

УДК 621.81

В. М. Пашкевич

МЕТОДОЛОГИЯ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ МЕХАНИЗМОВ НА ОСНОВЕ ИСПОЛЬЗОВАНИЯ ФУНКЦИОНАЛЬНЫХ СЕМАНТИЧЕСКИХ СЕТЕЙ И ПРОЦЕДУРЫ ОБУЧЕНИЯ

UDC 621.81

V. M. Pashkevich

METHODOLOGY OF COMPUTER-AIDED DESIGN OF MECHANISMS BASED ON THE USE OF FUNCTIONAL SEMANTIC NETWORKS AND THE LEARNING PROCEDURE

Аннотация

Рассматривается методология автоматизированного проектирования механизмов на базе функциональных семантических сетей, обеспечивающая «открытую» архитектуру вычислений и управления расчетными параметрами.

Ключевые слова:

расчет и проектирование, детали и узлы приводов машин, функциональные семантические сети, отношения, поиск оптимальных решений, обучение.

Abstract

The paper considers the methodology of computer-aided design of mechanisms based on the use of functional semantic networks, which provides an open architecture of calculations and design parameters control.

Key words:

calculation and design, components and assemblies of machine drives, functional semantic networks, relations, search for optimal solutions, machine learning.

Введение

Современный подход к проектированию механизмов включает ряд типовых действий среди которых выделяют энерго-кинематический расчет, прочностные расчеты деталей, расчет размерных цепей и др. [1].

Как правило, каждый из этих расчетов рассматривается как некоторый этап общего процесса проектирования объекта, в связи с чем полученные результаты часто используются как промежуточные – для дальнейших этапов, входящих в общую последовательность проектирования. В то же время каждый из таких этапов считается условно неза-

висимым от предыдущих.

Данный подход, однако, не учитывает функциональных взаимосвязей, возникающих между расчетными параметрами ввиду наличия указанной последовательности. Так, например, результаты энергетического (силового) расчета привода влияют на размерные параметры деталей машины или агрегата. Изменение конструктором компоновочных размеров, как правило, приводит к изменению силовых характеристик привода, которое может быть скомпенсировано размерами деталей механизма, изменением их материала и т. д. Из-за отсутствия гибкости в учете

таких взаимосвязей между параметрами конструкции и результатами расчетов результат проектирования может обеспечить завышение массогабаритных характеристик конструкций или, наоборот, их низкую прочность. Ввиду этого процесс проектирования в большинстве случаев носит циклический характер,

когда требуемые результаты получают путем последовательных приближений и корректировок.

Структурная схема, иллюстрирующая такую методику, представлена на рис. 1. Стрелки на схеме показывают реализуемую последовательность действий.

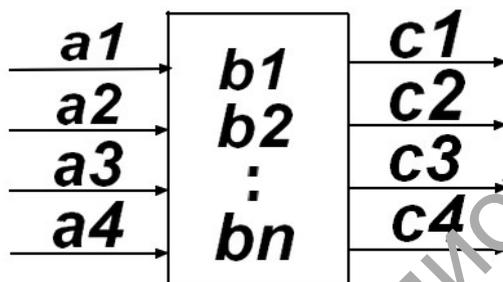


Рис. 1. Структурная схема, иллюстрирующая традиционную методику расчета деталей и узлов механизмов

В качестве основы для расчета принимаются так называемые входные, или исходные, данные $a_1 \dots a_m$. На базе положенного в основу конкретной методикой алгоритма они, при необходимости, преобразуются в промежуточные результаты расчета $b_1 \dots b_n$. Далее, в соответствии с используемым алгоритмом, получают окончательные результаты $c_1 \dots c_k$ – выходные данные.

Описанная последовательность является методологической основой для каждого типа из вышеприведенных расчетов.

Несмотря на простоту такого подхода и его логическую очевидность для конечного пользователя, он обладает рядом недостатков.

Как правило, связь между параметрами a_i и c_j нелинейная, а описывающие ее уравнения часто являются трансцендентными. В том случае, если результат расчетов не устраивает проектировщика, в промежуточные результаты или исходные данные вносятся поправки, соответствующие степени ука-

занной невязки, после чего расчет повторяют. Последовательность корректировок позволяет получить требуемый результат за некоторое число итераций, которое может быть значительным.

