



СИСТЕМА АВТОМАТИЗИРОВАННОЙ РАЗРАБОТКИ МНОГОМЕРНЫХ ЛОГИЧЕСКИХ РЕГУЛЯТОРОВ С ПЕРЕМЕННЫМИ В ВИДЕ СОВОКУПНОСТИ АРГУМЕНТОВ ДВУЗНАЧНОЙ ЛОГИКИ

А.Ф. Антипин (ГОУ ВПО УГНТУ)

Рассматривается принцип работы и инструментарий системы автоматизированной разработки многомерных логических регуляторов с переменными в виде совокупности аргументов двузначной логики и минимизированным временем отклика, инвариантной по отношению к языкам программирования стандарта IEC 61131-3.

Ключевые слова: система автоматизированной разработки, многомерный логический регулятор, семантический анализ правил.

За последнее время в технологии автоматизации сложных объектов и процессов, имеющих важное народнохозяйственное значение, сложилась устойчивая тенденция к использованию одномерных логических (нечетких и с четкими термами) регуляторов. Как правило, упомянутые объекты управления удается описать только вербально (словесно), к тому же подавляющее большинство подобных объектов являются многомерными с взаимосвязанными регулируемыми параметрами. Принципиальный недостаток такого подхода состоит в автоматизации технологических систем с влияющими друг на друга регулируемыми параметрами с помощью сепаратных (автономных) регуляторов, выходы которых независимы по определению, то есть реагируют исключительно на "свой" вход.

Из сказанного выше вытекает, что при разработке многомерных регуляторов основной проблемой являются перекрестные связи. Задачей синтеза многомерной системы, в первую очередь, является компенсация взаимного влияния каналов регулирования за счет введения дополнительных компенсирующих связей.

Известно [1, 2], что современные многомерные нечеткие и дискретно-логические регуляторы из-за

большой погрешности и низкого быстродействия не позволяют с приемлемой точностью устранить взаимное влияние контуров регулирования. Особенно ярко это проявляется при управлении многомерными объектами, представленными в виде описания на естественном языке, то есть вербально.

Проблемную ситуацию призван разрешить многомерный логический регулятор (МЛР) с переменными в виде совокупности аргументов двузначной логики, минимизированным временем отклика и компенсацией взаимного влияния контуров регулирования, в котором блок логического вывода представлен в виде системы управляющих воздействий с механизмом формирования идентификационных номеров продукционных правил [3].

Однако современные системы и программные комплексы для программирования промышленных и ПК-основанных логических контроллеров не содержат в своем составе специализированного инструментария, предназначенного для разработки и/или анализа структуры МЛР [4], что вызвано, прежде всего, относительной новизной регуляторов данного типа.

Таким образом, необходимость создания системы автоматизированной разработки МЛР обусловлена:

а) высокой сложностью реализации системы продукционных правил МЛР, которая напрямую зависит от числа четких термов, используемых для внутреннего представления входных и выходных переменных;

б) отсутствием специализированных систем и сред разработки, предназначенных для программной реализации МЛР.

На рис. 1 представлен скриншот основного окна системы автоматизированной разработки МЛР (САР МЛР), разработанной в Стерлитамакском филиале ГОУ ВПО Уфимского государственного нефтяного технического университета. Система физически представляет собой один исполняемый файл, созданный в среде объектно-ориентированного языка программирования Delphi, с оригинальным названием CAD M-DLR 1.x.exe, где x – порядковый номер версии системы, который предназначен для работы в операционной среде Microsoft Windows XP и выше. Для запуска системы не требуется установки дополнительного программного обеспечения.

