

Проектирование алгоритмов управления на основе аппарата нечеткой логики

Л. П. Козлова

Санкт-Петербургский государственный
электротехнический университет
«ЛЭТИ» им. В.И. Ульянова (Ленина)
tigrenok59@mail.ru

О. А. Козлова

Санкт-Петербургский государственный университет
телекоммуникаций им. проф. М.А. Бонч-Бруевича
K_olga_a@mail.ru

Аннотация. Современный технологический процесс предполагает высокую степень интеллектуализации информационных управляющих систем. Данная задача сопряжена с рядом сложностей, которые требуют решения. В статье пойдет речь о нечеткой логике, которая является одним из возможных методов, помогающих сделать систему управление более гибкой.

Ключевые слова: нечеткая логика; управление; технологический процесс; интеллектуальные системы

I. ВВЕДЕНИЕ

В настоящее время технологический процесс предполагает высокую степень автономности систем, не требующих человеческого вмешательства. Действительно, современные системы показывают результаты, намного превосходящие человеческие возможности.

Сложность заключается в том, что классические методы при увеличении интеллектуализации серьезно повышают сложность, что, в свою очередь, делают системы более дорогими, и не всегда допускают реальную эксплуатацию.

Для того чтобы интегрировать интеллектуальные системы в реальный процесс, необходимо использовать методы, отличающиеся от базовых разработок таким образом, чтобы упростить структуру и сделать ее доступной для внедрения.

Отдельно можно выделить разработку систем интеллектуального управления, которые отвечают за отслеживание и регулирования процесса в целом.

II. СТРУКТУРА УПРАВЛЯЮЩЕЙ СИСТЕМЫ

Общую структуру технологического процесса можно разделить на следующие составляющие:

- Управляющие устройства, которые, в общем смысле, могут включать программно-аппаратный комплекс, а в некоторых случаях и человека.
- Объект управления, который выполняет действия, необходимые для решения задачи работы системы в целом.
- Устройство обратной связи, которое анализирует выходное значение для возможности коррекции работы системы.

Алгоритм такой системы прост:

- На первом шаге входная информация поступает на вход алгоритмов управляющего устройства, которое, на основе вычислительного базиса формирует решение для управления объектом.
- Управляющее воздействие приводит в движение объект управления, который начинает формировать выходное значение.
- Выходное значение поступает на блок системы обратной связи, которое анализирует его на соответствие заложенным параметрам качества. Если оно попадает в заложенные границы, то система продолжает работать без изменения. Если же значение отклоняется, то на управляющее устройство поступает корректирующий сигнал.

Общая структура приведена на рис. 1.



Рис. 1. Структура автоматизированной системы управления

Данный алгоритм может продолжаться до тех пор, пока требуется в технологическом процессе, либо, пока его искусственно не остановят.

Трудность возникает в том, что алгоритмы систем управления построены на строгом расчете математических значений.

С одной стороны, это – достоинство. Действительно, что может быть точнее алгебраических, дифференциальных уравнений и т. п. Однако проблема заключается в том, что системы, предполагающие высокий уровень автоматизации зачастую не имеют числовых значений, напротив, основными терминами могут служить лингвистические понятия. В этом случае, произвести классические математические расчеты не представляется возможным.

Другой проблемой может стать отсутствие четко структурированной входной информации, более того, она может носить высокий уровень априорной

неопределенности, что требует наличия нескольких алгоритмов, хранящихся на управляющих устройствах, а также использование принципов выбора одной из схем управления в каждом конкретном случае. Это, в свою очередь, приводит к увеличению необходимых системных ресурсов и усложнению структуры.

И, наконец, к проблемам классических систем отнесем их постоянный характер функционирования, тогда как структура интеллектуальных систем, зачастую требует переменных или случайных значений.

Как уже говорилось выше, для решения всех этих проблем, требуются альтернативные методы. Именно к таким отнесем принцип нечеткой логики.

III. ПОНЯТИЕ НЕЧЕТКИХ СИСТЕМ

В основу концепции нечеткой логики ложится необходимость преодоления неоднозначности.