Кроме указанного обстоятельства, проектировщику в большинстве случаев недоступны для изменения промежуточные результаты расчетов c_k , т. к. для этого требуется решение трансцендентных уравнений.

Данные обстоятельства показывают, что традиционная методология проектирования опирается на «закрытую» схему вычислений, в которой расчет ограничивается жестким и, как правило, последовательным сценарием.

Альтернативу такому подходу может составить методология проектирования с «открытой» архитектурой, где для изменений доступны не только выходные данные, но и результаты промежуточных вычислений.

При этом оператор может заранее задавать целевые величины c_k , а также оперировать промежуточными результа-

тами расчетов b_j таким образом, если бы они являлись входными параметрами.

Структурная схема, иллюстриру-

ющая идею таких вычислений, обеспечивающих «открытый» характер проектирования, приведена на рис. 2.

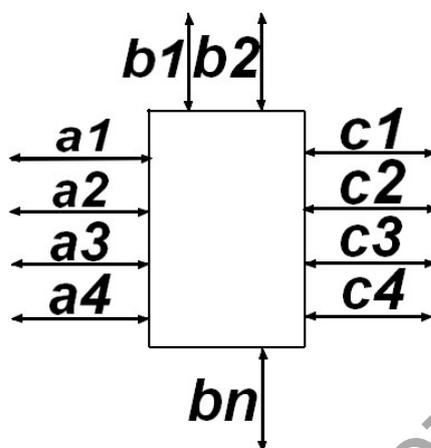


Рис. 2. Структурная схема, иллюстрирующая «открытую» архитектуру проектирования деталей и узлов механизмов

Каждый из параметров a_i , b_j и c_k представляет собой равнозначные величины, изменение каждой из которых приводит к соответствующей корректировке значений других параметров в соответствии с набором заранее выбранных функциональных зависимостей между ними. Таким образом, в приведенной схеме отсутствует понятие входных и выходных параметров, т. е. каждый из них может выступать как в роли входного, так и выходного. Ввиду этого на схеме используются двухсторонние стрелки, иллюстрирующие возможность каждого из параметров выступать в роли как входных, так и выходных данных.

Описанный подход позволяет в единой вычислительной среде объединить все приведенные этапы расчета, т. е. решение задачи приобретет комплексный характер при тесном взаимодействии ее параметров, обеспечивающем как решение прямых, так и обратных задач.

Так, например, при изменении геометрических параметров (размеров, входящих в размерные цепи конструкции) могут быть автоматически ском-

пьютированы прочие компоновочные размеры элементов узла или, например, при изменении материала какого-либо элемента конструкции могут быть соответствующим образом скорректированы размеры прочих ее элементов для обеспечения требуемых массогабаритных характеристик.

Следует отметить, что элементы такого подхода уже реализуются частично в некоторых CAD-системах в форме процедур адаптивной сборки. В этой процедуре при изменении размеров одного из элементов сборочной единицы автоматически изменяются и размеры сопряженных элементов [2].

Однако реализован такой подход только для этапа геометрической компоновки механизма, т. е. только для расчета соответствующих размерных цепей.

Представленный подход базируется на использовании технологии функциональных семантических сетей и позволяет реализовать описанную «открытую» архитектуру проектирования деталей и узлов механизмов с учетом энергетических, размерных, прочностных и др. взаимосвязей.

Основная часть

Описанная во введении методология вычислений опирается на представлении совокупности расчетных функций вида

$$c_k = q_k(a_i; b_j) \quad (1)$$

в форме отношения

$$f(a_i; b_j; c_k) = 0. \quad (2)$$

При этом изменение каждого из параметров отношения должно приводить к изменению других его параметров для обеспечения справедливости выражения (2).

Реализовать данную концепцию можно на основе использования функциональных семантических сетей [3], представляющих собой логическую схему, описывающую функциональные взаимосвязи между аргументами совокупности указанных математических зависимостей.

Семантическую сеть удобно приводить в виде графа, в котором вершины отображают параметры и отношения, а ребра (дуги) – наличие связей между ними. Таким образом семантическую сеть можно представить тройкой объектов (V, E, θ) , где V – множество вершин графа; E – множество ребер; θ – функция инцидентности, которая каждому элементу множества E ставит в соответствие пару элементов из множества V [4].

