

Решение задач оптимизации и проектирования схем ЭВА на основе использования гибридных интеллектуальных методов

Л.А. Гладков, А.Е. Лисовцова

Технологический институт Южного федерального университета, leo@tsure.ru, gu-siafluffy@bk.ru

Аннотация — В работе рассмотрены основные аспекты применения нечетких генетических алгоритмов для решения задач оптимизации и проектирования. Приведена обобщенная структура нечеткого логического контроллера и описана основная идея его применения. Рассмотрены основные направления и проблемы создания нечетких генетических алгоритмов.

Ключевые слова — Нечеткий генетический алгоритм, нечеткий логический контроллер, вещественное кодирование, гибридные генетические операторы на основе методов локального поиска.

I. ВВЕДЕНИЕ

В настоящее время проблемы повышения качества и сложности создаваемых автоматизированных устройств и систем в различных областях науки и техники связывают с возможностью их интеллектуализации, т.е. придания создаваемым техническим объектам и системам ряда функций обычно выполняемых человеком. Такими функциями можно считать работу по анализу и принятию решений в условиях неполной, нечеткой или противоречивой входной информации, поиск и выделение в массивах входной информации ранее неизвестных, нетривиальных, но практически полезных закономерностей, их оценка и интерпретация.

Одним из наиболее эффективных на сегодняшний день инструментариев при решении NP-полных комбинаторных задач проектирования являются нечеткие гибридные методы, модели и алгоритмы. К числу таких методов относятся генетические, эволюционные, бионические, адаптивные и другие методы поиска.

На практике разработка математически обоснованных четких моделей и методов, либо экономически неприемлема, либо практически нереализуема. В то же время системы, функционирующие на основе использования интегрированных, нечетких гибридных механизмов и моделей прекрасно зарекомендовали себя при решении такого рода задач, и представляют собой наиболее разумный компромисс.

В этой связи перспективным представляется использование для решения данной задачи гибридных

методов, относящихся к вычислительному интеллекту: нечеткие модели и методы, эволюционные и генетические алгоритмы. Они позволяют эффективно работать с нечеткой, плохо формализованной информацией и, в то же время, имеют серьезную математическую основу, обеспечивающую достаточный запас прочности.

II. НЕЧЕТКИЕ ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ

С концептуальной точки зрения, создаваемые системы поиска и оптимизации проектных решений можно классифицировать как смешанные искусственные системы, т.е. системы созданные человеком и объединяющие искусственные и естественные подсистемы [1]. Они также являются целеориентированными системами, т.е. системами основой функционирования которых являются факторы целесообразности [2].

Модель, соответствующая уровню бионических систем может быть представлена следующим образом [3]:

$$SYS = (GN, KD, MB, EV, FC, RP),$$

где GN – генетическое начало (создание стартового множества решений);

KD – условия существования;

MB – обменные явления (эволюционные и генетические операторы);

EV – развитие (стратегия эволюционирования);

FC – функционирование;

RP – репродукция.

В настоящее время наибольшие успехи в интеграции систем и подходов нечеткой логики и генетических алгоритмов (ГА) достигнуты в следующих двух областях [4]:

1) Применение механизмов генетических и эволюционных алгоритмов для решения проблем оптимизации и поиска в условиях нечеткой, неопределенной или недостаточной информации об объекте, параметрах и критериях решаемой задачи, совместно с использованием систем нечетких логики.

2) Использование нечетких инструментов и методов, основанных на нечеткой логике для моделирова-

ния различных компонентов и операторов генетических алгоритмов, а также для адаптации и управления основными параметрами генетического алгоритма для динамической настройки и улучшения работы ГА.

Как правило, под нечетким генетическим алгоритмом (НГА) понимают гибридные структуры, относящиеся ко второй области. Таким образом, нечеткий генетический алгоритм можно определить как алгоритм, сочетающий поисковые возможности генетических алгоритмов и возможности математического аппарата нечеткой логики. Рассмотрим подробнее основные направления создания и модификации НГА.

Контроль и адаптация основных управляющих параметров генетического алгоритма. Для контроля и динамического изменения соответствующих параметров генетического алгоритма в систему вводится Нечеткий Логический Контроллер (НЛК), который, используя опыт и знание экспертов в рассматриваемой области, соответствующим образом динамически изменяет параметры генетического поиска в ходе выполнения ГА для того, чтобы избежать проблемы преждевременной сходимости.

Наличие нечеткого логического контроллера является важной особенностью нечеткого генетического алгоритма. НЛК преобразует заданные параметры к нечеткому виду (операция *фаззификации* (*fuzzification*)), затем на основе имеющихся в системе знаний и правил определяет управляющее воздействие и возвращает скорректированные значения контрольных параметров [5].

