

УДК 68.13

**Р. Н. Кветный, д. т. н., проф.; В. Ю. Коцюбинский, к. т. н., доц.;  
Л. Н. Кислица; Н. В. Казиминова; А. А. Кириленко**

## **АДАПТИВНАЯ СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ НА ОСНОВЕ НЕЧЕТКОГО ЛОГИЧЕСКОГО ВЫВОДА**

*В статье разработана адаптивная система поддержки принятия решений на основе правил нечеткой логики, которая учитывает результаты прогнозирования поведения объектов и может эффективно применяться в условиях неопределенности.*

**Ключевые слова:** система поддержки принятия решений, нечеткая логика, адаптивная система, прогнозирование случайных процессов.

### **Актуальность**

Принятие решений является одним из важных этапов для любой целенаправленной человеческой деятельности. В технических отраслях они обязательно рассматриваются перед разработкой технологии создания новых устройств, выбором управляющих воздействий сложными агрегатами или системами и т. д. Именно поэтому на сегодняшний день существует необходимость в создании системы поддержки принятия решений, которая заключалась бы в последовательном применении методов качественного и количественного характера для анализа развития изучаемого явления.

Существует множество подходов для разработки систем поддержки принятия решений, использующих различные подходы: корреляционный и регрессионный анализ, сценарные методы, теория игр, нечеткая логика и т. д. Но практически все предыдущие экспертные системы моделировали процесс принятия экспертом решения как дедуктивный процесс с использованием вывода, основывавшегося на классификационных правилах. Это означало, что в систему закладывалась совокупность правил вида "если ... то ...", согласно которым на основании входных данных генерировалось то или иное решение интересующей проблемы.

В последнее время развивается "неклассический" подход в теории управления и принятия решения. Он связан с применением алгоритмов на основе нечеткой логики, нейронных сетей и генетических алгоритмов, сценарных методов и т. д. Кроме того, широко используется ситуационное управление на основе иерархических моделей с нечеткими параметрами; модели и алгоритмы принятия решений для защиты информации на основе методов искусственного интеллекта [1].

Использование результатов моделирования и прогнозирования хода случайных процессов, описывающих поведение, – важный этап в процессе принятия решений для повышения их эффективности и снижения вероятности появления неверных решений. Именно поэтому актуальным является не только исследование, в какой степени результаты прогнозирования влияют на оценку альтернативных решений, но и разработка адаптивной системы поддержки принятия решений на основе результатов прогнозирования случайных процессов.

Как известно, принятие решений в проблемноориентированных информационных системах и системах управления осуществляется в условиях априорной неопределенности, обусловленной неточностью или неполнотой исходных данных, стохастической природой внешних воздействий, отсутствием адекватной математической модели, нечеткостью сформулированной цели, человеческим фактором [1, 2] и пр. Неопределенность системы может привести к увеличению рисков принятия неэффективных решений, в результате чего могут наблюдаться негативные экономические, технические и социальные последствия.

Неопределенности в системах принятия решений компенсируются различными методами

искусственного интеллекта. Для эффективного принятия решений при неопределенности условий функционирования системы применяют методы на основе правил нечеткой логики.

Такие методы основываются на нечетких множествах и используют лингвистические величины и выражения для описания стратегий принятия решений [3]. Одним из таких методов, который авторы предлагают использовать для разработки системы поддержки принятия решений, является метод нечеткого логического вывода. Это удобный механизм решения задач принятия решений, обеспечивающий прозрачность алгоритма принятия решений, легкость его корректировки, предоставляет возможность учитывать количественные значения и качественные характеристики моделируемых систем.

### **Постановка задачи**

**Целью работы** является разработка адаптивного подхода на основе правил нечеткой логики для создания экспертных систем поддержки принятия решений, которые могут учитывать результаты прогнозирования поведения объектов и эффективно применяться в условиях неопределенности.

### **Разработка структурной схемы адаптивной системы управления поддержки принятия решений на основе нечеткого логического вывода**

Адаптация – это процесс изменения параметров, структуры и действий системы на основе текущей информации с целью достижения оптимального состояния системы при начальной неопределенности в меняющихся условиях работы.

Адаптивная система (самообучающаяся система) – система, алгоритм функционирования которой строится и совершенствуется в процессе самообучения. Этот процесс сводится к «пробам» и «ошибкам». Система выполняет пробные изменения алгоритма и одновременно контролирует результаты этих изменений. Если они благоприятны с точки зрения целей управления, то изменения идут в том же направлении к достижению наилучших результатов или до начала ухудшения процесса управления [4, 5].

Структурная схема адаптивной системы поддержки принятия решений на основе нечеткого логического вывода представлена на рис. 1.

Данная система построена с учетом результатов прогнозирования будущего поведения объекта управления (ОУ) и процедур получения знаний от экспертов в области принятия решений.