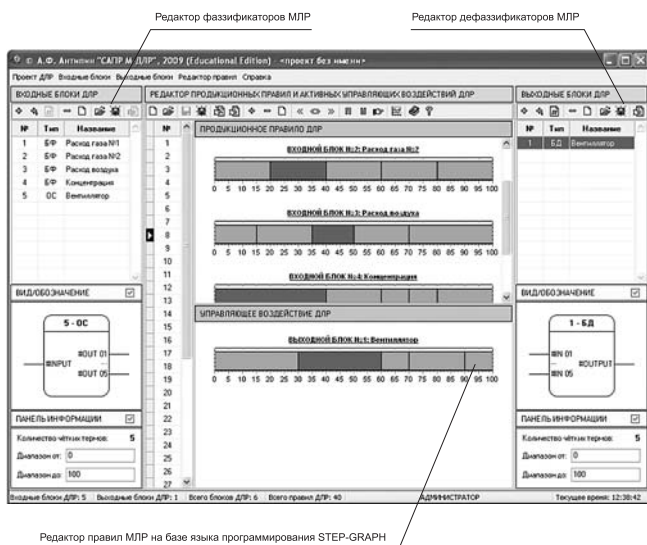


Рис. 1

Функционально САР МЛР состоит из:

1) редакторов фаззификаторов и дефаззификаторов МЛР, предназначенных для создания, редактирования и интегрирования блоков фаззификации и дефаззификации в МЛР, а также преобразования их внутреннего кода в формат языков программирования промышленных и ПК-основанных логических контроллеров, определенных стандартом IEC 61131-3. Параметры фаззификаторов и дефаззификаторов МЛР задаются в окне (рис. 2);

2) редактора производственных правил и управляющих воздействий МЛР. Для САР МЛР разработан специализированный графический язык программирования STEP-GRAPH, который базируется на системе фаззификаторов и дефаззификаторов МЛР. При создании правила в инструментальной среде STEP-GRAPH отображаются отрезки универсальной числовой оси всех функциональных блоков МЛР, разбитые на области (четкие термы), отмеченные красным цветом. Принцип работы со STEP GRAPH основан на создании системы правил, путем выделения необходимых областей, которые будут автоматически окрашены в зеленый цвет. Ключевой особенностью здесь является наглядное представление производственных правил и управляющих воздействий МЛР, что не требует от пользователей системы навыков в области программирования.

3) подпрограммы семантического анализа (ПСА) структуры программного кода МЛР, без которого невозможно произвести преобразование программы МЛР в формат языков программирования стандарта IEC 61131-3. На рис. 3 представлен скриншот окна ПСА САР МЛР.

На вкладке ПСА "Результат анализа термов" расположена таблица с результатом анализа четких термов МЛР, где поля T1...T9 предназначены для вывода информации о характере и полноте использования первых 9 термов непрерывных физических величин МЛР. В таблице используются следующие обозначения: "+" – показывает, что четкий терм задействован и используется в системе правил МЛР; "-" – показывает, что терм не используется в системе правил МЛР; "НЗ" – отражает незадействованные термы МЛР.

На вкладке ПСА "Результат анализа правил" расположена таблица с результатом анализа правил МЛР, где поля "№ повтор.правил", "№ против.правил", "№ доп.правил" предназначены для вывода идентификационных номеров правил, которые повторяют, противоречат или дополняют другие правила МЛР соответственно.

Принцип работы с "САР МЛР" основан на составленной автором методике разработки МЛР, которая включает следующую последовательность этапов:

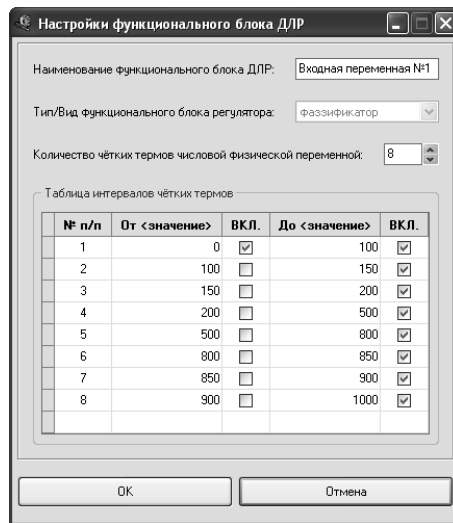


Рис. 2

1) определяется максимальное число входных/выходных переменных МЛР, их номинальные значения и числовой диапазон (только для физических величин).

