

**Санкт-Петербургский национальный исследовательский
университет информационных технологий, механики и оптики
Факультет информационных технологий и программирования
Кафедра «Компьютерные Технологии»**

**А.Д. Богуцкий
И.Б. Сметанников**

**Использование генетических алгоритмов для
построения функций принадлежности для
заданных нечетких множеств**

**Санкт-Петербург
2011**

Оглавление

Введение	3
1. Описание и постановка задачи	4
2. Модель котла с нечеткой системой управления	5
2.1 Общее описание котла.....	5
2.2 Система управления котлом	5
3. Реализация генетического алгоритма	6
3.1 Способ задания индивидуума.....	6
3.2 Тип генетического алгоритма.....	6
3.3 Функция приспособленности	6
3.4 Метод скрещивания двух особей	7
3.5 Метод мутации особи	7
4. Полученные результаты	8
5. Список литературы	10

Введение

Целью выполнения данной работы является реализация системы для генетического выращивания функций принадлежности для заданной системы нечеткого управления. В качестве модельной системы нечеткого управления рассматривается система управления газовым котлом [1]. Выращивание функций принадлежности позволяет снизить время на разработку системы нечеткого управления, повысить ее эффективность, а также позволяет не привлекать для подбора данной функции специалиста.

Для реализации системы была написана библиотека, позволяющая разрабатывать систему нечеткого управления с меньшими временными затратами и высокой эффективностью. Библиотека написана на языке Java.

1. Описание и постановка задачи

Для понимания принципа работы системы нечеткого управления необходимо дать небольшое описание математического аппарата нечеткой логики. Приведем основные понятия нечеткой логики:

1. Функция принадлежности $MF_C(x)$ — функция, характеризующая степень принадлежности некой величины к нечеткому множеству. Область допустимых значений функции равна $[0; 1]$.
2. Нечеткое множество A над множеством X — множество упорядоченных пар значение $x \in X$ и его принадлежность множеству. $A \stackrel{\text{def}}{=} \{(x, MF_C(x))\}$
3. Нечеткий логический вывод — система правил следующего вида:

$$\begin{cases} R_1: \text{if } (x_1 \in A_{11}) \text{ and } (x_n \in A_{1n}) \text{ then do smth} \\ R_2: \text{if } (x_2 \in A_{21}) \text{ and } (x_n \in A_{2n}) \text{ then do smth} \\ \vdots \\ R_m: \text{if } (x_m \in A_{m1}) \text{ and } (x_n \in A_{mn}) \text{ then do smth} \end{cases}$$

В данной работе требовалось осуществить генетический вывод функций принадлежности. В связи с тем, что функция принадлежности имеет значение в пределах $[0; 1]$, было решено в качестве функции принадлежности к определенному множеству взять функцию $\frac{1}{1+|f(x)|}$ выращивая, в свою очередь, функцию $f(x)$.

Чтобы проверить результаты работы генетического алгоритма, было решено реализовать математическую модель управления газовым котлом [1] и сравнить результат работы модели на приведенных в статье функциях с результатами работы модели на генетически выращенных функциях принадлежности.

Общая схема программы, решающей поставленную задачу, имеет вид, представленный на рис. 1.

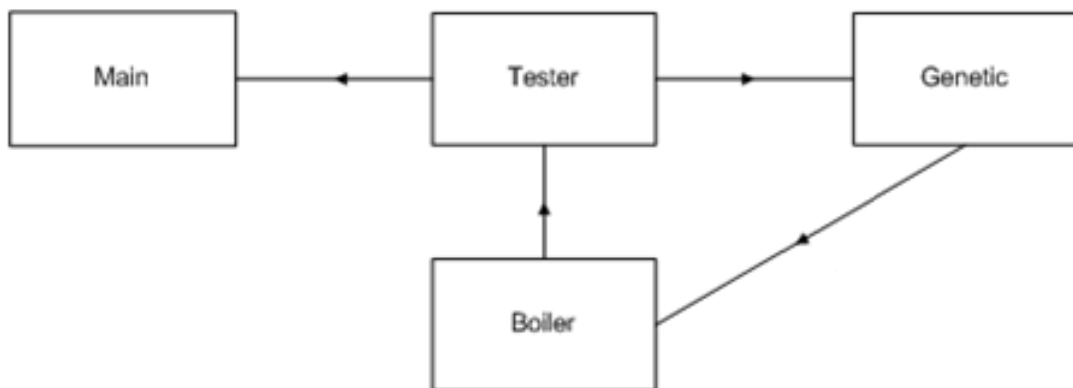


Рис. 1 – Простейшая структура программы

2. Модель котла с нечеткой системой управления

2.1 Общее описание котла

Основной задачей котла является поддержание температуры воды в заданном температурном интервале. С учетом множества дестабилизирующих факторов температура воды на выходе будет колебаться, в связи с этим, имеется необходимость регулировать работу котла, чем и занимается система нечеткого управления. Структура модели котла приведена на рис. 2.