На основе разработанной структурной схемы процесс принятия решений с помощью адаптивной системы состоит из следующих этапов:

*Этап 1.* Формирование базы знаний/пополнение новой информацией, который осуществляется в два подэтапа. Первый подэтап происходит при заполнении базы экспертными данными. Второй подэтап – это пополнение базы знаний новыми случаями, которые рассматриваются во время работы системы. После принятия решения и оценки результата его действия текущая ситуация превращается в предыдущий случай и заносится в базу данных. Отрицательный результат также является информативным и заносится в базу.

*Этап 2.* Прогнозирование заключается в описании поведения объекта исследования с помощью математической модели. Для этого необходимо определить тип случайного процесса, соответствующего объекту, с помощью набора идентификационных тестов и затем выбрать соответствующую математическую модель и проверить ее на адекватность и силу прогнозирования. Если результаты удовлетворительные, данная модель будет использоваться в качестве инструмента для оценки факторов, которые учитываются пользователем при принятии решения.

*Этап 3.* Формирование нечеткого логического вывода (НЛВ), состоящего из нескольких блоков. Рассмотрим подробнее их формирование (см. рис. 1).

**Блок фазификации.** В этом блоке происходит процесс фазификации – построение

нечетких множеств для лингвистических термов входных параметров на – факторов оценки состояния объекта:  $x_1, x_2 \dots x_m$ .

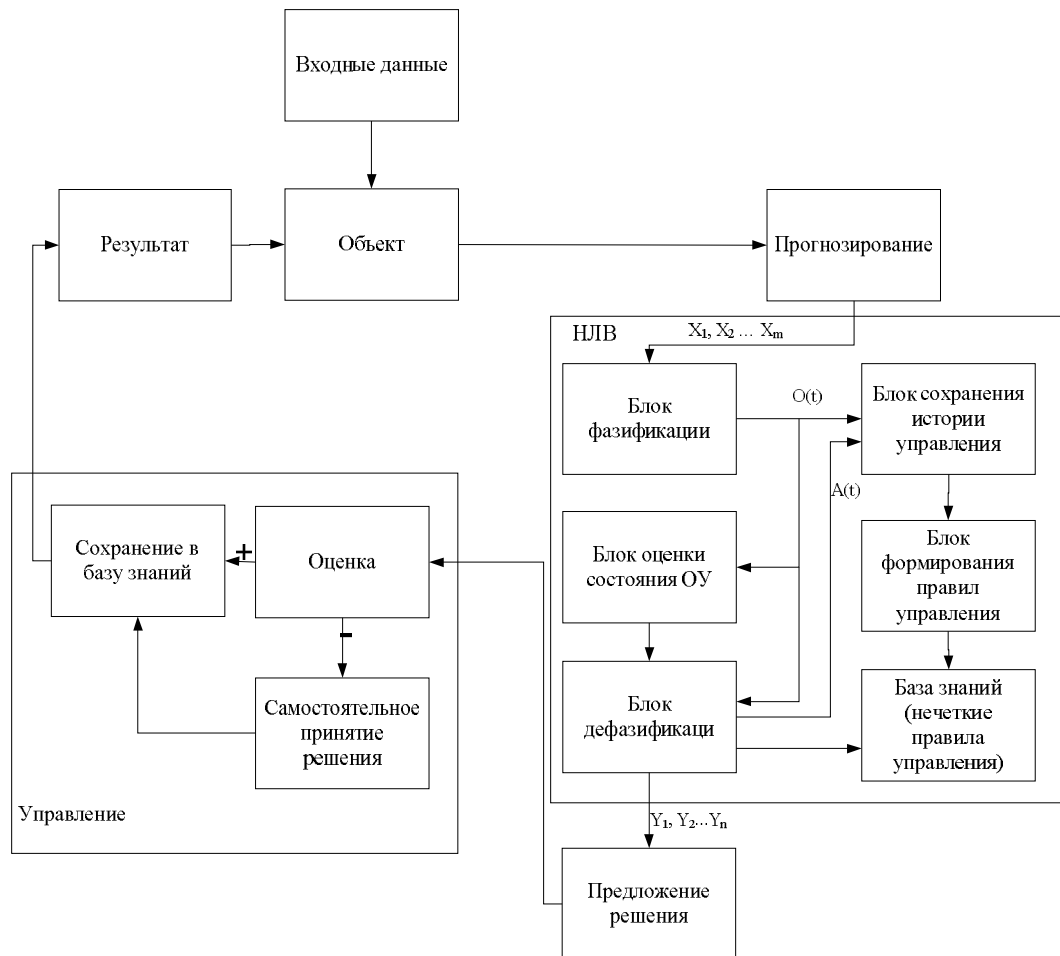


Рис. 1. Структурная схема адаптивной системы управления поддержки принятия решений на основе нечеткого логического вывода

В блоке оценки состояния ОУ формируется логический вывод на основе фазифицированных значений  $O(t)$ , которые затем передаются в блок дефазификации. Дефазификация осуществляется путем преобразования нечеткого множества в четкое число. Процедура дефазификации является процедурой получения решения с помощью нечеткой модели. Результаты дефазификации  $A(t)$  хранятся в базе знаний и в блоке хранения истории управления. На основе данных истории управления  $A(t)$  и  $O(t)$  могут быть сформированы новые правила управления.