2) задается требуемая точность МЛР, путем представления физических величин эквивалентной совокупностью четких термов. Логические переменные будут иметь не более 2 термов, обозначающие соответствующие состояния переменных – "0" и "1". Для непрерывных физических величин выбирается характер и порядок распределения четких термов (прямой или обратный) на универсальной числовой оси.

3) формируется требуемое количество блоков фаззификации (фаззификаторов) и дефаззификации (дефаззификаторов) МЛР, согласно данным, собранным на предыдущем этапе.

4) в инструментальной среде "САР МЛР" составляется система фаззификаторов, дефаззификаторов, входных и выходных параметров МЛР с учетом обратных связей, согласно блок-схеме МЛР с *n* входными параметрами *X*, *k* и *m* выходными параметрами *Z* и *Y* соответственно, представленной на рис. 4, где $\Phi_1^X, \Phi_2^X, \dots, \Phi_n^X$ – фаззификаторы МЛР; $D_1^Y, D_2^Y, \dots, D_m^Y$ и $D_1^Z, D_2^Z, \dots, D_k^Z$ – дефаззификаторы МЛР; $K(X_1), K(X_2), \dots, K(X_n)$ – количество четких термов входных параметров *X* МЛР; $K(Y_1), K(Y_2), \dots, K(Y_m)$ и $K(Z_1), K(Z_2), \dots, K(Z_k)$ – количество четких термов выходных параметров *Y* и *Z* МЛР соответственно; N_{max} – максимальный идентификационный (ID) номер производственного правила МЛР; $N_{тек}$ – ID номер производственного правила МЛР, antecedent которого равен логической единице в момент времени *t*.

5) производится настройка STEP-TIME алгоритма фаззификации, которая заключается в определении значений интервалов четких термов переменных МЛР, согласно способу интерпретации непрерывных физических величин в виде совокупности аргументов дву-