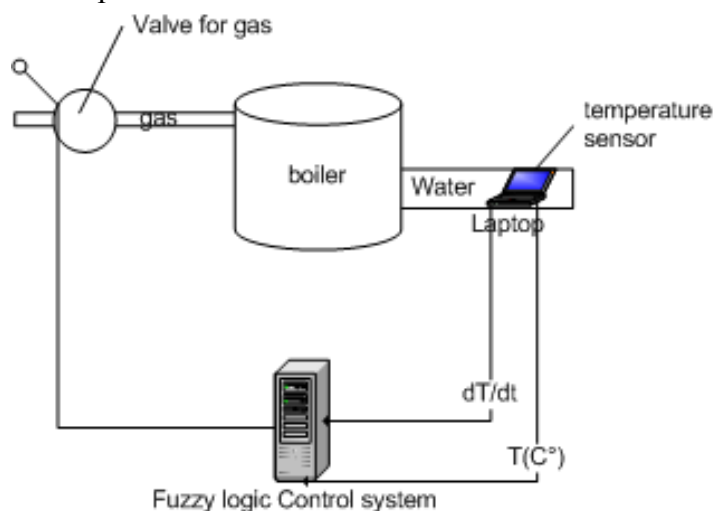


Рис. 2 – Структура, реализуемая в модели

2.2 Система управления котлом

Был смоделирован котел, представленный чугунной цилиндрической цистерной с водой, содержащей 200 кг воды. При вычислении температуры на каждом шаге работы тестировщика системы управления вычисляется энергия, полученная от сгорания природного газа, потраченная на теплообмен воды с окружающей средой и потраченная на теплообмен котла с окружающей средой. Остальными параметрами авторы решили пренебречь. В зависимости от показателей выбирается нечеткое множество, которому эти данные наиболее подходят, и выполняются действия, соответствующие данному множеству. Также каждое действие имеет некоторую стоимость, а выход температуры за заданные рамки облагается дополнительными штрафами. Впоследствии итоговая стоимость используется для вычисления функции приспособленности.

3. Реализация генетического алгоритма

В данном разделе описана реализация генетического алгоритма. Вначале описывается способ задания индивидуума, а также тип генетического алгоритма. Затем описывается реализация кроссовера и мутации.

3.1 Способ задания индивидуума

Множество простейших функций $\{f_i(x)\}$ будем называть базисным множеством, а функцию, взятую из данного множества – базисной. В данной работе особи задаются в виде набора функций принадлежности для каждого из заданных множеств $C_1 \dots C_m$. Для каждого множества функция принадлежности задается средним арифметическим некоторого набора базисных функций, записанных в специальном виде. Функция принадлежности в общем виде представлена следующей формулой:

$$MF_{C_j}(x) \stackrel{\text{def}}{=} \frac{1}{n_j} \sum_{k=1}^{n_j} \frac{1}{(1 + |f_{ik}(x)|)} \quad (1)$$

3.2 Тип генетического алгоритма

Для генерации нового поколения особей используется клеточный генетический алгоритм [2]. Все особи расположены в таблице с заданным размером. Противоположные стороны таблицы замыкаются между собой. В каждой клетке таблицы расположена только одна особь.

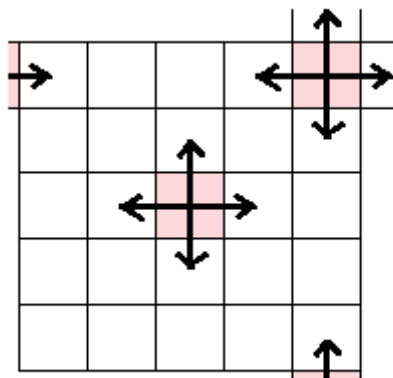


Рис. 3 – Схема клеточного генетического алгоритма

Новое поколение получается путем скрещивания каждого индивидуума с его четырьмя соседями (расположенными сверху, снизу, слева, справа). Из полученных в результате скрещивания потомков выбирается лучший и помещается в ячейку вместо родителя.

3.3 Функция приспособленности

Функция приспособленности заданной системы управления котлом складывается из расходов котла на газ и на работу с сервоприводом в определенный интервал времени. Кроме того, введен прогрессивный штраф за выход температуры воды в котле за заданный интервал. Система управления с наименьшей функцией приспособленности считается лучшей.

3.4 Метод скрещивания двух особей

Скрещивание двух особей происходит следующим образом: из обеих систем управления выбираются функции принадлежности для одинаковых множеств. Каждая функция принадлежности представлена некоторой усредненной суммой из формулы (1). Для двух новых особей формируются функции принадлежности путем создания двух новых усредненных сумм, слагаемые которых взяты из двух старых сумм случайным образом, как это показано на рис. 4.

