \frac{\mathbf{v}_B^+ \cdot \mathbf{n} + k\mathbf{v}_B^- \cdot \mathbf{n}}{1+k}, \end{aligned}$$

подставим  $\mathbf{n} = \begin{pmatrix} c_\varphi \\ s_\varphi \end{pmatrix}$ . Будем иметь

$$\boxed{\|\mathbf{v}_R\| = \mathbf{v}_R \cdot \mathbf{n} = \frac{v_{Bx}^+ c_\varphi + v_{By}^+ s_\varphi + kv_{Bx}^- c_\varphi + kv_{By}^- s_\varphi}{1+k}; \quad \mathbf{v}_R = \|\mathbf{v}_R\| \begin{pmatrix} c_\varphi \\ s_\varphi \end{pmatrix}. \quad (2)}$$

**Простая модель траектории мяча.** Путь мяча над столом состоит из двух этапов: этап до удара со столом и этап после удара.

**Упрощенная модель движения мяча.** Мяч начинает лететь с начальной скоростью  $\mathbf{v}_0$ . Будем считать, что единственная сила, действующая на него, это сила тяжести:

$$\begin{aligned} \Sigma \mathbf{F} &= m\mathbf{a}; \\ m\mathbf{g} &= m\mathbf{a}. \end{aligned}$$

Проецируя на ось  $y_0$ , получим:

$$\begin{aligned} \ddot{y} &= -g; \\ \dot{y}(t) &= -gt + -v_{0y}; \\ y(t) &= -\frac{1}{2}gt^2 + v_{0y}t + y_0. \end{aligned}$$

В проекции на ось  $x$  запишем:

$$\begin{aligned}\ddot{x} &= 0; \\ \dot{x}(t) &= \dot{x}_0 = v_{0x}; \\ x(t) &= v_{0x}t + x_0;\end{aligned}$$

$$x(t) = v_{0x}t + x_0 \Rightarrow t = \frac{x(t) - x_0}{v_{0x}}.$$

Подставим  $t$ :

$$\begin{aligned}y(x) &= -\frac{1}{2}g\left(\frac{x(t) - x_0}{v_{0x}}\right)^2 + v_{0y}\left(\frac{x(t) - x_0}{v_{0x}}\right) + y_0; \\ y(x) &= \underbrace{\left(\frac{-g}{2v_{0x}^2}\right)}_a x^2 + \underbrace{\left(\frac{-gx_0}{v_{0x}^2} + \frac{v_{0y}}{v_{0x}}\right)}_b x + \underbrace{\left(-\frac{gx_0^2}{2v_{0x}^2} - \frac{v_{0y}x_0}{v_{0x}} + y_0\right)}_c.\end{aligned}\quad (3)$$

Определим желаемую траекторию мяча. Для этого можно выбрать три точки. В нашей модели будем выбирать траекторию простым или произвольным образом. Первая точка — это точка удара мяча о ракетку. Вторая точка должна быть выше сетки и иметь случайную высоту. Третья точка должна быть на стороне противника.

После того как три точки выбраны, найдем константы уравнения  $y(x)$ , решив систему уравнений:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ x_3^2 & x_3 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix}.\quad (4)$$

Найдем требуемую скорость мячика после удара из уравнения (13):

$$\begin{aligned}a = \left(\frac{-g}{2v_{0x}^2}\right) &\Rightarrow v_{0x}^+ = \sqrt{\frac{-g}{2a}}; \\ b = \left(\frac{-gx_0}{v_{0x}^2} + \frac{v_{0y}}{v_{0x}}\right) &\Rightarrow v_{0y}^+ = bv_{0x}^+ + \frac{gx_0}{v_{0x}^+}.\end{aligned}\quad (5)$$

**Модель простого столкновения со столом.** Предполагая, что условия удара почти идеальны и что вращением мяча пренебрегают, можно считать удар мяча об стол подчиняющимся законам отражения света, но необходимо учесть

уменьшение составляющих скорости, так как оно имеет значительное влияние на форму траектории:

$$\begin{aligned} v_{ex} &\approx v_{ix}; \\ v_{ey} &= -k_y v_{iy}, \end{aligned}$$

где  $v_e$  — скорость после удара об стол;  $v_i$  — скорость до удара об стол;  $k_y$  — коэффициент реституции, зависящий от особенностей мяча и стола, обычно принимают  $k_y \approx 0,88$  [8, 9].

Траектория после удара об стол также является параболой, движение начинается со скоростью  $\mathbf{v}_e$ . Уравнения параболы имеют вид

$$\begin{aligned} y(t) &= -\frac{1}{2}g(t-t_b)^2 + v_{ey}t + y_b; \\ x(t) &= v_{ex}(t-t_b) + x_b, \end{aligned}$$

где  $t_b$  — момент столкновения со столом;  $x_b, y_b$  — координаты точки столкновения.

**2D-моделирование.** Выполним симуляцию, чтобы убедиться, что полученные уравнения движения и столкновения верные. Начнем с 2D-моделирования, это упростит дальнейшие построения и позволит понять метод работы в 3D-пространстве.

