

# СИНТЕЗ АЛГОРИТМА КОРРЕКЦИИ ОШИБОК ДЛЯ РЕШЕНИЯ ПРЯМОЙ ЗАДАЧИ КИНЕМАТИКИ ПАРАЛЛЕЛЬНЫХ РОБОТОВ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ

---

*rl\_bgtu@intbel.ru*

**Рыбак Л.А., д-р техн. наук, проф.,  
Гапоненко Е.В.,  
Мамаев Ю.А.,  
Мальшев Д.И.**

*Белгородский государственный технологический  
университет им. В.Г. Шухова*

**Аннотация.** В данной статье рассматриваются преимущества использования параллельных роботов. Из-за больших затрат времени на решение прямой задачи кинематики для параллельного робота, кинематический и динамический алгоритмы управления не могут быть реализованы в реальном времени. Чтобы сократить время на решение прямой задачи кинематики, в статье использован алгоритм решения данной задачи с помощью нейронных сетей. Обучение нейронной сети на основе алгоритма коррекции ошибок позволит осуществлять управление параллельным роботом с помощью обычных алгоритмов управления. Результаты моделирования показывают, что нейронные сети позволяют сократить время, необходимое для получения достаточно точного приближения к решению прямой задачи кинематики для параллельных роботов.

**Ключевые слова:** параллельный робот, нейронная сеть, прямая задача кинематики, персептрон, алгоритм компенсации ошибки, обучение нейронной сети.

В последние несколько лет наблюдается существенное расширение области применения роботов в промышленности, что обусловлено, в основном, их гибкостью. Наиболее распространенный в настоящее время манипуляторы имеют явно антропоморфный характер и обычно сильно напоминают человеческую руку. Они сконструированы из последовательности твердых тел, каждое из которых соединено с последующим и предыдущим неподвижными шарнирами, позволяющие твердому телу, например, вращаться вокруг оси или совершать поступательное перемещение. Такая архитектура называется последовательным роботом. Однако механическая архитектура

наиболее распространенных типов роботов не является приспособленной к определенным типам задач. Общеизвестно, что абсолютная точность последовательного робота не высока. В связи с этим изучаются и все более широко применяются в промышленности параллельные роботы, у которых выходное звено связано с основанием множеством кинематических цепей, каждая из которых несет на себе лишь часть общей полезной нагрузки.

Главной сложностью при проектировании параллельных роботов является решение прямой задачи кинематики. Ее решением занимались многие исследователи, однако, из-за отсутствия общей методик ее решения исследования в этом направлении далеки от завершения. В настоящее время существует 4 типа методов ее решения: аналитические методы [3, 4, 6], исследование использование дополнительных датчиков [8], численные методы [1, 5] и применение методов искусственного интеллекта, в частности, нейронных сетей.

В работе [2] использовался многослойный перцептрон с обратным распространением ошибки. Они предлагали использовать перцептрон для решения прямой задачи кинематики не имея точного аналитического ее решения. В работе [7] с помощью нейронных сетей практически решена прямая задача кинематики для конкретной конфигурации робота. Однако в этой работе нет математического обоснования работоспособности предложенного метода; метода выбора функций активации нейронов и анализа применимости предложенного подхода к обобщенному параллельному роботу, т.е. к платформе Гауфа–Стюарта.

В данной работе предлагается метод решения прямой задачи кинематики, основанный на применении нейронной сети, решения прямой задачи кинематики и алгоритма коррекции ошибок в соответствии со схемой, приведенной на рис. 1.