Для данного этапа величина  $Y$  на выходе – это результат принятия решения с помощью нечеткого логического вывода, который передается в следующий блок-управления.

*Этап 4.* Управление выполняется пользователем для того, чтобы оценить сгенерированное системой решение и выполнить действие. Если пользователь соглашается с данным решением, то он подтверждает его. В случае несогласия ему необходимо самостоятельно выполнить процедуру принятия решения. Данные о случае и результаты сохраняются в базе знаний [6].

Таким образом, во время работы пользователя с системой происходит ее адаптация к его особенностям за счет пополнения исходной базы знаний информацией, полученной в результате принятых или не принятых пользователем решений.

### Разработка математических моделей для прогнозирования случайных процессов

Особенностью адаптивных систем является самообучение или способность подстраивать внутренние параметры под динамику прогнозируемого случайного процесса. Обучение может проводиться «без учителя» и «с учителем»: в первом случае изменение параметров модели происходит согласно внутреннему алгоритму, заложенному в модель, а во втором случае требуется явное указание, какое изменение лучше или хуже.

Для большинства ситуаций в качестве «мнения учителя» используется значение ошибки прогноза, называемое целевой функцией, и цель обучения – настроить параметры таким образом, чтобы это значение было минимальным.

Прежде чем непосредственно перейти к этапу принятия решения, необходимо рассмотреть анализ и прогнозирование случайных процессов, описывающих поведение объекта управления.

Анализ случайных процессов состоит из следующих этапов:

- идентификация вида случайного процесса;
- моделирование процесса;
- прогнозирование его поведения.

Вид случайного процесса требует построения и использования соответствующих математических моделей для их анализа и прогнозирования. Выделяют два основных класса случайных процессов – стационарные и нестационарные. Но за последние десятилетия стали наблюдать несколько иную природу временных рядов, которые отличаются от классических. Это так называемые процессы с длинной памятью, которые занимают промежуточное место в классификации между стационарными (короткая память) и нестационарными (бесконечная память) и требуют разработки новых моделей для их математического описания.

Учитывая разнообразную природу случайных процессов, перед их анализом необходимо определить вид процесса.

Для проверки стационарности и определения порядка интеграции исследуемых рядов используется несколько альтернативных тестов. Все тесты можно разделить на две большие категории в зависимости от того, какая гипотеза рассматривается как нулевая. Расширенный тест Дикки – Фуллера (ADF) и тест Филлипса – Перрона (PP), которые являются одними из наиболее популярных и известных инструментов анализа поведения временных рядов, проверяют нулевую гипотезу о нестационарности процесса при альтернативной гипотезе о том, что процесс стационарный [2, 3]. Кроме того, существует еще один мощный тест для проверки стационарности ряда – тест KPSS, который разработали Квятковский, Филлипс, Шмидт, Шин. В отличие от теста Дикки – Фуллера, он проверяет нулевую гипотезу о стационарности исходного ряда. Для тестирования длинной памяти существует еще один тест, принадлежащий этой группе, – тест Ломак (LoMac).

В результате проведения тестов ADF, KPSS и LoMac можно провести идентификацию случайного процесса и достоверно определить его тип – стационарный, нестационарный, с «длинной памятью» [3].

Далее необходимо выбрать математическую модель в соответствии с типом случайного процесса. Для их описания были выбраны следующие математические модели:

- для стационарного случайного процесса:

$$\begin{aligned}\varphi(L)y_t &= \Theta(L)\varepsilon_t \\ \varepsilon_t &\approx i.i.d.D(0,1),\end{aligned}\tag{1}$$

- для нестационарного случайного процесса:

$$\begin{aligned}\varphi(L)y_t &= \Phi(L)(1-L)^d y_t = \Theta(L)\varepsilon_t \\ \varepsilon_t &\approx i.i.d.D(0,1),\end{aligned}\tag{2}$$

- для процесса с длинной памятью:

$$\begin{aligned}\varphi(L)(1-L)^d y_t &= \Theta(L)\varepsilon_t \\ \varepsilon_t &\approx i.i.d.D(0,1),\end{aligned}\quad (3)$$