Система выработки правил на основе знаний экспертов и использования рассуждений, делает определенный вывод, который после выполнения обратного преобразования (операция *дефаззификации* (*defuzzification*)), превращается из нечеткого правила в воздействие на параметры алгоритма. Изменение параметров алгоритма влечет за собой изменение процесса поиска и текущих результатов, которые затем в блоке фаззификации из переменных состояния преобразуются в нечеткие множества. Обобщенную структуру НЛК можно представить следующим образом (Рис. 1) [4].

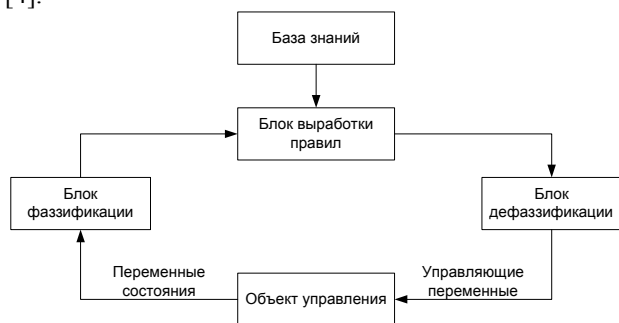


Рис. 1. Обобщенная структура нечеткого логического контроллера

База правил, иногда называемая лингвистической моделью, представляет собой множество нечетких правил $R^{(k)}$, $k = 1, \dots, N$ вида:

$R^{(k)} : IF(x_1 \text{ это } A_1^k \text{ AND } x_2 \text{ это } A_2^k \dots \text{ AND } x_n \text{ это } A_n^k) \text{ THEN}(y_1 \text{ это } B_1^k \text{ AND } y_2 \text{ это } B_2^k \dots \text{ AND } y_m \text{ это } B_m^k).$

где N – количество нечетких правил;

A_i^k – нечеткие множества $A_i^k \subseteq X_i \subset R$, $i=1\dots n$.

B_j^k – нечеткие множества $B_j^k \subseteq Y_j \subset R$, $j=1\dots m$;

$x_1, x_2, \dots, x_n$ – входные переменные лингвистической модели, причем

$$(x_1, x_2, \dots, x_n)^T = x \in X_1 \times X_2 \times \dots \times X_n$$

$y_1, y_2, \dots, y_m$ – выходные переменные лингвистической модели, причем

$$(y_1, y_2, \dots, y_m)^T = y \in Y_1 \times Y_2 \times \dots \times Y_m$$

НЛК оперирует нечеткими множествами. Поэтому значение входного параметра НЛК подлежит операции *фаззификации*, в результате которой ему будет сопоставлено нечеткое множество.

На выходе блока выработки решения формируется одно или несколько нечетких множеств с соответствующими функциями принадлежности. Соответственно необходимо решить задачу преобразования этих результирующих нечетких множеств (нечеткого множества) в управляющее воздействие на объекты управления НЛК. Такое преобразование называется *дефаззификацией* (*defuzzification*).

Наиболее распространенными являются следующие методы дефаззификации:

1. Метод дефаззификации по среднему центру (center average defuzzification).
2. Метод дефаззификации по сумме центров (center of sums defuzzification).
3. Метод центра тяжести (center of gravity method, center of area method).
4. Метод максимума функции принадлежности.

Известно, что эффективность работы генетического алгоритма зависит от выбора его параметров. Использование нечетких логических контроллеров для изменения параметров генетического алгоритма позволяет улучшить работу генетического алгоритма за счет более осторожного, взвешенного и целенаправленного контроля.

В генетических алгоритмах (ГА) управляющими параметрами, как правило, являются значения вероятности выполнения генетических операторов кроссинговера (P_c) и мутации (P_m), а также размер популяции. Идея состоит в том, чтобы использовать НЛК, на входе которого – любая комбинация критериев качества работы ГА или текущих параметров контроля, а на выходе новые параметры контроля ГА. Текущие критерии качества работы ГА направляют в НЛК, который рассчитывает новые значения управляющих параметров ГА. Вероятности кроссинговера и мутации могут определяться НЛК исходя из оценки не

всей популяции, а по определенной выборке решений, учитывающей значения функции пригодности и разнообразия популяции. Также могут одновременно использоваться несколько НЛК [6].

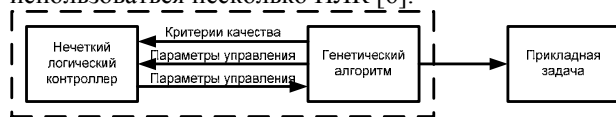


Рис. 2. Схема взаимодействия блоков нечеткого генетического алгоритма

Критерии остановки. Такие параметры как степень истинности, оценка достоверности логической функции могут использоваться для получения оптимальных решений с заданной точностью, для задания и регулирования критериев остановки генетического алгоритма.