У функциональной семантической сети множество вершин V является объединением непересекающихся подмножеств P и R , т. е.

$$V = P \cup R, \quad (3)$$

где P – множество параметров, включающее исходные, промежуточные и выходные параметры рассчитываемой задачи; R – множество отношений, определяющих расчетные зависимости решаемых с помощью семантической

сети задач, т. е.

$$R_i = \{ f(P_1, \dots, P_j) = 0 \}, \quad (4)$$

где P_j – элемент множества параметров P сети; R_i – i -е отношение сети, определяющее функциональные зависимости между параметрами P_1, \dots, P_j .

На рис. 3 приведена возможная структура функциональной семантической сети, представляющей собой в общем случае двудольный граф и состоящей из N отношений $f(P_1, \dots, P_j)$ между M параметрами P_i . При этом вершины-кружки являются параметрами проектируемой системы, а вершины-прямоугольники содержат расчетные зависимости (отношения).

Для двудольного графа, представленного на рисунке, функция инцидентности имеет вид:

$$\theta(e) = \{ (P_i, R_j) \mid P_i \in P, R_j \in R \}, \quad (5)$$

что означает, что любое ребро сети соединяет некоторую вершину из множества P с некоторой вершиной из множества R .

В функциональной семантической сети под отношением R , определенном на k своих атрибутах (значениях параметров P_1, \dots, P_k), понимается некоторое подмножество декартова произведения доменов D_i (множеств возможных значений), т. е.

$$R(P_1, \dots, P_k) \subseteq D_1 \times \dots \times D_k, \quad (6)$$

где $P_j \in D_j$ ($j = 1, 2, \dots, k$).

Отношение R указывает на существование определенных зависимостей между значениями атрибутов в пространстве, определенном декартовым произведением их доменов.

Любой кортеж, удовлетворяющий отношению R , является его элементом, что эквивалентно условию