где

$$\begin{aligned}(1-L)^d &= 1 - dL - \frac{\alpha(1-d)}{2!}L^2 - \frac{\alpha(1-d)(2-d)}{3!}L^3 - \dots = \sum_{k=0}^{\infty} \frac{\Gamma(k-d)}{\Gamma(-d)\Gamma(k+1)}L^k = 1 - \sum_{k=0}^{\infty} c_k(d)L^k \\ 0 < d < 1, c_1(d) &= d, c_2(d) = \frac{1}{2}d(1-d), \dots \text{ и } \Gamma(\cdot) \text{ означает гамма-функцию } \Gamma(z) = \int_0^{\infty} t^{z-1} e^{-t} dt; \\ \Phi(L) &= 1 - \psi_1 L - \dots - \psi_p L_p; \Theta(L) = 1 + \theta_1 L + \dots + \theta_q L_q \quad - \text{ лаговые операторы процессов} \\ &\text{авторегрессии и скользящего среднего соответственно; } \varepsilon_t \text{ - «белый шум»}.\end{aligned}$$

Можно выделить еще одну характеристику случайного процесса, которая имеет значительное влияние на адекватность математической модели, – это явление гетероскедастичности. Если остатки временного ряда имеют постоянную дисперсию, то такие ряды называются гомоскедастичными, если они непостоянны, то – гетероскедастичными. Для ее определения предлагается использовать тест Лjung – Бокса и Уайта.

После идентификации гетероскедастичного типа процесса необходимо выбрать математическую модель, описывающую его поведение. В данной статье рассматриваются этапы разработки математических моделей для различных типов гетероскедастичных процессов.

*Стационарные процессы.* Рассмотрим случайный процесс  $y_t$ . Пусть после идентификации вида процесса и подбора адекватной модели, описывающей временной ряд, был сделан вывод, что процесс  $y_t$  является стационарным и описывается моделью ARMA( $p, q$ ) с определенными параметрами  $p$  и  $q$ . Применим GARCH-методы для коррекции полученной модели [4].

Тогда этот процесс может быть представлен как процесс ARMA( $m, p$ ):

$$[1 - \alpha(L) - \beta(L)]\varepsilon_t^2 = \alpha_0 + [1 - \beta(L)]v_t, \quad (4)$$

где  $m = \max\{p, q\}$  и  $v_t \equiv \varepsilon_t^2 - h_t$ .

Формально модель ARMA-GARCH( $p, q$ ) может быть определена следующим образом:

$$\begin{aligned}\Phi(L)y_t &= \Theta(L)\varepsilon_t \\ \varepsilon_t &= h_t z_t \\ z_t &\approx i.i.d.D(0,1) \\ h_t &= \alpha_0 + \alpha(L)\varepsilon_t + \beta(L)h_{t-1},\end{aligned}\quad (5)$$

где  $D(\cdot)$  – функция плотности распределения.

*Нестационарные процессы.* Пусть после идентификации вида процесса и подбора адекватной модели, описывающей временной ряд, был сделан вывод, что процесс  $y_t$  является нестационарным и описывается моделью ARIMA( $p, d, q$ ) с определенными параметрами  $p, d$  и  $q$ . Применим GARCH-методы для коррекции полученной модели. Тогда формально модель ARIMA-GARCH( $p, d, q$ ) может быть определена следующим образом:

$$\begin{aligned}\Phi(L)\omega_t &= \Theta(L)\varepsilon_t \\ \omega_t &= \Delta^d y_t \\ \varepsilon_t &= h_t z_t \\ z_t &\approx i.i.d.D(0,1) \\ h_t &= \alpha_0 + \alpha(L)\varepsilon_t + \beta(L)h_{t-1}.\end{aligned}\quad (6)$$

*Процессы с длинной памятью.* Поскольку в данной работе предполагается использование дробно интегрированных моделей AFRIMA для моделирования рядов с длинной памятью, характеризующиеся гиперболической автокорреляционной функцией, для таких случаев были разработаны специальные классы GARCH-моделей – это FIGARCH и HYGARCH [5, 6].

Частично интегрированный процесс GARCH (или FIGARCH  $(p, d, q)$ ) может быть описан путем замены в модели GARCH оператора первых разниц  $(1-L)$  оператором частичного дифференцирования  $(1-L)^d$ , где  $d$  – параметр памяти и  $0 < d < 1$ :

$$\Phi(L)(1-L)^d \varepsilon_t^2 = \alpha_0 + [1 - \beta(L)]v_t. \quad (7)$$

Возможность использования значений  $d$  в интервале от нуля до единицы придает модели дополнительную гибкость, которая может быть необходима при моделировании долгосрочных зависимостей, характерных для условной дисперсии многих временных рядов, например, финансовых [6].