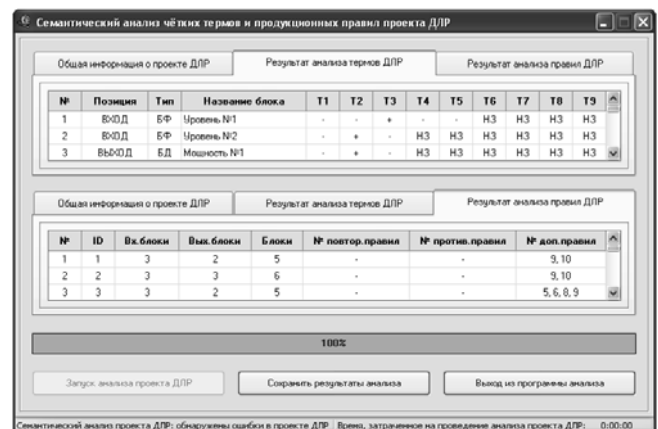


Рис. 3

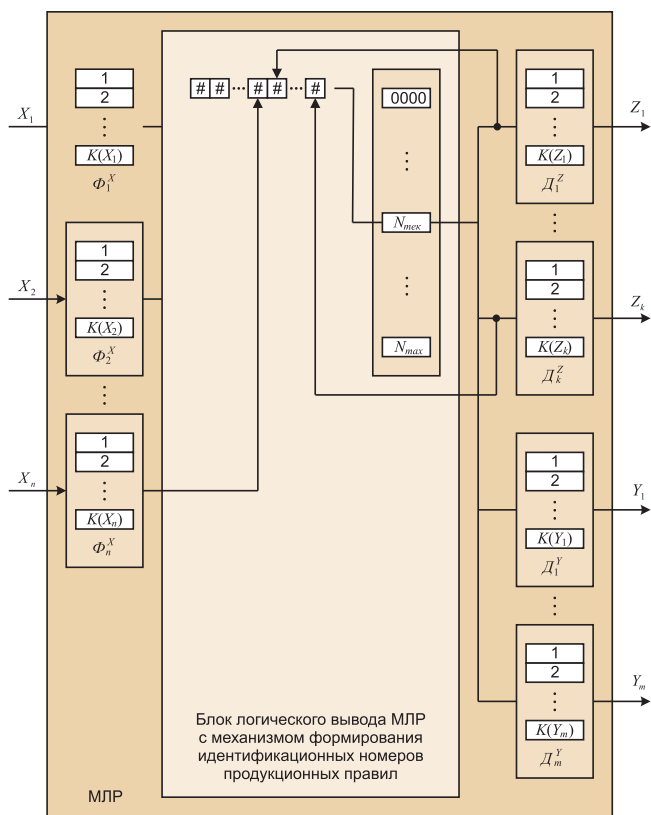


Рис. 4

значной логики, с выбором характера и порядка распределения термов на универсальной числовой оси.

б) производится выбор функции дефаззификации или непосредственно числовых значений для каждого четкого терма выходной непрерывной физической величины. Для переменных типа "BOOL" производится выбор логических состояний – "0" и "1" для каждой из возможных ситуаций.

7) для каждой переменной МЛР определяется система условий (продукционных правил), при которых те или иные четкие термы переменной будут равны логической единице и логическому нулю.

8) на основе информации, полученной на предыдущем этапе, формируется база данных продукционных правил и управляющих воздействий, которая представляет собой систему массивов (таблиц) значений выходных переменных МЛР для каждой из возможных ситуаций.

9) система массивов базы данных продукционных правил и управляющих воздействий МЛР в САР МЛР составляется следующим образом:

- индексы массивов обозначают соответствующие идентификационные номера продукционных правил МЛР.

- максимальное значение индексов массивов соответствует максимальному числу продукционных правил МЛР;

- элементы массивов содержат в себе идентификационные номера четких термов выходных переменных и обозначают собой управляющие воздействия МЛР.

10) "САР МЛР" автоматически производит расчет основных параметров МЛР, таких как:

а) максимальное количество продукционных правил R МЛР с n -входными параметрами X , k и m -выходными параметрами Z и Y соответственно

$$R = R_X R_Z + 1,$$

где R_X – максимальное количество продукционных правил, образованных n входными параметрами X ; R_Z – максимальное количество продукционных правил, образованных k выходными параметрами Z , четкие термы которых задействованы в условной части продукционных правил блока логического вывода МЛР (внутренняя обратная связь).

$$R_X = K(X_1)K(X_2).../K(X_n),$$

$$R_Z = K(Z_1)K(Z_2).../K(Z_m).$$

б) максимальное число операций сравнения L МЛР с n входными параметрами X , k и m выходными параметрами Z и Y соответственно зависит от суммарного числа четких термов непрерывных физических величин и рассчитывается согласно выражению:

$$L = \sum_{i=1}^n (K(X_i) + 1) + \sum_{j=1}^k (K(Z_j) + 1).$$

в) максимальный и текущий ID номер продукционного правила МЛР:

$$N_{max} = R - 1;$$

$$N_{max} = (K(P_1) - 1)K(P_2)...K(P_k) + ... +$$

$$+ (K(P_1) - 1)K(P_1) + K(P_1);$$

$$N_{max} = (K_1(P_1) - 1)K(P_2)...K(P_k) +$$

$$+ (K_1(P_2) - 1)K(P_3) ... K(P_k) + ... +$$

$$+ (K_1(P_1) - 1)K(P_1) + K_1(P_1),$$

где вектор P описывает совокупность n входных и k выходных переменных X и Z МЛР соответственно; $K_1(P_1)$, $K_1(P_2)$, ..., $K_1(P_k)$ – порядковые номера четких термов переменных $P_1, P_2, ..., P_k$ МЛР соответственно, значения которых в момент времени t равны логической единице; $K(P_1), K(P_2), ..., K(P_k)$ – количество четких термов переменных $P_1, P_2, ..., P_k$ МЛР соответственно.