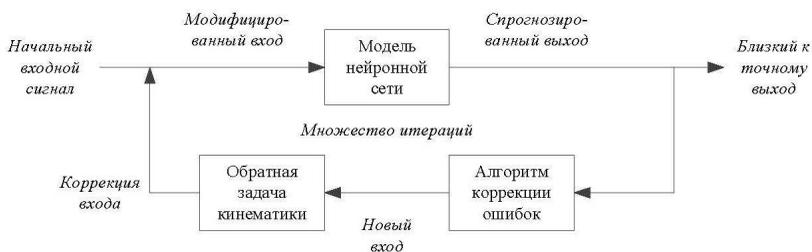


Рисунок 1 – Схема метода решения прямой задачи кинематики

Прямая задача кинематики широко известно и далее рассматриваться не будет. Рассмотрим более подробно алгоритмы работы нейронной сети и коррекции ошибок.

Впервуюочередьнеобходимоопределитьмодельнейроннойсети,предназначенной для аппроксимации функции  $F$ , отображающей пространство шарниров на пространство задачи. Воспользуемся многослойным персептроном с обучениемметодом обратного распространения ошибки (рис. 2).

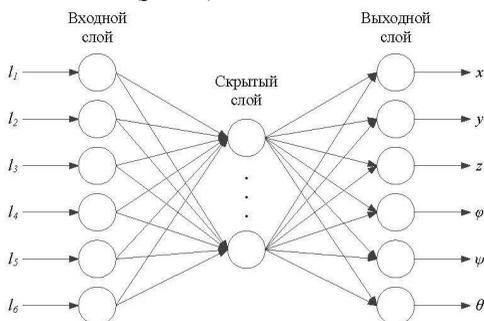


Рисунок 2 – Структура нейронной сети

Мы воспользовались двумя структурами многослойного персептрона: стандартной и с дополнительными прямыми связями. В стандартном персептроне все нейроны одного слоя соединены со всеми нейронами последующего слоя. В персептроне с дополнительными прямыми связями все нейроны одного слоя соединены со всеми нейронами всех последующих слоев, что увеличивает количество соединений и, следовательно, увеличивает возможности моделирования прямых и косвенных связей между входами и выходами.

После создания модели нейронной сети был разработан алгоритм компенсации ошибки. Введем обозначения, описывающие кинематические задачи параллельных роботов:

а) параметры, описывающие пространство шарниров, представим вектором  $U$ , а параметры, описывающие пространство задачи — вектором  $V$ .

б) Пространство задачи и пространство шарниров связаны двумя функциями. Обратная задача кинематики описывается функцией  $U=I(V)$ , прямая задача кинематики — функцией  $V=F(U)$ .

в) функции  $I$  и  $F$  задают однозначное соответствие между пространством параметров и пространством задачи и наоборот, т.е. если  $U_i=I(V_i)$ , то  $V_i=F(U_i)$ .

Алгоритм коррекции ошибок решения прямой задачи кинематики приведен на рис.3 и состоит из следующих шагов:

Шаг 1. Обучить нейронную сеть, отображающую вектора пространства шарниров на вектора пространства задачи на основе имеющихся данных. Обозначим ее математически как  $V = F(U)$ .

Шаг 2. Подать на обученную нейронную сеть начальный входной вектор  $U_i$  и получить начальный выходной вектор  $V_i$ .

Шаг 3. Из начального выходного вектора  $V_i$  с помощью уравнений обратной задачи кинематики  $U'_i = I(V_i)$  получить новый входной вектор  $U'_i$ .

Шаг 4. Рассчитать ошибку  $\Delta U_i = U'_i - U_i$ .

Шаг 5. С помощью полученной ошибки изменить начальный входной вектор  $U_i^{new} = U_i - f(\Delta U_i)$ , где  $f(\Delta U_i)$  — постоянная, либо линейная функция, зависящая от  $\Delta U_i$ . Идея состоит в выборе функции таким образом, чтобы обеспечить быструю сходимость к приемлемому решению, однако не слишком быструю, чтобы не вызвать колебательности.

Шаг 6. Подать новый входной вектор  $U_i^{new}$  на обученную нейронную сеть и получить новый выходной вектор  $V_i$ .