$$(P_1, \dots, P_j, \dots, P_k) \in R. \quad (7)$$

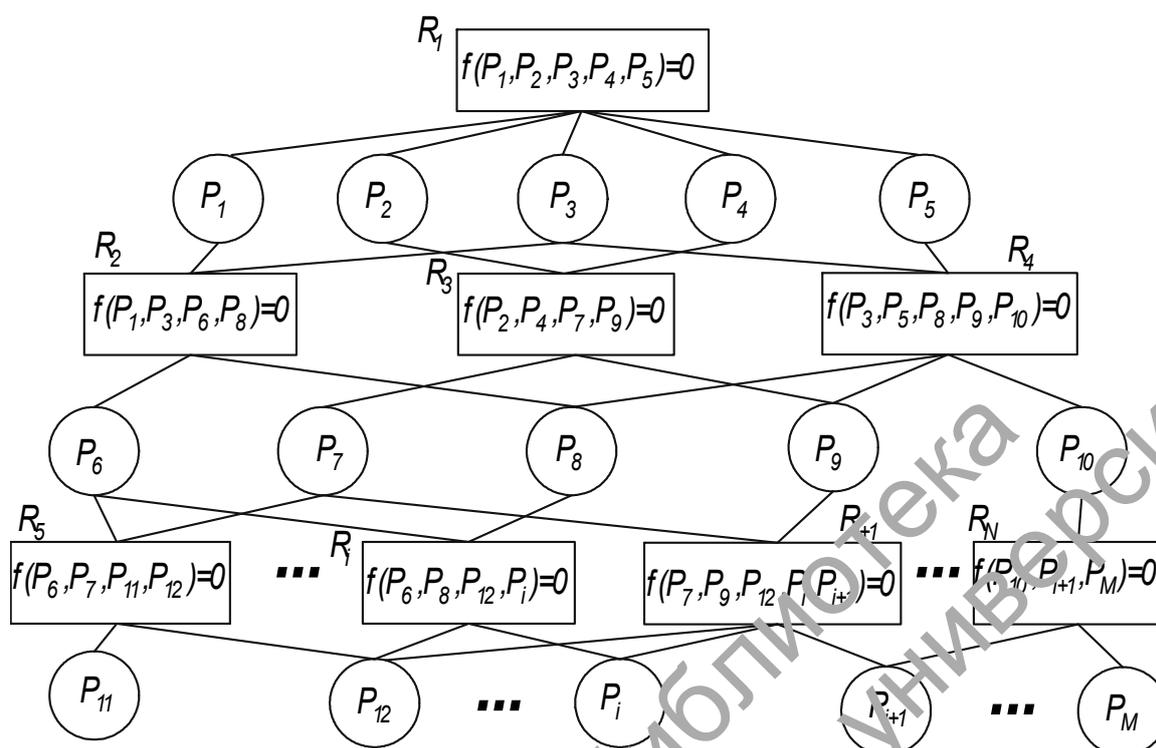


Рис. 3. Пример структуры функциональной семантической сети

Это условие означает, что любое изменение одного из параметров P_j должно автоматически приводить к функциональным изменениям остальных параметров, обеспечивающим выполнение условия (4). Таким образом, семантической сетью будет поддерживаться отбор допустимых сочетаний расчетных параметров P .

Следует отметить, что функциональная семантическая сеть является неориентированным графом, т. к. только при пошточке условия задачи станет известно, какие параметры являются входными, а какие выходными для отношения сети.

Данное обстоятельство позволяет организовать поиск решений на семантической сети следующим образом.

Выделим в кортеже какой-либо атрибут P_j , обозначив его через y . Отношение будет функциональным, если для всего множества кортежей кортежи $(P_1, \dots, P_{j-1}, P_{j+1}, \dots, P_k)$ будут различны-

ми. В этом случае кортежу $(P_1, \dots, P_{j-1}, P_{j+1}, \dots, P_k)$ соответствует не более одного значения $P_j = y$ такого, что $(P_1, \dots, P_j, \dots, P_k) \in R$. Следовательно, значение $P_j = y$ однозначно определяется значениями кортежа $(P_1, \dots, P_{j-1}, P_{j+1}, \dots, P_k)$ и соответствует функции

$$y = F^{(P_j)}(P_1, \dots, P_{j-1}, P_{j+1}, \dots, P_k), \quad (8)$$

называемой разрешением функционального отношения для атрибута $P_j = y$.

При одном выделенном атрибуте y ранг отношения полагается равным единице. Если взять отношение ранга z , то будет определено z функций, зависящих от $k - z$ переменных:

$$y_i = F_i(P_1, \dots, P_{k-z}), \quad (9)$$

где $i = 1, 2, \dots, z$.

Поиск решения в данном случае заключается в поиске кортежа (P_1, \dots, P_{k-z}) , удовлетворяющего одновременно всему множеству функций u_i .

Если отношение имеет только одно разрешение, то используется так называемый алгоритм паросочетаний, когда устанавливается взаимно однозначное соответствие между обоими типами вершин семантической сети.

Если же ранг отношений больше единицы, то для нахождения минимально замкнутой системы отношений, необходимой для решения поставленной задачи, требуются более сложные алгоритмы, в том числе переборные.

На следующем этапе у отношений, входящих в минимально замкнутую си-

стему, выявляются входы и выходы. В результате этого происходит преобразование отношений в соответствующие функции:

$$R = \{f(P_1, \dots, P_k) = 0\} \rightarrow$$

$$F^{(P_i)}(P_1, \dots, P_{j-1}, P_{j+1}, \dots, P_k). \quad (10)$$

Пример преобразования отношения R , состоящего из k параметров, в функцию F , обеспечивающую нахождение параметра P_i , показан на рис. 4.

Далее, после преобразования всех отношений R в соответствующие функции F , формируется цепочка функций для решения поставленной задачи (рис. 5).