г) максимальный размер базы данных продукционных правил S_{BD} МЛР с n входными, k и m выходными параметрами

$$S_{BD} = R(k + m).$$

11) экспериментально снимаются зависимости управляющих воздействий от задающих параметров в автономном (сепаратном) и многосвязном режимах работы МЛР.

12) на основе информации, полученной на предыдущем этапе, формируется компенсационная система продукционных правил (КСПП), предназначенная для исключения в установившемся режиме взаимного влияния контуров регулирования МЛР.

13) производится семантический анализ [5] структуры МЛР на предмет выявления "пустых", противоречащих и дополняющих друг друга правил, а также неиспользуемых четких термов, блоков фаззификации и дефаззификации.

Семантический анализ правил основан на концепции, состоящей из следующих основных положений:

а) если продукционное правило не содержит термины блоков МЛР, то это правило защиты (аварийное или нулевое правило). В МЛР может быть только одно аварийное правило, предназначенное для контроля над выходом значений физических величин за пределы указанного диапазона;

б) если правило или только управляющее воздействие не содержат четкие термины функциональных блоков МЛР, то это пустое правило;

в) если правило A соответствует правилу B , то это повторяющиеся правила;

г) если управляющее воздействие правила A идентично управляющему воздействию правила B , но при этом условные части правил не соответствуют друг другу, то это взаимодополняющие правила. Их можно и нужно объединять при возможности. Пример дополняющих друг друга правил:

$$\text{IF } (X_1 = 10) \text{ AND } (X_2 = 35) \text{ THEN } Y_1 = 10;$$

$$\text{IF } (X_2 = 48) \text{ AND } (X_3 = 24) \text{ THEN } Y_1 = 10,$$

где X , Y – входные и выходные переменные МЛР соответственно.

д) если условная часть правила A соответствует условной части правила B , но при этом значения их управляющих воздействий не равны друг другу и не содержат одинаковых порядковых номеров четких термов для каждого из используемых функциональных блоков МЛР, то это противоречащие правила. В противном случае это будут взаимодополняющие правила. Пример правил данного вида:

$$\text{IF } X_1 = 10 \text{ THEN } Y_1 = 10;$$

$$\text{IF } X_1 = 10 \text{ THEN } Y_1 = 20.$$

Таким образом, если значения "10" и "20" физической величины Y_1 лежат в числовых пределах одного четкого терма, то указанные правила дополняют друг друга и их можно объединить. В противном случае, данные правила будут противоречить друг другу. Данное заключение следует из того факта, что у любой физической величины в момент времени t существует один и только один четкий терм, значение которого равно логической единице;

е) если правило A не равно правилу B , то:

- если условные части правил не содержат одинаковых номеров четких термов для каждого из используемых функциональных блоков, то правила A и B – "нормальные", то есть удовлетворяют структуре МЛР;

- если правила A и B не "нормальные", но при этом значения их управляющих воздействий не равны друг другу и не содержат одинаковых порядковых номеров четких термов для каждого из функциональных блоков МЛР, то это противоречащие правила. В противном случае это будут взаимодополняющие правила.

Например,

$$\text{IF } X_1 = 10 \text{ THEN } Y_1 = 10;$$

$$\text{IF } X_1 = 30 \text{ THEN } Y_1 = 20.$$

Таким образом, если значения "10" и "30" непрерывной физической величины X_1 не лежат в числовых пределах одного четкого терма, то указанные правила являются "нормальными", то есть удовлетворяют структуре МЛР. В противном случае, если значения "10" и "20" физической величины Y_1 лежат в числовых пределах одного четкого терма, то указанные правила дополняют друг друга и их можно объединить. Иначе, это будут противоречащие правила.

Необходимо отметить, что результатом семантического анализа МЛР, помимо информационной составляющей, является запрет генерации программного кода МЛР в случае, если суммарное количество "пустых", повторяющихся и противоречащих друг другу правил больше или равно 1, а также при наличии неиспользуемых фаззификаторов и дефаззификаторов МЛР.