Шаг 7. Повторить шаги 2–6 *n* раз. Критерием останова может быть либо фиксированное число *n*, либо совпадение полученного выходного вектора с экспериментальным.

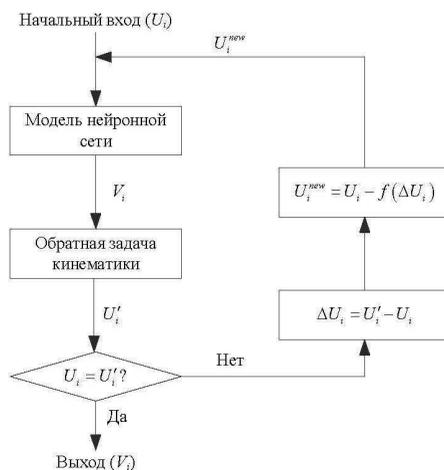


Рисунок 3 – Блок-схема алгоритма компенсации ошибки

Предложенный подход частично совпадает с подходом, изложенным в работе [7], в которой ищется приближенное решение прямой задачи кинематики для параллельного робота со звеньями-пантографами, а не гораздо более распространенными призматическими звеньями. В этой работе прямая задача кинематики решается для вектора углов звеньев ( $\alpha_1, \beta_1, \alpha_2, \beta_2, \alpha_3, \beta_3$ ) и соответствующего вектора параметров задачи ( $x, y, z, \varphi, \theta, \psi$ ), где  $\alpha, \beta, \varphi, \theta, \psi$  — угловые величины, а  $x, y, z$  — линейные. Однако, такая конфигурация, в отличие от конфигурации с призматическими звеньями, мало распространена. Кроме того, в качестве функции  $f(\cdot)$  в этой работе используется коэффициент передачи, равный 1,0, математическое обоснование метода отсутствует. Для математического обоснования принципа действия предложенного алгоритма предположим, что имеется обученная методом обратного распространения ошибки нейронная сеть с по крайней мере одним скрытым слоем и сигмоидальной функцией активации нейронов скрытого и выходного слоев.

Введем следующие обозначения:

$U_i (i=1, \dots, n)$  — вектор входных сигналов, подаваемых на входной слой;

$V_k (k=1, \dots, m)$  — вектор выходных сигналов, получаемых с выходного слоя;

$Z_j (j=1, \dots, p)$  — вектор, представляющий нейроны скрытого слоя;

$b_j$  — смещение, соответствующее  $j$ -ому нейрону скрытого слоя;

$w_{ij}$  — вес связи между нейронами входного и скрытого слоев;

$z\_in_j$  — сумма входных величин нейронов скрытого слоя;

$z_j$  — активация отдельных нейронов скрытого слоя;

$\sigma$  — параметр сигмоидной функции активации.

Предположим, что обученная нейронная сеть преувеличивает значения выходных параметров  $V_i$ . Тогда, после решения обратной задачи кинематики получаем новое значение выхода  $U'_i$ , превышающее начальное значение  $U_i$ . Это обозначает, что при увеличении  $V_i$  увеличивается  $U'_i$ , и наоборот, что вытекает из однозначного соответствия прямой и обратной задач кинематики. Новый входной вектор  $U_i^{new}$  после выполнения шага 5 будет иметь значение, несколько меньшее значения исходного вектора  $U_i$ . Эти новые значения подаются на обученную нейронную сеть. В результате измененные значения выходного вектора дают новый выходной вектор, преувеличенность которого несколько уменьшена. Опишем этот процесс более строго.