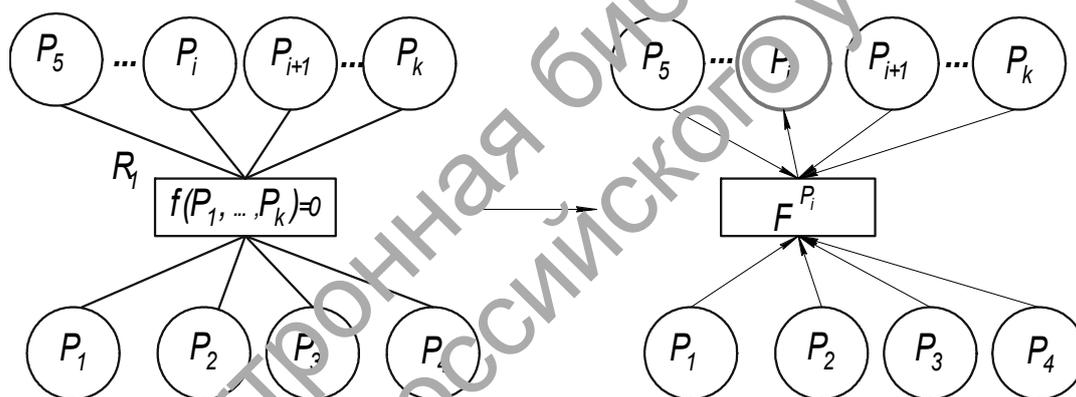


Рис. 4. Пример преобразования отношения в функцию

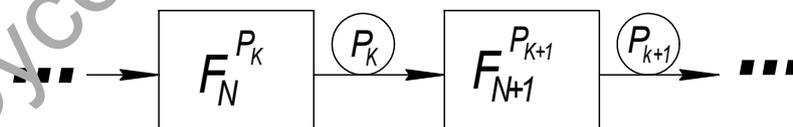


Рис. 5. Формирование цепочки функций

В результате этого происходит преобразование неориентированного двудольного графа отношений в ориентированный граф решения задачи, в ко-

тором вершины-кружки являются входными параметрами проектируемой системы, а вершины-прямоугольники содержат функции, в которые отношения

между параметрами проектируемой системы были преобразованы. Вычисление значений функций относительно входных параметров позволяет определить выходные параметры системы.

Следует отметить, что преобразование неориентированного двудольного графа отношений в ориентированный граф решения задачи не всегда происходит по линейному сценарию, при этом в цепочке функций могут появляться так называемые стыки, вилки и циклы, для устранения которых могут использоваться нетривиальные алгоритмы [3, 5].

Описанный алгоритм поиска решения позволяет организовать процедуру автоматизированного проектирования механизмов для любых сочетаний их входных и (или) выходных параметров или установить ее принципиальную невозможность при условии неполноты входной информации, обеспечивающей преобразование отношений семантической сети в функции. Указанный алгоритм использует разработанная в Белорусско-Российском университете интеллектуальная система SEMANTIC для проектирования технологических приспособлений на базе функциональных семантических сетей [6].

Другой особенностью системы является возможность ее обучения или самообучения.

Результаты обучения при этом рассматриваются как набор исходных, промежуточных и выходных параметров P_i задачи, успешно решенной ранее под управлением оператора. Такой набор параметров (прототип решения) может быть получен также на основе процедуры самообучения системы, когда генератором задач формируются случайные сочетания исходных данных для проектирования механизма, а результаты автономного решения соответствующей задачи размещаются в базе знаний системы [7].

Наличие указанной базы позволяет организовать процедуру автоматизи-

рованного проектирования по аналогии, в соответствии с хранящимися в базе знаний прототипами. Суть такой процедуры сводится к решению задачи метрического распознавания образов [7] и отражена на рис. 6.

Очевидно, что каждый прототип решения может рассматриваться как набор координат P_i многомерной точки в факторном пространстве задачи общей размерности n . Так, на рис. 6 отображен поиск прототипа в двухмерном пространстве параметров P_1 и P_2 . В базе знаний системы хранятся решения-прототипы Z_1, Z_2, \dots, Z_k . Для условия поставленной задачи Z_0 рассчитываются евклидовы расстояния до прототипов Z_i (только по координатам, соответствующим известным входным параметрам), из которых выбирается ближайший, удовлетворяющий условию

$$L_{opt} = \sqrt{\sum_{i=1}^k (P_{0i} - P_{ki})^2} \rightarrow \min. \quad (11)$$

Отобранный прототип принимается системой как результат предварительного расчета механизма.

В том случае, если заданный критерий качества или требуемые выходные параметры обеспечены, прототип принимается в качестве окончательного результата проектирования. Если же решение не получено, программой [6] производится поиск решения в окрестностях прототипа на основе метода случайного поиска с возвратом [4].

Суть указанного метода проиллюстрирована на рис. 7.

В данном алгоритме поиск значений параметров P_1, P_2, \dots, P_n , доставляющих экстремум (максимум) функции $Z = f(P_1, P_2, \dots, P_n)$, начинается из исходной точки $Z_{исх}$, соответствующей пробному решению, в которой определяется значение целевого критерия $Z_{исх}(P_1, P_2, \dots, P_n)$.