Сам принцип может использоваться в разных частях работы технологического процесса, поэтому первично необходимо выяснить, что он привносит в стандартный алгоритм работы.

Классические системы определяют переменные одним единственным образом. Это естественно в тех случаях, когда характеристики, по которым определяются системы, имеют однозначный характер. Другими словами, можно сказать, что они определяют выбранный показатель на 100 %.

Если в идеальных системах такая ситуация естественна, то в реальном мире любой объект может обладать набором показателей, характеризующихся по одному основанию (например, цвет). При необходимости выбора классические системы всегда используют один принцип: наибольший показатель определяет отношение к объекту.

Теперь стоит обратить внимание, что технологический процесс использует множество переменных, каждая из которых имеет набор своих показателей. Далее, появляется необходимость сочетания переменных для реализации основной задачи.

Если переменные подкрепляют друг друга, то проблем у классических систем не возникает. Но есть другая ситуация: когда элементы системы ослабляют друг друга, а, соответственно, и всю систему в целом.

Естественно, в данной ситуации имеется алгоритм, позволяющий использовать значения таким образом, чтобы все показатели проходили по установленному стандарту качества, хотя бы по минимальному его уровню. Вот только в классических системах они все равно берутся по максимально возможному для не вывода из строя остальных показателей, значению. Тогда как в реальных системах, зачастую, можно использовать минимальное (ну или не максимальное) допустимое, чтобы более актуальный, для данной проблемы показатель, мог использовать более высокие значения.

Для преодоления данной сложности нечеткая логика вводит понятия степени принадлежности.

Предполагается, что каждая характеристика объекта использует диапазон $[0; 1]$ для своего определения. Обязательное условие, чтобы все степени принадлежности для одного показателя в сумме давали единицу.

Формализация данного условия выглядит следующим образом:

$$C = \{MF(\omega) / \omega\}, \quad (1)$$

здесь ω – выбранная характеристика; $MF(\omega)$ – степень принадлежности, причем $MF(\omega) \in [0, 1]$.

А. Понятие сегментации

Важным моментом в задаче перехода от классических систем в формат нечеткой логики является алгоритм сегментации. Именно он определяет дальнейшие результаты работы с системой.

Принцип данного процесса заключается в том, что для исходного объекта M определяется набор признаков $f(x_1, x_2, \dots, x_n)$, где n – общее значение элементов.

Следующий процесс производит разбиение множества M на подмножества ω_j .

Работа сегментации требует выполнения следующих свойств:

- При выделении сегмента обозначается только один примитив. В формате формализации данное условие будет выглядеть следующим образом:

$$\omega_i \cap \omega_j = 0, \forall i \neq j \quad (2)$$

- Показатели внутри одного сегмента предполагают исключительно однородными, т.е. предикат $P(\omega_i) = true, \forall i$.
- Сегменты не могут полностью повторять друг друга, т.е. всегда у сегментов должно быть хотя бы одно отличие.
- Граница, разделяющая сегменты должна обладать свойством очевидности и конкретизировать пространство.

При классификации алгоритмов сегментации, в общем смысле, самым простым является разделение на ручное разделение, когда человек определяет границы сегмента, и механическое, когда задача полностью возложена на технические системы [1].

Ручная сегментация полностью зависит от качества работы специалиста, который ее выполняет. Автоматическую можно разделить на однородную сегментацию и сегментацию с заданными свойствами.

Наиболее универсальным считается однородная сегментация, поскольку в ней отсутствует необходимость задавать алгоритмы, выделяющие границы примитивов.

В случае использовании сегментации с заданными параметрами, необходимо выделить те свойства примитивов, которые являются основными для данной выборки, чтобы на основе их организовать границы

сегментов. Остальные свойства в этом случае используются только второстепенным образом.

Результат работы сегментации, как правило, оценивается в совокупности всех этапов работы алгоритма. Однако, существуют случаи, когда необходимо обособить критерии качества непосредственно сегментации. В этом случае, происходит выделение основных признаков, характеризующих систему. Эти показатели задают критерии качества сегментации, в результате чего, несложно проверить ее итоговый результат.