Рассмотрим случай преувеличенного выхода, т.е.  $U_i > U_i^{new}$ . Для входного вектора, соответствующего  $U_i$ , шаг прямого распространения выполняется следующим образом:

$$z_{in_j} = b_{0j} + \sum_{i=1}^n U_i w_{ij},$$

а для вектора  $U_i^{new}$  получим

$$z_{in_j}^{new} = b_{0j} + \sum_{i=1}^n U_i^{new} w_{ij}$$

Если  $U_i > U_i^{new}$ , то  $z_{in_j} > z_{in_j}^{new}$ , при условии неизменности всех остальных параметров. Активация каждого элемента скрытого слоя  $z_j$  для  $U_i$  происходит как

$$z_j = f(z_{in_j}) = \frac{2}{1 + e^{-\sigma \cdot z_{in_j}}} - 1.$$

а для  $U_i^{new}$  — как

$$z_j^{new} = f(z_{in_j}^{new}) = \frac{2}{1 + e^{-\sigma \cdot z_{in_j}^{new}}} - 1$$

Следует отметить, что при постоянстве всех остальных параметров имеем  $e^{-\sigma \cdot z_{in_j}} < e^{-\sigma \cdot z_{in_j}^{new}}$ . Это обозначает, что  $\frac{2}{1 + e^{-\sigma \cdot z_{in_j}}} - 1 < \frac{2}{1 + e^{-\sigma \cdot z_{in_j}^{new}}} - 1$  и, соответственно,  $z_j > z_j^{new}$ . Тот факт,

что для  $U_i > U_i^{new}$  получаем  $z_j > z_j^{new}$ , обозначает, что новый выходной вектор, полученный на основе  $U_i^{new}$ , будет иметь меньшее значение,

чем выход, соответствующий  $U_i$ , т.е.  $V_i^{new} > V_i$  и, следовательно,  $V_i^{new}$  ближе к точному решению. Это рассуждение может быть распространено на другие скрытые слои (если они есть) и на входной слой. Для случая преуменьшенного выхода обученной нейронной сети

рассуждения аналогичны. Здесь  $U_i < U_i^{new}$ , следовательно,  $z_j < z_j^{new}$  и, таким образом, конечная оценка выше полученной на предыдущем шаге:  $V_i < V_i^{new}$ .

С учетом определенного числа итераций этого алгоритма ошибка аппроксимации может быть снижена до значений, близких к нулю. На практике в большинстве систем управления, допустимая ошибка должна быть меньше, минимального перемещения линейного привода. Таким образом в качестве критерия завершения алгоритма можно выбрать либо минимальное перемещение линейного двигателя, либо фиксированное количество операций, полученное экспериментально. В любом случае такой алгоритм решения прямой задачи кинематики эффективен.

НС весьма успешно применяются для решения самых разнообразных задач. Привлекательность НС состоит в их высокой надежности, быстродействии и возможности решать плохо формализованные задачи. НС могут производить обучение на экспериментальных данных, что позволяет учесть факторы, влияние которых очень трудно или вообще невозможно выразить аналитически по причине сложности математической модели. Применительно к решению ПЗК можно утверждать, что использование НС позволяет получить выигрыш во времени, по сравнению с численными методами.

Далее проведем экспериментальную проверку решения прямой задачи кинематики для параллельного робота на базе НС прямого распространения. Рассмотрим, как влияют структура, сложность и количество тренировочных примеров на точность решения и время обучения сети. Данные для обучения и тестирования НС, включающие длины штанг и соответствующие им координаты рабочего инструмента, были получены посредством решения обратной задачи кинематики, полученной в аналитическом виде. Таким образом, решения ПЗК каким-либо другим методом не требовалось. На вход обученной сети подавались тестовые значения длин штанг, на выходе получали декартовы координаты. В качестве варьируемых параметров выступали: количество точек тренировочного набора, изменяемое от 60 для 8000, число скрытых слоев, и число нейронов, составляющих скрытый слой. Число нейронов принималось равным 10 и 60.

В результате были построены зависимости, которые показывают связь числа нейронов, количества тренировочных точек с ошибкой по положению и времени обучения (рис. 4-7). Время, затрачиваемое на обучение, сильно зависит от конфигурации используемого компьютера: частоты и разрядности процессора, объема оперативной памяти, поэтому на разных компьютерах будет наблюдаться разный результат.