Из n переменных P_1, P_2, \dots, P_n вы-

бирается одна, например P_1 , значения же остальных фиксированы.

Далее совершается переход от исходной к новой допустимой точке Z_1 и для нее оценивается значение целевого

критерия $Z_1(P_1, P_2, \dots, P_n)$, которое сравнивается со значением, найденным в точке $Z_{исх}(P_1, P_2, \dots, P_n)$.

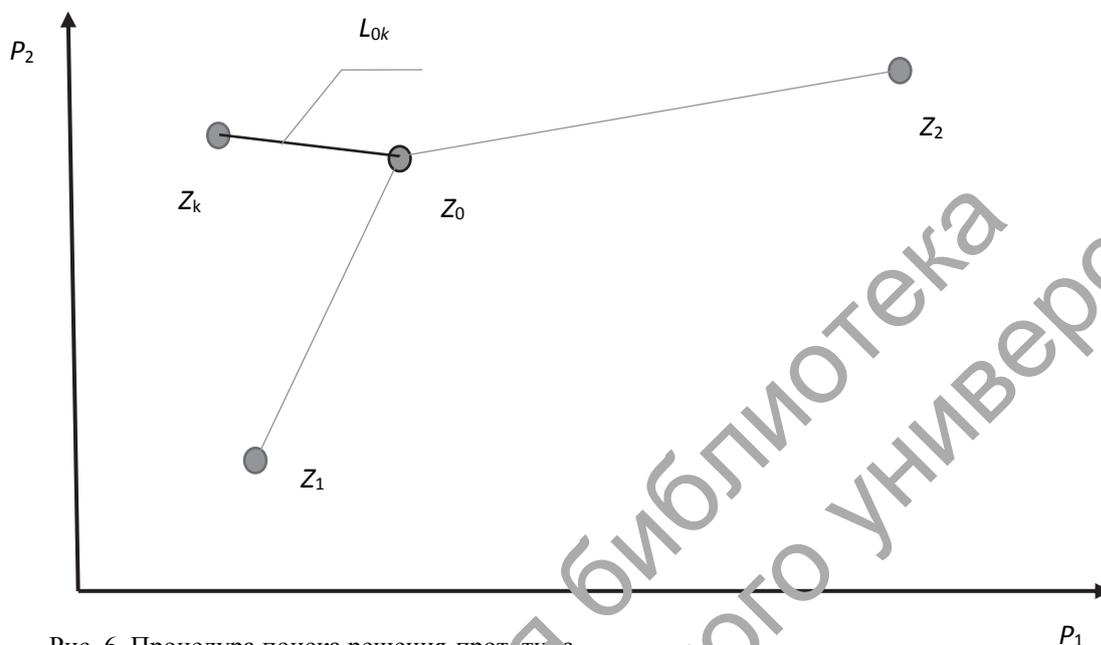


Рис. 6. Процедура поиска решения-прототипа

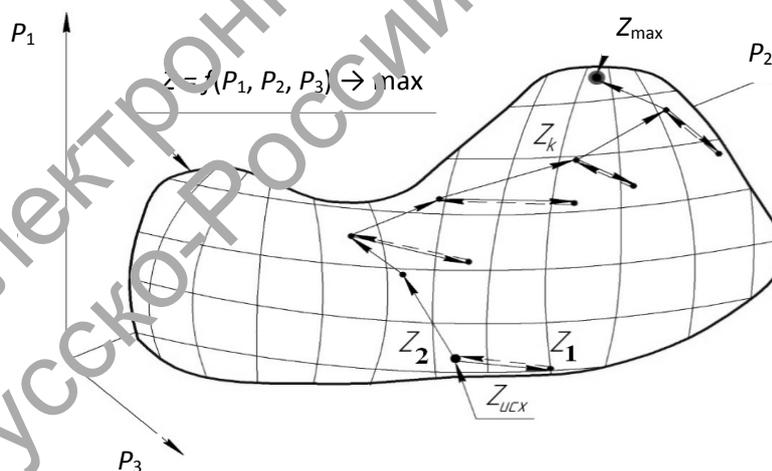


Рис. 7. Схема случайного поиска с возвратом

Переход к новой точке осуществляется в соответствии с формулой

$$Z_1(P_{1(k+1)}, P_2, \dots, P_n) = Z_1(P_{1k} \pm a_k r_k, P_2, \dots, P_n), \quad (12)$$

где a_k – величина k -го шага, определяемая случайным образом; r_k – единичный вектор, в направлении которого производится этот шаг.