Формально модель ARFIMA-FIGARCH  $(p, d, q)$  может быть определена следующим образом:

$$\begin{aligned} \Phi(L)(1-L)^d (y_t - \mu) &= \Theta(L)\varepsilon_t \\ \varepsilon_t &= h_t z_t \\ z_t &\approx i.i.d.D(0,1) \\ h_t(\alpha_0, \varphi, \beta, d) &= \alpha_0 + [1 - (1 - \beta(L))^{-1} \varphi(L)(1-L)^d \varepsilon_{t-1}^2]. \end{aligned} \quad (8)$$

Модель одновременно предполагает наличие длинной памяти с использованием различных типов распределения погрешностей и включением некоторых дополнительных переменных [7].

На практике, как правило, используют модели ARFIMA-FIGARCH  $(1, d, 0)$  или ARFIMA-FIGARCH  $(1, d, 1)$ , а параметр  $d$  рассчитывают методом максимального правдоподобия.

Было установлено, что FIGARCH-процессы тоже нестационарные, как и классические GARCH. Это означает, что концепцию существования единичного корня, характерную для линейных процессов, очень трудно применить для нелинейных. Кроме того, трудно выбрать параметр памяти  $d$ , чтобы модель FIGARCH была адекватной и имела высокую силу прогнозирования. Чтобы избежать указанных трудностей, класс FIGARCH-моделей был расширен. В данной статье предложена модель HYGARCH  $(p, d, q)$ , которая, в отличие от существующих, описывает гиперболические процессы GARCH.

В HYGARCH-моделях оператор  $(1-L)^d$  заменяют  $[(1-\alpha) + \alpha(1-L)]^d$ . Тогда модель HYGARCH может быть определена как:

$$h_t = \alpha_0 (1 - \beta(L))^{-1} + [1 - (\varphi(L)(1 + \alpha[(1-L)^d - 1]))(1 - \beta(L))^{-1} \varepsilon_t^2], \quad (9)$$

где параметры  $\alpha$  и  $d$  предполагаются положительными ( $\alpha > 0, d > 0$ ).

Формально модель ARFIMA-HYGARCH  $(p, d, q)$  может быть определена следующим образом:

$$\begin{aligned} \Phi(L)(1-L)^d (y_t - \mu) &= \Theta(L)\varepsilon_t \\ \varepsilon_t &= \sigma_t z_t \\ z_t &\approx i.i.d.D(0,1) \\ h_t &= \alpha_0 (1 - \beta(L))^{-1} + [1 - (\varphi(L)(1 + \alpha[(1-L)^d - 1]))(1 - \beta(L))^{-1} \varepsilon_{t-1}^2]. \end{aligned} \quad (10)$$

Следовательно, прежде чем перейти к этапу формирования нечеткого логического вывода и принятия решения на его основе, необходимо оценить поведение объекта, т. е. идентифицировать временной ряд, который его описывает, и подобрать соответствующую

математическую модель для его описания из рассмотренных в данном подразделе.

### Применение алгоритма Мамдани для формирования нечеткого логического вывода

Оценка решения в данной адаптивной системе осуществляется с помощью методов, основанных на правилах нечеткой логики, а именно – нечеткого логического вывода.

Существует несколько алгоритмов нечеткого вывода: Мамдани, Цукамото, Сугено и Ларсен. Среди них наибольшую популярность имеет алгоритм Мамдани. Прозрачность нечетких моделей Мамдани является одним из главных преимуществ, благодаря которому нечеткие технологии успешно конкурируют с другими методами. Они наиболее подходят для тех прикладных задач, где возможность содержательной интерпретации важнее точности моделирования.

Проиллюстрируем применение данного метода на примере разработки правил для принятия решений на финансовых рынках.

Пусть база нечетких правил принятия решений содержит определенные экспертами зависимости прибыли от некоторых входных переменных  $X_1$  и  $X_2$ , где  $X_1$  – это положение цены финансового актива относительно скользящего среднего (СС), а  $X_2$  – изменение процентной ставки. Введем лингвистические переменные:  $X_1 = (H, MH, L, ML)$   $X_2 =$  (снижение ставки, неизменная ставка, повышение ставки); Результат = (большая прибыль, низкая прибыль, низкие убытки, большие убытки).

Приведем несколько из возможных правил:

$R_1$ : если  $X_1 \in L$  и  $X_2 \in$  неизменная ставка, то результат – низкие убытки;

$R_2$ : если  $X_1 \in H$  и  $X_2 \in$  снижение ставки, то результат – низкие убытки;

$R_3$ : если  $X_1 \in L$  и  $X_2 \in$  снижение ставки, то результат – большой доход.