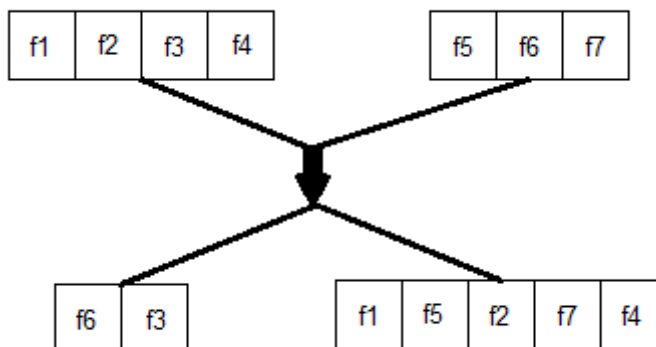


Рис. 4 – Скрещивание двух функций принадлежности

3.5 Метод мутации особи

При мутации особи изменяются ее функции принадлежности. При этом в каждой функции принадлежности выбираются некоторые случайные слагаемые и меняются на новые, случайно сгенерированные слагаемые, представленные в формуле (1).

4. Полученные результаты

Была реализована система для генетического выращивания функций принадлежности, также был проведен ряд тестов для выращенной системы нечеткого управления. Было произведено сравнение результатов работы выращенной системы нечеткого управления и стандартной системы, приведенной в статье [1].

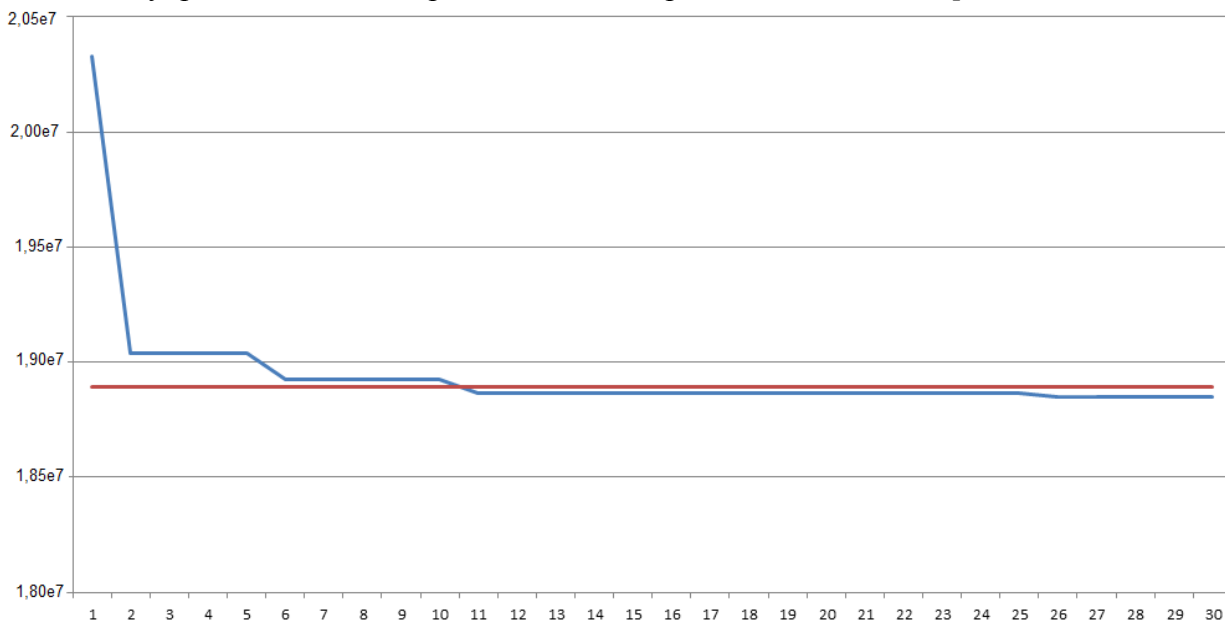


Рис. 5 – Сравнение выращенной системы принятия решений с системой принятия решений из статьи [1]. Функция приспособленности лучшего индивидуума из текущего поколения обозначается красной линией, а функция приспособленности системы из статьи [1] обозначается синей линией

Авторы статьи [1] получили достаточно неплохие результаты, используя при этом достаточно простые функции. Построенная в ходе выполнения работы система выращивания функций принадлежности дает маленький, но достаточно стабильный выигрыш относительно результатов, полученных при использовании системы управления из статьи [1]. Кроме того, эта система является достаточно универсальной.