Результат анализа программы МЛР можно сохранить в файл формата НТМ (от англ. Hyper Text Markup – гипертекстовая разметка).

Результатом работы "САР МЛР" является законченный программный код МЛР в формате языков программирования промышленных и ПК-основанных контроллеров стандарта IEC 61131-3.

Таким образом, разработка многомерных логических регуляторов с переменными в виде совокупности аргументов двузначной логики и компенсацией взаимного влияния контуров регулирования в инструментальной среде "САР МЛР" значительно сокращает затраты времени в плане:

- создания системы фаззификаторов и дефаззификаторов МЛР с требуемым числом, характером и порядком распределения четких термов физических величин;
- проверки и, при необходимости, корректировки параметров фаззификаторов и дефаззификаторов с последующим их интегрированием в структуру МЛР;
- импорта/экспорта параметров функциональных блоков МЛР, преобразования программного кода блоков в форматы языков программирования, соответствующих международному стандарту IEC 61131-3;
- создания или редактирования системы продукционных правил и значений управляющих воздействий МЛР в графическом языке программирования STEP GRAPH, основанном на системе фаззификаторов и дефаззификаторов МЛР;
- формирования элементов массивов базы данных продукционных правил и управляющих воздействий МЛР;
- проведения семантического анализа структуры МЛР на предмет выявления логических ошибок или отклонений от концепции построения;
- составления единственно верной компенсирующей системы продукционных правил, предназначенной для исключения взаимного влияния контуров регулирования МЛР.
- генерации программы МЛР в формате языков программирования стандарта IEC 61131-3. Важно отметить, что в "САР МЛР" версии 1.0 реализована полно-

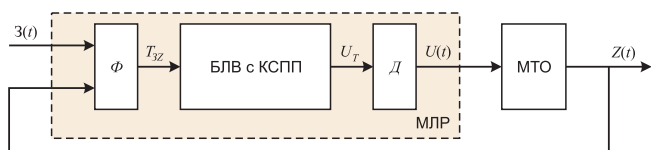


Рис. 5

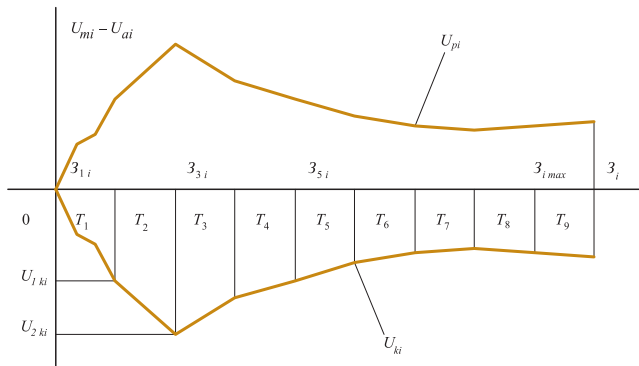


Рис. 7

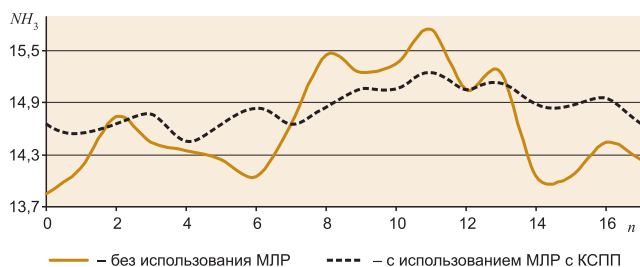


Рис. 8

ценная генерация программного кода МЛР в формате языка программирования SIMATIC S7-SCL для промышленных контроллеров SIMATIC S7-300/S7-400.