III. КОДИРОВАНИЕ РЕШЕНИЙ

Классическое бинарное представление решений может быть преобразовано в нечеткое представление, где гены принимают значения в интервале между нулем и единицей. Это позволяет выражать более сложные особенности как генотипа, так и фенотипа различных решений популяции наподобие тех, которые встречаются в природе.

Также необходимо отметить модификации генетических алгоритмов, использующие хромосомы с вещественным кодированием. Кодирование на основе вещественных чисел наиболее эффективно при решении задач параметрической оптимизации с переменными заданными в непрерывных областях значений, а также при проектировании аналоговой электронной аппаратуры. В таком кодировании хромосома-решение представляет собой вектор чисел с плавающей запятой, причем размер чисел соответствует длине вектора. Генетические алгоритмы, основанные на вещественном представлении чисел, называют генетическими алгоритмами с вещественным кодированием (*real-coded GAs – RCGA*).

Применение вещественного кодирования решений при оптимизации и проектировании, позволяет строить множество различных специальных операторов поиска, использующих вероятностные критерии оценки разнообразия популяции решений и «близости» различных решений по отношению друг к другу, основанных на некоторой мере расстояния между родителями [7].

III. НОВЫЕ МОДИФИКАЦИИ ОПЕРАТОРА КРОССИНГОВЕРА

Нечеткие правила можно использовать для создания эффективных операторов кроссинговера, которые регулируют уровень разнообразия популяции и решают проблему преждевременной сходимости.

При решении задач оптимизации функций в непрерывных областях поиска, одной из проблем является обеспечение высокой точности получаемых решений. Были предложены генетические операторы для хромосом с вещественным кодированием. Одним из таких операторов является оператор неоднородной мутации [4], который уменьшает интенсивность мутации решений по мере продвижения к оптимуму. Также известны меметические алгоритмы с вещественным кодированием (*real-coded memetic algorithms – RCMA*). Они используют методы градиента для нахождения локальных оптимумов. В генетических алгоритмах методы локального поиска применяются к популяции решений с целью исследования лучших областей пространства поиска, полученных в ходе выполнения генетического алгоритма. Такое сочетание методов генетического и локального поиска позволяет значительно повысить эффективность решения сложных задач оптимизации и проектирования.

Можно также отметить еще одну разновидность RCMA, так называемые алгоритмы локального поиска на основе кроссинговера (*crossover-based local search algorithms – XLS*). По мере смены поколений, популяция теряет разнообразие. Так как оператор кроссинговера производит решения-потомки группирующиеся вокруг родительских решений, это свойство используют в качестве средства выхода из локального оптимума. Это стимулирует эффективную локальную настройку алгоритма, что представляет особый интерес при вещественном кодировании [8].

В последнее время предложено много различных модификаций операторов кроссинговера, использующих три, четыре и более родительские хромосомы. При этом число потомков также неограниченно.

Предлагается оператор кроссинговера, который является модификацией родительски-центрированного оператора кроссинговера (BLX или PBX), предложенного Шелманом и Сшифером. Идея заключается в следующем. Пусть $X = (x_1, \dots, x_n)$ и $Y = (y_1, \dots, y_n)$ ($x_i, y_i \in [a_i, b_i] \subset \mathbb{R}$, $i = 1 \dots n$) являются двумя хромосомами с вещественным кодированием, выбранные на основе некоей стратегии для выполнения оператора кроссинговера. В результате получаем потомков: $Z1 = (z_1^1, \dots, z_n^1)$ или $Z2 = (z_1^2, \dots, z_n^2)$, где z_i^1 – произвольным образом выбранное число на интервале $[l_i^1, u_i^1]$, причем

$$l_i^1 = \max\{a_i, x_i - I \cdot \alpha\} \text{ и } u_i^1 = \min\{b_i, x_i + I \cdot \alpha\},$$

а z_i^2 выбирается на интервале $[l_i^2, u_i^2]$ и

$$l_i^2 = \max\{a_i, y_i - I \cdot \alpha\} \text{ и } u_i^2 = \min\{b_i, y_i + I \cdot \alpha\},$$

где $I = [x_i - y_i]$.

Основными достоинствами такого оператора является возможность изменять степень разнообразия популяции решений. Также важно, что оператор позволяет количественно оценить разнообразие популяции и степень «близости» вновь полу-

ченных решений к родительским. Таким образом, использование данного оператора в RCGAs дает возможность динамически, в процессе поиска оценивать возможности адаптации решений-потомков.

Оператор кроссинговера на основе глобального поиска многократно применяется к выбранной паре родительских решений до тех пор, пока не будет получено заданное число n_{off} потомков. После этого лучший из потомков заменяет худшего родителя (если это не приводит к ухудшению целевой функции). Процесс повторяется в течении n_{it} итераций, после чего в популяцию возвращают двух текущих исключенных родителей. Очевидно, что при выполнении такого оператора необходимо заранее задать значения n_{off} и n_{it} , и выбрать начальную пару родителей, (p_1, p_2) .