Было проведено исследование генетического алгоритма на сходимость. Алгоритм был запущен 50 раз, при этом собирались данные о лучших особях на каждом шаге эволюции. Строились графики (рис. 6) функции приспособленности лучшей особи в лучшем, худшем и среднем случаях в зависимости от номера поколения.

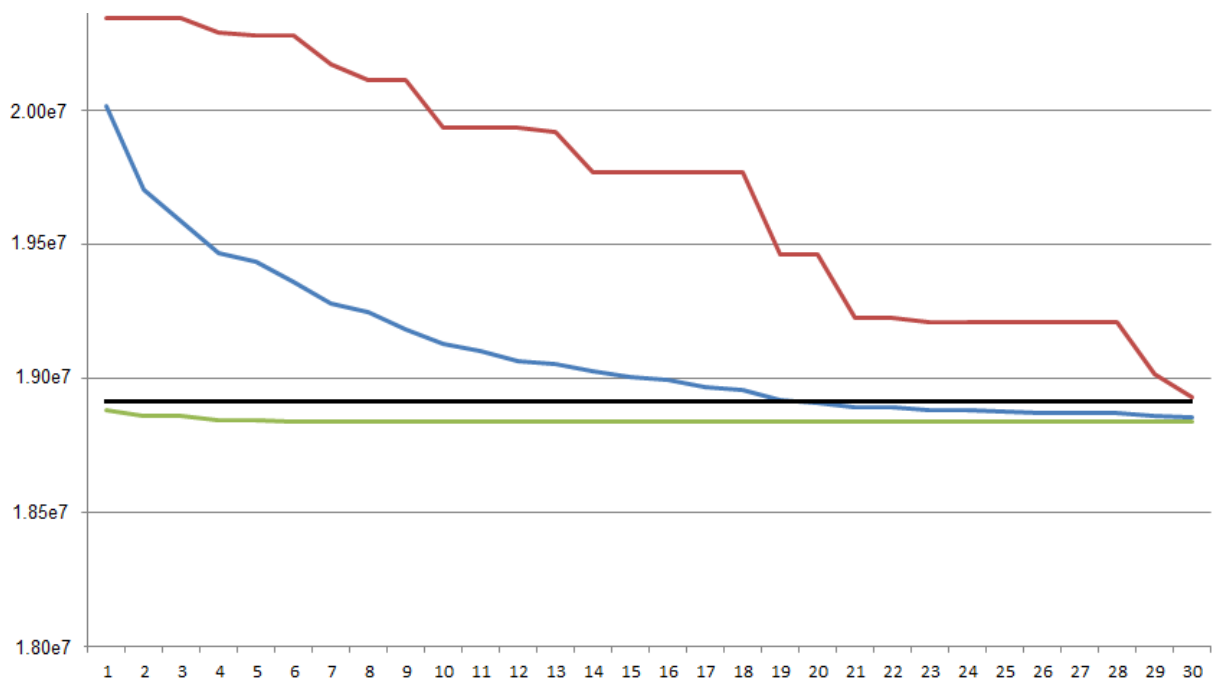


Рис. 6 – Функции приспособленности лучших особей на каждом шаге эволюции. Красной линией обозначен худший случай, зеленой линией – лучший, а синей линией – средний. Черной линией обозначен результат работы при использовании системы управления из статьи [1]

В результате тестирования было установлено, что, как правило, тридцати поколений достаточно для выращивания системы управления, при которой достигаются результаты работы не хуже, чем при использовании системы управления из статьи [1].

Котел выдает наилучшие результаты для системы нечеткого управления при использовании приведенных ниже функций принадлежности:

Пример 1:

- sqrt cos sh sin sin cos cos sin sin cos
- poly2
- poly2 cos
- sin sin ch
- sin

Пример 2.

- cos poly2
- cos
- poly2 poly3 line sin
- sin line sh cos sh poly4 cos line cos sin sh line line
- sin sin poly3 poly2 sin sin sin line cos poly3 sin

Данные системы принятия решений достигают примерно одного и того же значения функции принадлежности на заданном тесте. Как можно заметить в первом тесте, излишняя простота функций принадлежности со второй по пятую привела к значительному усложнению первой функции. Что же касается второго примера, то тут можно наблюдать усложнение в четвертой и пятой функции принадлежности, что привело к значительному упрощению остальных.

5. Список литературы

1. Белоусов О.А., Иванов С.В. Система нечеткого управления газовым котлом // Промышленные АСУ и контроллеры. – 2011. – № 1. – стр. 20– 24.
2. R.Poli, W. Langdon, N. McPhee. A field guide to genetic programming. Lulu Enterprises. UK Ltd. – 2008. – p. 252.
3. Genetic algorithms tutorial in 3 parts. <http://www.leolol.com/drupal/tutorials/theory>
4. Y. Yuan, H. Zhuang. A genetic algorithm for generating fuzzy classification rules // Fuzzy Sets