Приведем пример, иллюстрирующий алгоритм управления многомерным технологическим объектом (МТО) в рамках нечеткой логики, включающий производные правила и компенсацию перекрестных связей. На рис. 5 представлена структурная схема системы управления элементом дистилляции на основе МЛР с КСПП, где $Z(t)$, $Z(t)$ и $U(t)$ – векторы задающих, регулируемых и регулирующих параметров соответственно; T_{3Z} и U_T – векторы термов задающих, регулируемых и регулирующих параметров соответственно; БЛВ – блок логического вывода МЛР. Важным параметром дистилляционной колонны, значение которого требуется постоянно поддерживать в пределах, установленных технологическим регламентом, является концентрация аммиака NH_3 в жидкости [6] из дистиллера. На рис. 6 представлен вариант интерпретации вышеупомянутого параметра совокупностью четких термов, который предлагается использовать для построения регулирующей системы производных правил в "САР МЛР". Источником информации для построения КСПП являются экспериментально снятые зависимости задающего и регу-

Антипин Андрей Федорович – аспирант кафедры "Автоматизированные технологические и информационные системы (АТИС)" ГОУ ВПО Уфимский государственный нефтяной технический университет (г. Стерлитамак). Контактный телефон (917) 781-76-02. E-mail: andrejantipin@ya.ru

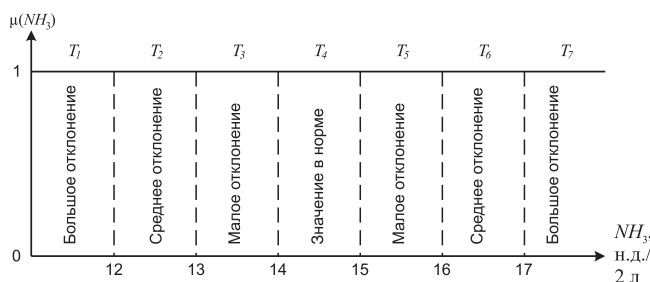


Рис. 6

лируемого параметра в автономном (U_{ai}) и много-связном (U_{mi}) режимах работы МЛР. На рис. 7 представлены графики разностной функции U_{pi} и функции компенсации U_{ki} влияния $(n - i)$ контуров на i -контур регулирования МЛР: $U_{pi} = U_{mi} - U_{ai}$.

Таким образом, для построения КСПП МЛР системы управления элементом дистилляции концентрация NH_3 в жидкости в определенный период будет поддерживаться в заданном диапазоне значений в сепаратном режиме работы МЛР (активен только контур подачи известкового молока), затем – в многосвязном режиме.

На рис. 8 представлены временные зависимости значений концентрации аммиака в жидкости, составленные на основе n ежечасных показаний приборов, в системе управления элементом дистилляции, из которых следует, что использование МЛР с КСПП позволило снизить перерегулирование в среднем на 48% и повысить точность регулирования концентрации аммиака на (43...45) %.

Таким образом, использование инструментария "САР МЛР" позволяет снизить продолжительность разработки МЛР на (45...55) %, а также создать конкурентное преимущество МЛР над многомерными нечеткими и дискретно-логическими регуляторами в линейке современных технологий и устройств управления сложными объектами и процессами с взаимосвязанными регулируемыми параметрами.

Список литературы

1. Усков А.А. Интеллектуальные системы управления на основе методов нечеткой логики. Смоленск: Смоленская городская типография. 2003.
2. Антипин А.Ф. Сравнительный анализ быстрействия дискретно-логического регулятора // Программные продукты и системы. 2010. № 1.
3. Муравьева Е.А., Антипин А.Ф. Многомерный дискретно-логический регулятор расхода воздуха парового котла с минимизацией времени отклика // Вестник УГАТУ. Серия "Управление, вычислительная техника и информатика". 2009. Т. 13. № 2.
4. Петров И.В. Программируемые контроллеры. Стандартные языки и приемы прикладного проектирования. М.: Солон-Пресс. 2004.
5. Беллман Р., Задэ Л. Принятие решений в расплывчатых условиях // Вопросы анализа и процедуры принятия решений. Сборник переводов. М.: Мир. 1976.
6. Крашенинников С.А. Технология кальцинированной соды и очищенного бикарбоната натрия. М.: Высшая школа. 1985.