Предположим, что лингвистические термы входов описываются такими нечеткими множествами:

$H = \{-2/0; -1,5/0; 0/0,2; 1,5/0,6; 2/0,3\};$

$L = \{-2/0,1; -1,5/0,5; 0/0,2; 1,5/0; 2/0\};$

неизменная ставка =  $\{-0,25/0,2; -0,2/0,5; 0/1; 0,2/0,4; 0,25/0,15\};$

снижение ставки =  $\{-0,25/0,5; -0,2/0,2; 0/0; 0,2/0; 0,25/0\}.$

Термы выхода описываются такими множествами:

низкие убытки =  $\{-150/0,5; -100/1; 0/0,1; 100/0; 150/0\};$

большая прибыль =  $\{-150/0; -100/0; 0/0; 100/0,1; 150/0,6\}.$

Необходимо определить результат при  $ML$  и повышении ставки.

Обратим внимание на то, что входные данные не определяют термов  $ML$  и повышение ставки. Исходную реакцию на эти нечеткие значения необходимо получить в процессе логического вывода на основе базы правил.

Пусть на вход системы поступают нечеткие множества:

$ML = \{-2/0,7; -1,5/0,25; 0/0; 1,5/0; 2/0\};$

и изменение ставки:

увеличение ставки =  $\{-0,25/0; -0,2/0; 0/0; 0,2/0,2; 0,25/0,5\}.$

Операции определения минимума и максимума обозначим в виде  $\wedge$  и  $\vee$  соответственно.

Для вычисления выхода выполним этапы нечеткого логического вывода:

1. Вычисление уровней истинности правил.

$a_1 = \min[\max(0,7 \wedge 0,1; 0,25 \wedge 0,5; 0 \wedge 0,2; 0 \wedge 0; 0 \wedge 0), \max(0 \wedge 0,2; 0 \wedge 0,5; 0 \wedge 1; 0,2 \wedge 0,4; 0,5 \wedge 0,15)] = \min[\max(0,1; 0,25; 0; 0; 0), \max(0; 0; 0; 0,2; 0,15)] = \min[0,25; 0,2] = 0,2$

$a_2 = \min[\max(0,7 \wedge 0; 0,25 \wedge 0; 0 \wedge 0,2; 0 \wedge 0,6; 0 \wedge 0,3), \max(0 \wedge 0,5; 0 \wedge 0,2; 0 \wedge 0; 0,2 \wedge 0; 0,5 \wedge 0)] = \min[\max(0; 0; 0; 0; 0; 0), \max(0; 0; 0; 0,2; 0,15)] = \min[0; 0] = 0$

$a_3 = \min[\max(0,7 \wedge 0,1; 0,25 \wedge 0,5; 0 \wedge 0,2; 0 \wedge 0; 0 \wedge 0), \max(0 \wedge 0,5; 0 \wedge 0,2; 0 \wedge 0; 0,2 \wedge 0; 0,5 \wedge 0)] =$

$\min[\max(0,1; 0,25; 0; 0; 0), \max(0; 0; 0; 0; 0)] = \min[0,25; 0] = 0$ .

2. Вычисление выходов правил.

$B_1 = \{-150/\min(0,2; 0,5); -100/\min(0,2; 1); 0/\min(0,2; 0,1); 100/\min(0,2; 0); 150/\min(0,2; 0)\}$   
 $= \{-150/0,2; -100/0,2; 0/0,1; 100/0; 150/0\};$

$B_2 = \{-150/\min(0; 0,5); -100/\min(0; 1); 0/\min(0; 0,1); 100/\min(0; 0); 150/\min(0; 0)\} = \{-150/0;$   
 $-100/0; 0/0; 100/0; 150/0\};$

$B_3 = \{-150/\min(0; 0,5); -100/\min(0; 1); 0/\min(0; 0,1); 100/\min(0; 0); 150/\min(0; 0)\} = \{-150/0;$   
 $-100/0; 0/0; 100/0; 150/0\};$

3. Агрегирование выходов.

$B = B_1 \vee B_2 \vee B_3 = \{-150/\max(0,2; 0; 0); -100/\max(0,2; 0; 0); 0/\max(0,1; 0; 0); 100/\max(0; 0; 0);$   
 $150/\max(0; 0; 0)\} = \{-150/0,2; -100/0,2; 0/0,1; 100/0; 150/0\}.$

4. Дефазификация выхода.

$$y = \frac{-150 \cdot 0,2 - 100 \cdot 0,2 + 0 \cdot 0,1 + 100 \cdot 0 + 150 \cdot 0}{0,2 + 0,2 + 0,1 + 0 + 0} = -100.$$

Таким образом, для заданных нечетких множеств при *ML* и *повышении ставки* результатом будут *низкие убытки*.

Учитывая все вышеуказанное, определим этапы принятия решений в условиях неопределенности с использованием алгоритма Мамдани:

1) выбор факторов  $X_k$ , на основе которых будет приниматься решение. Для каждого фактора задается множество его значений (терм-множество), задаются функции принадлежности для каждого лингвистического термина из базового терм-множества. А также определяется уровень истинности для каждого правила  $a_m$ ;

2) вычисление выходов каждого из правил  $B_m$  с использованием операции минимума;

3) объединение всех нечетких множеств, полученных на выходе правил, с использованием операции максимума в единое нечеткое множество  $B$ ;

4) переход от нечеткого множества к конкретному значению  $y$  (в случае необходимости).