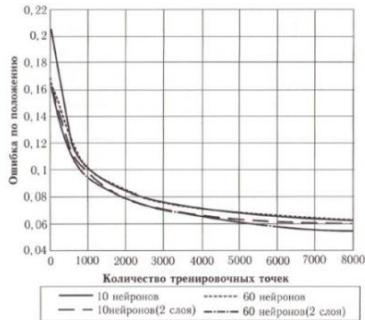


Рисунок 4 – Зависимость ошибки обучения от объема выборки

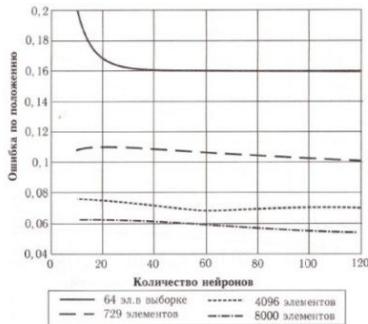


Рисунок 5 – Зависимость ошибки обучения от числа нейронов в скрытом слое

На рис. 6 представлено семейство кривых, каждая из которых получена при количестве числа нейронов скрытого слоя для разного числа точек тренировочной выборки (60, 700, 1100, 8000). Из графиков видно, что при достаточном количестве примеров ошибка незначительно уменьшается с увеличением числа нейронов в скрытом слое. Ошибка на графиках для точек выборки 60 и 700 возрастает, потому что данное число выборок недостаточно. Таким образом, можно сделать вывод о том, что количество точек тренировочного набора должно превышать количество связей в НС в не менее чем в пять раз.

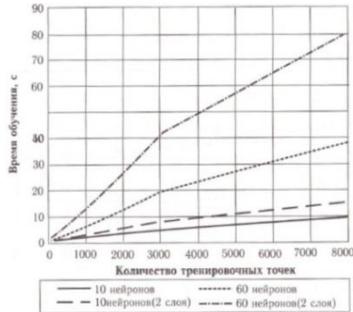


Рисунок 6 – Зависимость времени обучения от объема выборки

Соотношения числа нейронов в скрытом слое и величины ошибки представлены на рис. 7. Данные зависимости указывают, что при увеличении количества нейронов в сети, величина ошибки изменяется мало, а время обучения сети, напротив значительно возрастает (рис. 7).

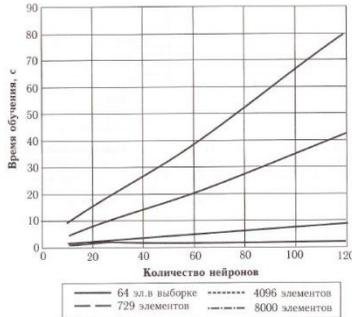


Рисунок 7 – Зависимость времени обучения от числа нейронов в скрытом слое

Данные зависимости указывают, что при увеличении количества нейронов в сети, величина ошибки изменяется мало, а время обучения сети, напротив значительно возрастает (рис. 7). Поэтому, выбирая структуру НС, следует ограничиться небольшим числом нейронов скрытого слоя.

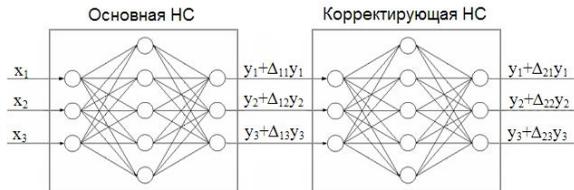


Рисунок 8 – Результирующая структура нейронной сети

Для увеличения точности обучения можно создавать цепочку узлов из так называемых корректирующих НС. Данная сеть будет обучаться на величинах ошибок, которые получены при расхождении между результатами работы основной НС и истинными значениями. При этом корректирующая НС не привязана к конкретному диапазону изменения длин штанг, а определяется лишь границами разброса ошибок, полученных в результате работы первой НС. Принцип работы такой цепочки НС сводится к следующему. На вход первой НС подаются значения длин штанг, на выходе получаем приближенное решение ПЗК. Далее полученные приближенные значения подаются на вход корректирующей сети и на выходе получаем более точное решение.