**Шаги симуляции игры в 2D-пространстве.** Для симуляции необходимо выполнить следующие шаги:

- 1) определить точку удара мяча о ракетку  $P_h(x, y)$ ;
- 2) вычислить  $\mathbf{v}^-$  в точке удара  $P_h(x, y)$ ;
- 3) из уравнения (14) определить желаемую траекторию мячика;
- 4) из выражения (15) вычислить  $\mathbf{v}^+$ ;
- 5) по вычисленным  $\mathbf{v}^-, \mathbf{v}^+$  с использованием (11) и (12) найти  $\phi$  и  $\mathbf{v}_R$ ;
- 6) по определенным  $P_h(x, y)$  и  $\phi$ , используя (1) и (2) и (3), вычислить  $q_{1,f}, q_{2,f}, q_{3,f}$ ;
- 7) по известным  $q_{1,f}, q_{2,f}, q_{3,f}$ , используя выражения (4), вычислить  $J$  (матрицу Якоби);
- 8) зная  $J$  и  $\mathbf{v}_R$  и используя (5), вычислить  $\dot{q}_{1,f}, \dot{q}_{2,f}, \dot{q}_{3,f}$ ;
- 9) из выражения (6) вычислить константы уравнений звеньев робота.

**Реализация модели.** Мы реализовали эту модель на языке C# с помощью библиотеки OpenGL [10]. Желаемая траектория мяча выбирается произвольным образом, и это происходит через произвольный выбор трех точек, первая из которых является точкой удара с ракеткой, вторая расположена выше сетки, а третья находится на половине стола в стороне оппонента. Результат симуляции показан на рис. 5.

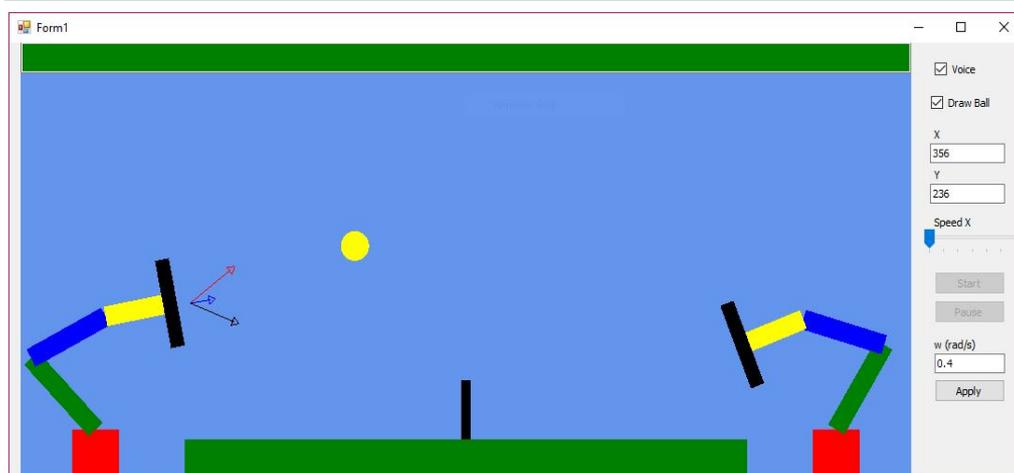


Рис. 5. Реализация модели на языке С# и в библиотеке OpenGL

**Заключение.** В ходе работы рассмотрена кинематическая модель робота, играющего в пинг-понг в плоскости. Представлены расчеты уравнений движения мяча, уравнения столкновения с ракеткой и со столом с учетом только силы тяжести и без учета вращения. Рассчитана скорость ракетки в момент удара, необходимая для того, чтобы мяч летел по заданной траектории. Решены обратная и прямая кинематические задачи для выбранного робота. С использованием известных скорости и позиции мяча в момент удара построены функции движения звеньев робота от времени. На языке С# с помощью библиотеки OpenGL реализована 2D-модель, в которой два манипулятора играют в пинг-понг друг против друга. Эта 2D-модель является первым шагом на пути построения 3D-модель робота, играющего в пинг-понг.

### Литература

- [1] Muelling K. Modeling and learning of complex motor tasks: a case study with robot table tennis. Ph.D. Thesis. Darmstadt, Technische Universität, 2013.
- [2] Li H., Wu H., Lou L., et al. Ping-pong robotics with high-speed vision system. Proc. ICARCV, 2012, pp. 106–111. DOI: <https://doi.org/10.1109/ICARCV.2012.6485142>
- [3] Huang Y., Xu D., Tan M., et al. Trajectory prediction of spinning ball for ping-pong player robot. Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2011, pp. 3434–3439. DOI: <https://doi.org/10.1109/IROS.2011.6095044>
- [4] Murray R.M., Li Z. A mathematical introduction to robotic manipulation. Routledge, 1994.
- [5] Dombre E., Wisama Kh., eds. Robot manipulators: modeling, performance analysis and control. John Wiley & Sons, 2013.
- [6] Cross R. Impact of a ball with a bat or racket. *Am. J. Phys.*, 1999, vol. 67, no. 8, pp. 692–702. DOI: <https://doi.org/10.1119/1.19354>
- [7] Cross R. The coefficient of restitution for collisions of happy balls, unhappy balls, and tennis balls. *Am. J. Phys.*, 2000, vol. 68, no. 11, pp. 1025–1031. DOI: <https://doi.org/10.1119/1.1285945>