Таким образом, использование блока нечеткого логического вывода для принятия решения имеет следующие преимущества: возможность оперировать нечеткими входными данными, нечеткая формализация критериев оценки и сравнения, введение зависимостей на языке, близком к естественному.

### Адаптивная система поддержки принятия решений «TradeKeeper»

На основе вышеописанного подхода принятия решений в условиях неопределенности была разработана экспертная адаптивная система «TradeKeeper», позволяющая пользователю анализировать финансовое поведение активов и в результате облегчающая принятие решения и повышающая его эффективность.

«TradeKeeper» состоит из следующих компонентов: общего и пользовательского интерфейса. Система предоставляет возможность зарегистрироваться и работать под отдельным логином. Каждый пользователь использует собственную базу знаний и механизм нечеткого логического вывода для создания стратегии принятия решений. Процесс формирования начальной выборки осуществляется на основе решений, полученных от экспертов на финансовом рынке, решавших реальные задачи. В процессе использования системы с каждым принятым решением выборка меняется, формируя собственную базу знаний пользователя, учитывает его особенности работы и психо-эмоциональные особенности.

Для принятия решения используются следующие факторы:

1)  $X_1$  – положение цены относительно скользящего среднего (лингвистический тип фактора);

2)  $X_2$  – изменения процентной ставки (количественный тип фактора);



3)  $X_3$  – соблюдение ценовой модели, изменения процентной ставки (лингвистический тип фактора);

4)  $X_4$  – субъективное состояние трейдера (дискретный тип фактора).

Возможные результаты принятия решения следующие:

- 1) Class 1 – низкий доход (менее 1%);
- 2) Class 2 – большой доход (более 1%);
- 3) Class 3 – низкие убытки (менее 1%);
- 4) Class 4 – большие убытки (более 1%).

Система предоставляет возможность с помощью он-лайн-ресурса Yahoo.Finance получить котировки необходимых финансовых инструментов, с которыми планируется работа. Выполняя некоторые операции, пользователь может создать собственную стратегию. Полученная стратегия помогает принимать решения для дальнейшей успешной торговли. На рис. 2 приведен пример стратегии, построенной с помощью мастера стратегий.

The screenshot displays the 'Strategy Wizard' interface with a navigation bar at the top containing links: 'Strategy Wizard', 'Start a Trade', 'Close a Trade', 'Past Trades', 'Import Trades', 'My Info', 'Management', 'Main', and 'Logout'. On the left, a vertical sidebar lists 'Step 1' through 'Step 4', with 'Step 4' currently selected. The main content area is titled 'STRATEGY INFORMATION:' and contains the following details:

- Strategy Name: *Test Strategy*
- Strategy Description: *first test strategy*
- Horizon: 213
- Tradeout: 43
- # of Trades: 7

Below this, the 'RULES:' section lists two rules:

- Rule 1:**
  - Rule Name: *Numeric value + newtestRule*
  - Indicator Name: *Numeric value*
  - Indicator Type: *Numeric*
  - Numeric Value: 5
  - Close Value: 5
  - Website Start URL:
  - Website End URL:
  - Indicator Values:
- Rule 2:**
  - Rule Name: *testIndicator + meCondition*
  - Indicator Name: *testIndicator*
  - Indicator Type: *Cardinal*
  - Website Start URL:
  - Website End URL:
  - Indicator Values:

At the bottom of the main area, it states: 'This strategy is using Decision Tree: NewDecisionTree2'. At the very bottom right, there are two buttons: 'Previous' and 'Finish'.

Рис. 2. Пример стратегии, построенной с помощью мастера стратегий

Кроме того, система позволяет не только создавать собственные стратегии управления и анализа финансовых операций, также пользователь может создавать собственные индикаторы, устанавливать и изменять их параметры, условия, добавлять новые правила классификации и нечеткие правила управления в базу знаний, которая их использует.

### Выводы

Разработана адаптивная система поддержки принятия решений на основе нечеткого логического вывода, которая учитывает результаты прогнозирования поведения случайных процессов (стационарных, нестационарных, с длинной памятью), описывающих объект управления. Такая система может быть эффективно применена для принятия решений, например, на рынках ценных бумаг при работе с финансовыми активами или в других отраслях человеческой деятельности.

Построена структурная схема такой адаптивной системы, детально описаны этапы ее работы с пользователем и приведен пример применения алгоритма Мамдани для формирования нечетких логических выводов. Адаптивная система поддержки принятия решений позволяет накапливать знания о решениях, принятых пользователем, в базе данных и использовать ее в дальнейшем при нечетком логическом выводе.