В таблице 1. приведены сравнительные характеристики работы НС разной структуры. Как видно, применение корректирующих НС является наиболее оптимальным вариантом, т. к. по сравнению с многослойной дает выигрыш и по величине ошибки, и по времени обучения. При этом следует заметить, что скорость прямого прохода по сети с корректирующей структурой будет также выше. Это объясняется тем, что каждая из сетей (основная и корректирующая) являются однослойными и по структуре являются более простыми, чем одна многослойная сеть.

Таблица 1 – Оценка точности решения прямой задачи кинематики с помощью НС

Структура НС	Характеристики	Время обучения НС, с	Ошибка обучения
НС с одним скрытым слоем	20 нейронов, 8000 точек	20,045	0,07984
Многослойная НС	3 слоя, 20 нейронов, 8000 точек	124,325	0,04805
Корректирующая НС	20 нейронов, 3 корр. сети, 8000 точек	102,034	0,00102

Для управления параллельным роботом по обратной связи необходимо решить прямую задачу кинематики. Из-за отсутствия ее решения в явном виде, в литературе, посвященной данному предмету, предлагается множество способов ее решения. Для использования решения прямой задачи кинематики в управлении параллельным роботом важны точность, единственность решения и возможность применения в реальном масштабе времени. В данной работе предлагается метод решения прямой задачи кинематики, основанный на

применении искусственных нейронных сетей, математической модели решения обратной задачи кинематики и алгоритма коррекции ошибок.

Стоит отметить, что предложенный метод решения прямой задачи кинематики допускает аппаратную реализацию, что потенциально ускорит вычисления и, соответственно, существенно сократит время, необходимое для получения достаточно точного приближения к решению прямой задачи кинематики для параллельных роботов.

*Исследования выполнены при финансовой поддержке РФФИ и Правительства Белгородской области в рамках проекта №14-41-08058 «р\_офи\_м», с использованием оборудования ЦВТ БГТУ им. В.Г. Шухова.*

#### **Список литературы:**

1. Ascher M. Cycling in the Newton–Raphson algorithm // Int J Math EducSciTechnol. 1974. Vol. 5, Iss. 2. P. 229-235.
2. Geng Z, Haynes L.S. Neural network solution for the forward kinematics problem of a Stewart platform // Robot ComputIntegrManuf. 1992. Vol. 9, Iss. 6. P. 485-495.
3. i P., Wu H. A closed-form forward kinematics solution for the 6–6<sup>p</sup> Stewart platform // IEEE Trans Robot Automat. 2001. Vol. 17, Iss. 4. P. 522-526.
4. Merlet J.–P. Solving the forward kinematics of a Gough-type parallel manipulator with interval analysis // Int J Robot Res. 2004. Vol. 23. Iss. 3. P. 221-235.
5. Parikh P.J., Lam S.S.Y. A hybrid strategy to solve the forward kinematics problem in parallel manipulators // IEEE Trans Robot. 2005. Vol. 21. Iss. 1. P. 18-25.
6. Wampler C.W. Forward displacement analysis of general six-in-parallel SPS (Stewart) platform manipulators using soma coordinates // Mech Mach Theory. 1996. Vol. 31. Iss. 3. P. 331-337.
7. Yee C.S., Lim K.B. Forward kinematics solution of Stewart platform using neural networks // Neurocomputing. 1997. Vol. 16, Iss. 4. P. 333-349.
8. Zanganeh K.E. Angeles J. Real-time direct kinematics of general six-degree-of-freedom parallel manipulators with minimum-sensor data // J Robot Syst. 1995. Vol. 12, Iss. 12. P. 833-844.