В. Создание структуры нечеткой системы

Формат работы нечеткой системы проще всего представить в виде следующей последовательности:

- Фазификатор. Данная процедура предполагает получение вектора входных значений и преобразование его в формат вектора нечетких множеств.
- Формирование правил работы системы в формате «если ... то...».
- Использование логических операций соответствующих математическим значениям нечеткой логики в случае, если созданные правила используются в совокупности.
- Функционирование аппарата нечеткого вывода использующего понятие обобщения и заключения.
- Процесс дефазификации, предполагающий выбор конкретных значений, которые будут использоваться в той задаче, которая выполняется в конкретный момент времени.

Таким образом, из многообразия получившихся результатов, можно выбрать тот, который будет оптимально подходить под сложившуюся совокупность параметров, что позволит более гибко управлять системой в целом [2].

IV. ИСПОЛЬЗОВАНИЕ НЕЧЕТКОЙ ЛОГИКИ В ПРОЦЕССАХ РЕГУЛИРОВАНИЯ

Исходя из принципа нечеткой логики, можно внедрить ее в регулирование технологическим процессом.

Общая схема приведена на рис. 2.

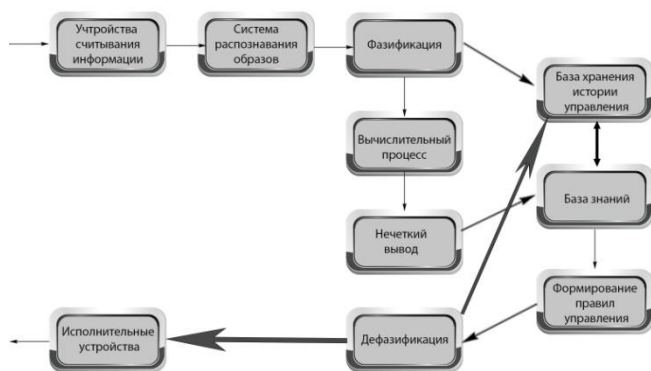


Рис. 2. Нечеткий аппарат в системе управления

В данной схеме предполагается следующее разделение:

- Блок входной/выходной информации, которая взаимодействует с внешней средой.
- Блок, отвечающей за обработку информации и принятия решений в ходе технологического процесса.
- Блок, отвечающий за хранение информации и выработку знаний, которые ложатся в основу принятия решений.

Устройства считывания информации могут состоять из датчиков, сканирующих панелей, устройств ввода операторов. Их основная задача получить нужную информацию из окружающего пространства.

Основная проблема в том, что каждое устройство, которое может входить в этот класс предполагает множество сложностей:

- Ошибки операторов.
- Световые эффекты, когда вопрос касается считывания изображений. В данной ситуации не имеет значения: реальные эти изображения или отсканированные.
- Проблемы оборудования.
- Помехи, мешающие работе аппаратуры и т. .

Все приведенное вносит множество сложностей в работу алгоритмов, именно поэтому нечеткий аппарат, который, обладает большей гибкостью, дает более качественные результаты.

Этап распознавания образов предполагает формирование примитивов для дальнейшей обработки. Именно он может определить ошибочные данные, которые, в последствие дадут погрешность, при дальнейшей работе алгоритмов.

Фактически, существует множество алгоритмов распознавания образов. В общем смысле можно поделить их на следующие этапы:

- Получение исходных данных с устройств считывания информации.
- На основе алгоритмов распознавания образов, разделение общего массива информации на отдельные составляющие – примитивы.
- Образовавшиеся примитивы сравниваются со значениями в базе эталонов (отсутствует на рис. 2).
- Исходя из понятий эталонной базы, присвоение значений примитивам, которые лягут в основу принятия решений.
- Оценка качества работы алгоритма.

Многообразие алгоритмов распознавания образов интересно еще и потому, что многие из них используют алгоритмы нечеткой логики, но данный аспект рассмотрен не в этой статье.