Разработанная модель адаптивной системы реализована в виде системы поддержки принятия решений «TradeKeeper» для работы с финансовыми активами. В данной системе использован нечеткий механизм автоматической классификации принимаемых пользователем решений в зависимости от ожидаемого значения прибыли как сигнала открытия той или иной позиции. Данная функциональность является преимуществом системы «TradeKeeper», поскольку повышает эффективность принятия решений пользователем и позволяет получить максимальную прибыль от совершаемых финансовых операций.

## СПИСОК ЛИТЕРАТУРЫ

1. Бочарников В. П. Fuzzy-технология: Математические основы. Практика моделирования в экономике / В. П. Бочарников. – Санкт-Петербург: «Наука» РАН, 2001. – 328 с.
2. Субботін С. О. Подання й обробка знань у системах штучного інтелекту ті підтримки прийняття рішень: Навчальний посібник/ С. О. Субботін. – Запоріжжя: ЗНТУ, 2008. – 341 с.
3. Блюмин С. Л. Модели и методы принятия решений в условиях неопределенности / С. Л. Блюмин, И. А. Шуйкова. – Липецк: ЛЭГИ, 2001. – 138 с.
4. Попов Е. П. Теория систем автоматического регулирования / Е. П. Попов., В. А. Бесекерский. – М.: Наука, 2003. – 759 с.
5. Ропштейн А. П. Нечеткая надежность алгоритмических процессов / А. П. Ропштейн, С. Д. Штовба. – Винница: Континент – ПРИМ, 1997. – 142 с.
6. Kvetniy R. N. Using of adaptive approach to make decision in difficult systems / R. N. Kvetniy, V. Y. Kotsubinskiy, L. N. Kisliitsa, N. V. Kazimirova // Матеріали Міжнародної науково-технічної конференції «ІРТК-2009», 2009. – С. 15 – 17.
7. Ферстер Э. Методы корреляционного и регрессионного анализа / Э. Ферстер, Б. Ренц. – М.: Финансы и статистика, 1983. – 302 с.
8. Бокс Дж. Анализ временных рядов. Прогноз и управление. Вып. 1 / Дж. Бокс, Г. Дженкинс. – М.: Мир, 1974. – 408 с.
9. Кендалл М. Многомерный статистический анализ и временные ряды / М. Кендалл, А. Стьюарт. – М.: Наука, 1976. – 265 с.
10. Bollerslev T. Fractionally integrated generalized autoregressive conditional heteroskedasticity / T. Bollerslev, R. T. Baillie, H. O. Mikkelsen // Journal of econometrics. – 1996. – № 74, – P. 3 – 30.
11. Baillie R. T. Analyzing inflation by the fractionally integrated ARFIMA-GARCH model / R. T. Baillie, C.-F. Chung, M.A. Tieslau // Journal of Applied Econometrics. – 1996. – № 15.– P. 12 – 60.
12. Штовба С. Д. Введение в теорию нечетких множеств и нечеткую логику [Электронный ресурс] // Режим доступа: <http://matlab.exponenta.ru/fuzzylogic/book1/index.php>.
13. Кравець П. Системи прийняття рішень з нечіткою логікою / П. Кравець, Р. Киркало // “Вісник національного університету “Львівська політехніка”. – 2009. – № 650. – С. 115 – 123.
14. Р. Н. Кветний Адаптивна експертна система підтримки прийняття рішення в Internet-трейдингу / Р. Н. Кветний, В. Ю. Коцюбинський, Л. М. Кислиця, Н. В. Казимірова // Міжнародний науково-технічний журнал «Інформаційні технології та комп'ютерна інженерія». – Вінниця.: ВНТУ, 2009. – Вип.№ 2(15). – С. 81 – 85.

**Кветний Роман Наумович** – д. т. н., проф., заведуючий кафедрой автоматизации и информационно-измерительной техники. (0432) 598243.

**Коцюбинський Владимир Юрьевич** – к. т. н., доцент кафедри автоматизации и информационно-измерительной техники. (0432) 598243.

**Кислиця Людмила Николаевна** – магистр, соискатель кафедры автоматизации и информационно-измерительной техники. (0432) 598243, [lus83@mail.ru](mailto:lus83@mail.ru).

**Казимирова Нина Владимировна** – магистр, аспирант кафедры автоматизации и информационно-измерительной техники. (0432) 598243, [nkazimirova@spilnasprava.vn.ua](mailto:nkazimirova@spilnasprava.vn.ua).

**Кириленко Анна Александровна** – студентка кафедры автоматизации и информационно-измерительной техники, (0432) 598243, [anyakurul1@rambler.ru](mailto:anyakurul1@rambler.ru).

Винницкий национальный технический университет.