Предложенный оператор может быть использован при выполнении RCGA, поскольку для его реализации достаточно популяции минимального размера, то есть не менее двух хромосом. При этом за счет изменения правил отбора можно модифицировать данный оператор и преобразовать его в мультихромосомный. По результатам сравнения можно изменять текущую пару родителей в процессе выполнения оператора. При использовании решений с вещественным кодированием оценка качества решений производится на основе вероятности распределения популяции решений. Это позволяет регулировать процессы схождения и разнообразия решений в популяции.

В литературе описывается большое количество различных операторов кроссинговера для генетических алгоритмов с вещественным кодированием [9].

1) Унимодальный оператор кроссинговера с нормальным распределением (*unimodal normal distribution crossover* – *UNDX*) на основе множества родителей создает решения-потомки, распределенные вокруг центра массы этих родителей. При этом, чем дальше решение находится от центра массы, тем меньшая вероятность ему назначается.

2) Симплексный кроссинговер (SPX) назначает однородное распределение вероятности для того, чтобы создать потомство в ограниченном пространстве поиска вокруг области занятой родителями.

Они также называются операторами усредненного центра и используются при построении специализированной модели генетического поиска.

3) Родительски центрированный оператор рекомбинации, который присваивает тем большую вероятность создаваемым решениям-потомкам, чем ближе они находятся к родителям в пространстве поиска.

4) Нечеткие операторы кроссинговера и мутации на основе логических операций. Новые решения образуются из уже имеющихся за счет использования ло-

гических операций И, ИЛИ, НЕ, И-НЕ, ИЛИ-НЕ. Например оператор кроссинговера на основе операции ИЛИ-НЕ [4]. Хромосома при этом представляет собой числовую последовательность, каждое число которой представляет собой значение некоторого весового коэффициента, который может соответствовать, например, вероятности включения данного разряда последовательности в получаемое решение. Длина хромосомы определяется используемым способом кодирования.

Применяется также ряд гибридных методик и технологий организации процесса поиска, сочетающих достоинства локальных методов поиска и методов генетического поиска. К ним относятся модели минимального разрыва поколений (Minimal generation gap -MGG) и обобщения поколений (G3-Generalized generation gap). В таких моделях из нескольких родительских решений создаются 200 решений-потомков, из которых отбираются два лучших решения.

Очевидно, что создание эффективных методов решения задач оптимизации и управления на основе гибридных нечетких генетических методов находится только в самом начале пути. Однако уже имеется достаточно большое число примеров успешного использования таких методов для решения различных прикладных задач оптимизации.

ЛИТЕРАТУРА

- [1] Перегудов Ф.И., Тарасенко Ф.П. Введение в системный анализ. – М.: Высшая школа, 1989.
- [2] Прангишвили И.В. Системный подход и общесистемные закономерности. – М.: СИНТЕГ, 2000.
- [3] Борисов В.В., Круглов В.В., Федулов А.С. Нечеткие модели и сети. – М.: Горячая линия – Телеком, 2007.
- [4] Гладков Л.А., Гладкова Н.В. Особенности использования нечетких генетических алгоритмов для решения задач оптимизации и управления // Известия ЮФУ. Технические науки. Тематический выпуск "Интеллектуальные САПР". – 2009, №4. – С. 130-136.
- [5] Herrera F., Lozano M. Fuzzy Adaptive Genetic Algorithms: design, taxonomy, and future directions // Soft Computing 7(2003), Springer-Verlag, 2003. – P. 545 – 562.
- [6] Herrera F., Lozano M. Adaptation of genetic algorithm parameters based on fuzzy logic controllers // In: F. Herrera, J. L. Verdegay (eds.) Genetic Algorithms and Soft Computing, Physica-Verlag, Heidelberg, 1996. – P. 95-124.
- [7] Курейчик В.В., Курейчик В.М., Родзин С.И. Концепция эволюционных вычислений, инспирированных природными системами // Известия ЮФУ. Технические науки. Тематический выпуск "Интеллектуальные САПР". – 2009, №4. – С. 16-24.
- [8] Lozano M., Herrera F., Krasnogor N., Molina D. Real-Coded Memetic Algorithms with Crossover Hill-Climbing // Evolutionary Computation № 12(3), Massachusetts Institute of Technology, 2004. – P. 273-302.
- [9] Deb K., Joshi D., Anand A. Real-Coded Evolutionary Algorithms with Parent-Centric Recombination. Kanpur Genetic Algorithms Laboratory (KanGAL), Kanpur, PIN 208 016, India. KanGAL Report No. 2001003, 2001.