После этапа распознавания образов идет этап работы нечеткого аппарата. Будем делить его на следующие составляющие:

- Этап фазификации, который преобразует вектор входных данных в формат терминов предусмотренных нечеткой логикой.
- На основе образовавшегося вектора данных составление правил в формате «Если..., то...».
- Использование булевых операций отвечающих формату нечетких правил в случае, когда за один аспект отвечает несколько правил.
- На основе обобщения и заключения работа аппарата нечеткого логического вывода.

На следующей стадии, система обращается к базе знаний, которая хранит в себе информацию о возможных действиях.

По сути, данный блок содержит в себе три элемента:

- Непосредственно сама база знаний.
- Блок, который на основе результатов нечеткого логического вывода сопоставленного с данными, хранящимися в базе знаний, формирует решение по дальнейшему функционированию системы.
- База хранения информации, в которую попадают данные с различных процессов, протекающих в системе. Одна из важных задач данного сегмента – формировать значения, позволяющие расширить или изменить базу знаний.

Из всего перечисленного формируются правила, которые позволяют произвести дефазификацию, т.е. выбор конкретных значений для конкретных параметров. Именно данные управляющие воздействия поступают на вход исполнительных устройств, которые выполняют общую поставленную задачу [3].

V. ФОРМИРОВАНИЕ БАЗЫ ЗНАНИЙ

Обособленной задачей всегда стоит формирование базы знаний. У детерминированных систем создание такой структуры вызывает сложность только на начальном этапе, поскольку база формируется один раз и не изменяется на протяжении всей жизни системы.

У интеллектуальных систем появляется необходимость изменять заложенные на начальной стадии работы эталоны.

Чтобы сформировать алгоритм, который позволяет полноценно воплощать данную задачу необходимо, чтобы исходная система обладала следующими возможностями:

- Алгоритмы изменения количества объектов в базе.
- Наличие алгоритмов, позволяющих изменять исходные объекты в базе.

- Алгоритмы способные анализировать, отличия исходных объектов от возможных изменений или новых объектов.

При внесении нового образца в базу знаний зачастую необходимо использовать принцип доверия.

Данный принцип относится к вероятностным показателям, что означает, что в каждой ситуации необходимо заново оценивать его надежность. Более того, значения, оказывающиеся верными для одной базы знаний, могут оказаться ошибочными для другой.

В случае, если принцип доверия эталону является транзитивным, необходимо принимать решение о доверии эталону из совокупности следующих факторов:

- В случае если эталон у заимствованной базы тоже заимствован, используется вероятностная характеристика этого процесса.
- Вероятностные характеристики определяются на начальной стадии, до заимствования эталона.

Вопрос достоверности используется во многих сферах технической активности, и созданные в их рамках алгоритмы возможно применять и в данной задаче.

VI. ЗАКЛЮЧЕНИЕ

Построение систем управления сложная неоднозначная и плохо формализованная задача. От ее решения зависит насколько адекватным будет технологический процесс для использования.

Внедрение современных методов, увеличивают процент точности работы системы. Так нечеткая логика дает большую вариативность и гибкость блока управления.

Из проблем, связанных с ней, можно выделить усложнение математического аппарата, однако это корректируется возможностью использования нечетких алгоритмов на тех этапах, где они действительно необходимы.

СПИСОК ЛИТЕРАТУРЫ

- [1] Козлова Л.П., Козлова О.А. Роботы тоже могут видеть // Известия СПбГЭТУ «ЛЭТИ», Санкт-Петербург. 2009, № 10, с. 47-50.
- [2] Kozlova Lyudmila P., Belov Aleksandr M., Kozlova Olga A. The Use of Neural Networks for Planning the Behavior of Complex Systems. Proceedings of the 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIconRus). Saint Petersburg, Russia, pp. 902-904. DOI: 10.1109/EIconRus.2018.8317234.
- [3] Kozlova L.P., Kozlova O.A. Fuzzy Logic Application in Solar Panels. Proceedings of the International Scientific and Practical Conference "Young Engineers of the Fuel and Energy. Complex: Developing the Energy Agenda of the Future" (EAF 2021). Published by Atlantis Press International B.V., Volume 213, 2022, pp. 45-50. <https://doi.org/10.2991/aer.k.220308.008>.