- [8] Haron A., Ismail K.A. Coefficient of restitution of sports balls: a normal drop test. *IOP Conf. Ser.: Mater. Sci. Eng.*, 2012, vol. 36, no. 1, art. 012038. DOI: <https://doi.org/10.1088/1757-899X/36/1/012038>
- [9] The Ball (version for 40 mm balls). Technical Leaflet T3. ITTF, 2003.
- [10] OpenGL Overview. *opengl.org: веб-сайт*. URL: <https://www.opengl.org/about/> (дата обращения: 12.12.2019).

**Алиюсеф Алмасалмах Фади** — студент магистратуры кафедры «Робототехнические системы и мехатроника», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

**Научный руководитель** — Назарова Анаид Вартановна, кандидат технических наук, доцент кафедры «Робототехнические системы и мехатроника», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

**Ссылку на эту статью просим оформлять следующим образом:**

Алиюсеф Алмасалмах Фади. Построение 2D-модели робота, играющего в пинг-понг. *Политехнический молодежный журнал*, 2020, № 03(44). <http://dx.doi.org/10.18698/2541-8009-2020-03-589>

---

**CONSTRUCTION OF A 2D MODEL OF A ROBOT PLAYING TABLE TENNIS**

Alyousef Almasalmah Fadi

aliyusef\_almasalmakhf@student.bmstu.ru

SPIN-code: 4501-9839

Bauman Moscow State Technical University, Moscow, Russian Federation

**Abstract**

The paper is devoted to kinematic model of a robot playing table tennis in plane. The conditions and limitations of the collision of the ball with the table and environmental conditions are determined. The equations of ball motion are presented, as well as the equations of its collision with a racket and with a table. Calculations are given of the racket speed at the moment of impact, necessary for the ball to fly along the chosen path. The inverse and direct kinematic problems are solved for the selected robot. Using the known speed and position of the ball at the moment of impact, the dependences of the robot links movement on time are calculated. A 2D model in which two manipulators play table tennis with each other is implemented in the C# language using the OpenGL library.

**Keywords**

Robot playing table tennis, manipulator, air resistance, Magnus force, recovery coefficient, direct kinematic problem, inverse kinematic problem, Jacobi matrix, trajectory planning, OpenGL

Received 12.12.2019

© Bauman Moscow State Technical University, 2020

**References**

- [1] Muelling K. Modeling and learning of complex motor tasks: a case study with robot table tennis. Ph.D. Thesis. Darmstadt, Technische Universität, 2013.
- [2] Li H., Wu H., Lou L., et al. Ping-pong robotics with high-speed vision system. *Proc. ICARCV*, 2012, pp. 106–111. DOI: <https://doi.org/10.1109/ICARCV.2012.6485142>
- [3] Huang Y., Xu D., Tan M., et al. Trajectory prediction of spinning ball for ping-pong player robot. *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2011, pp. 3434–3439. DOI: <https://doi.org/10.1109/IROS.2011.6095044>
- [4] Murray R.M., Li Z. A mathematical introduction to robotic manipulation. Routledge, 1994.
- [5] Dombre E., Wisama Kh., eds. Robot manipulators: modeling, performance analysis and control. John Wiley & Sons, 2013.
- [6] Cross R. Impact of a ball with a bat or racket. *Am. J. Phys.*, 1999, vol. 67, no. 8, pp. 692–702. DOI: <https://doi.org/10.1119/1.19354>
- [7] Cross R. The coefficient of restitution for collisions of happy balls, unhappy balls, and tennis balls. *Am. J. Phys.*, 2000, vol. 68, no. 11, pp. 1025–1031. DOI: <https://doi.org/10.1119/1.1285945>
- [8] Haron A., Ismail K.A. Coefficient of restitution of sports balls: a normal drop test. *IOP Conf. Ser.: Mater. Sci. Eng.*, 2012, vol. 36, no. 1, art. 012038. DOI: <https://doi.org/10.1088/1757-899X/36/1/012038>
- [9] The Ball (version for 40 mm balls). Technical Leaflet T3. ITTF, 2003.
- [10] OpenGL Overview. *opengl.org: website*. URL: <https://www.opengl.org/about> (accessed: 12.12.2019).

**Alyousef Almasalmah Fadi** — Master's Degree Student, Department of Robotics and Mechatronics, Bauman Moscow State Technical University, Moscow, Russian Federation.

**Scientific advisor** — Nazarova A.V., Cand. Sc. (Eng.), Assoc. Professor, Department of Robotics and Mechatronics, Bauman Moscow State Technical University, Moscow, Russian Federation.

**Please cite this article in English as:**

Alyousef Almasalmah Fadi. Construction of a 2D model of a robot playing table tennis. *Politekhnicheskiy molodezhnyy zhurnal* [Politechnical student journal], 2020, no. 03(44). <http://dx.doi.org/10.18698/2541-8009-2020-03-589.html> (in Russ.).