Если оказывается, что

$$Z_1(P_1, P_2, \dots, P_n) > Z_{\text{исх}}(P_1, P_2, \dots, P_n), (13)$$

то совершается переход из точки $Z_{\text{исх}}$ в точку Z_1 , после чего последняя становится новой исходной для продолжения поиска. Если же условие (13) не выполняется, то осуществляется возврат в исходную позицию.

Далее в качестве пробных выбираются точки, для которых последовательно изменяются остальные координаты (параметры P_1, P_2, \dots, P_n). Этот процесс продолжается до тех пор, пока сохраняется возможность улучшения це-

$$Z(P_1, \dots, P_{ik}, \dots, P_n) = \begin{cases} Z(P_1, \dots, P_{i(k-1)}, \dots, P_n), \\ \text{если } Z(P_1, \dots, P_{i(k-1)}, \dots, P_n) < Z(P_1, \dots, P_{ik}, \dots, P_n); \\ Z(P_1, \dots, P_{ik}, \dots, P_n), \\ \text{если } Z(P_1, \dots, P_{ik}, \dots, P_n) \leq Z(P_1, \dots, P_{i(k-1)}, \dots, P_n). \end{cases} (14)$$

Эффективность представленной методологии была подтверждена на задачах, связанных с проектированием станочных приспособлений [4]. Ее применение позволило обеспечить снижение массогабаритных показателей технологического оснащения на 15...25 % за счет использования функциональных взаимосвязей между параметрами проектируемых механизмов.

Заключение

1. Рассмотрены методологические основы построения автоматизированной процедуры проектирования и расчетов машин на базе функциональных семан-

тических сетей. Показано, что указанный подход обеспечивает «открытую» архитектуру взаимодействия между расчетными параметрами задач и может быть использован компьютерными интеллектуальными системами.

2. Описана процедура обучения (самообучения) компьютерной системы, обеспечивающая автономное решение задач проектирования по аналогии с прототипами, хранящимися в базе знаний.

3. Приведен алгоритм случайного поиска с возвратом, обеспечивающий поиск решений на функциональных семантических сетях в автономном режиме.

СПИСОК ЛИТЕРАТУРЫ

1. **Иванов, М. Н.** Детали машин : учебник для студентов машиностроительных спец. вузов / М. Н. Иванов, В. А. Финогенов. – 10-е изд., испр. – М. : Высш. шк., 2006. – 408 с.
2. Адаптивное моделирование в Autodesk Inventor / САПР и графика. – 2003. – № 10. – С. 25–27.
3. **Поспелов, Г. С.** Искусственный интеллект – основа новой информационной технологии / Г. С. Поспелов. – М. : Наука, 1988. – 280 с.
4. **Пашкевич, В. М.** Функциональные семантические сети для обеспечения точности механической обработки : монография / В. М. Пашкевич, М. Н. Миронова. – Могилев : Белорус.-Рос. ун-т, 2015. – 210 с.

5. **Пашкевич, В. М.** Проблема возникновения стыков при проектировании станочных приспособлений на основе использования семантических сетей / В. М. Пашкевич, М. Н. Миронова // Сб. науч. тр. Междунар. Балт. ассоц. машиностроителей ; под ред. В. Н. Тилипалова. – Калининград, 2009. – С. 98–103.

6. **Миронова, М. Н.** Интеллектуальная система для расчета станочных приспособлений / М. Н. Миронова // Вестн. Полоц. гос. ун-та. – 2010. – № 2. – С. 26–33.

7. **Пашкевич, В. М.** Самообучающиеся системы искусственного интеллекта в машиностроении : монография / В. М. Пашкевич, Ж. А. Мрочек. – Могилев : МГТУ, 2003. – 434 с.

Статья сдана в редакцию 21 июля 2016 года

Виктор Михайлович Пашкевич, д-р техн. наук, доц., Белорусско-Российский университет.
E-mail: pvm@bk.ru.

Viktor Mikhailovich Pashkevich, DSc (Engineering), Associate Prof., Belarusian-Russian University.
E-mail: pvm@bk.ru.