

ОРИГИНАЛЬНАЯ СТАТЬЯ

УДК 007.51: 004.891

<https://doi.org/10.26907/2541-7746.2025.1.181-195>**Прототипирование видеоигр на основе нечеткой логики****А.В. Шубин***Казанский (Приволжский) федеральный университет, г. Казань, Россия**shubin.aleksey.kpfu@gmail.com***Аннотация**

Рассмотрено применение нечеткой логики при создании игровых механик. Отмечена важность этой логики для процесса дизайна видеоигр, который требует сбалансированного подхода, объединяющего творчество и внимание к деталям. Обсуждены ограничения существующих средств и методов прототипирования и аргументирована необходимость разработки системы, способной интегрировать экспертные знания для принятия решений.

Анализ подобных исследований показывает, что нечеткая логика, которая хорошо моделирует процесс мышления человека, может предоставить удобный метод для рекомендации игровых механик на первых этапах разработки. Данное направление может оказаться более актуальным, особенно по сравнению с затратами при разработке и использовании нейронных сетей. Проектирование системы, предложенной в настоящей работе, не требует разработки сложных алгоритмов, что обеспечивает относительно простое усовершенствование разработанного подхода, позволяя расширить объем входных данных. Кроме этого, имеющаяся возможность легко усовершенствовать используемую экспертную базу знаний обеспечит уточнение информации и адаптацию системы к изменяющимся требованиям.

В статье также охарактеризованы разнообразные варианты применения нечеткой логики и ее потенциал для улучшения дизайна игр через поддержку выбора игровых механик с использованием экспертных систем.

Ключевые слова: игровой дизайн, игровая механика, автоматизация, нечеткая логика

Благодарности. Работа выполнена за счет средств Программы стратегического академического лидерства Казанского (Приволжского) федерального университета («ПРИОРИТЕТ-2030»).

Для цитирования: Шубин А.В. Прототипирование видеоигр на основе нечеткой логики // Учен. зап. Казан. ун-та. Сер. Физ.-матем. науки. 2025. Т. 167, кн. 1. С. 181–195.
<https://doi.org/10.26907/2541-7746.2025.1.181-195>.

ORIGINAL ARTICLE

<https://doi.org/10.26907/2541-7746.2025.1.181-195>

Video game prototyping using fuzzy logic

A.V. Shubin

Kazan Federal University, Kazan, Russia

shubin.aleksey.kpfu@gmail.com

Abstract

The use of fuzzy logic in the development of game mechanics was analyzed. The important role of fuzzy logic in the video game design process, which requires a balance between creativity and attention to detail, was emphasized. The shortcomings of current prototyping tools and methods were discussed. The growing need for a system effectively incorporating expert knowledge into decision-making was highlighted.

The results of previous research on this problem show that fuzzy logic, which mimics human reasoning, can be successfully used for selecting game mechanics in the early stages of game development. Hence, it appears to be a good alternative to resource-intensive neural networks. The design of the system proposed here does not require the construction of complex algorithms, offering a relatively simple refinement of the developed approach by enabling an expansion of the scope of input data. In addition, the possibility to easily improve the expert knowledge base used ensures the refinement of information and the adaptation of the system to changing requirements.

Various applications of fuzzy logic were covered, and its potential to improve game design by supporting the selection of game mechanics using expert systems was considered.

Keywords: game design, game mechanics, automation, fuzzy logic

Acknowledgments. This study was supported by the Kazan Federal University Strategic Academic Leadership Program (PRIORITY-2030).

For citation: Shubin A.V. Video game prototyping using fuzzy logic. *Uchenye Zapiski Kazanskogo Universiteta. Seriya Fiziko-Matematicheskie Nauki*, 2025, vol. 167, no. 1, pp. 181–195.
<https://doi.org/10.26907/2541-7746.2025.1.181-195>. (In Russian)

Введение

Создание видеоигр и игровой дизайн – комплексная задача, которая требует внимания к деталям, творческого подхода и баланса между различными аспектами игры для создания увлекательного и сбалансированного игрового опыта. Понимание различных аспектов видеоигры и их связь между собой могут показаться неочевидными для начинающего разработчика. Поэтому, как и во многих других сферах, одним из главных критериев хорошего игрового дизайнера является опыт разработки.

На плечи геймдизайнера ложится ответственность за проектирование видеоигры, разработку игрового процесса и, в частности, игровых механик, а также контроль реализации и согласованности различных элементов видеоигры друг с другом. Считается, что единственный и наиболее эффективный на данный момент подход к разработке видеоигр – итеративное развитие прототипов с последующим анализом игрового процесса и жизнеспособности игровых механик [1]. При этом создание прототипов – достаточно дорогой и долгий процесс, который требует от игрового дизайнера специфических знаний языка программирования или создания видеоигр на более простом движке. В связи с этим некоторые опытные разработчики видеоигр [2] предлагают создавать нецифровые прототипы, например, в виде классической настольной игры на бумаге, что является значительно устаревшей практикой.

Конечно, существуют инструменты, помогающие геймдизайнерам в тестировании различных концепций, такие как Machinations, Twine, Articy:draft, или инструменты, расширяющие функционал игровых движков (например, ProBuilder, Fungus). При этом они не обеспечивают поиск и развитие игровых механик, а лишь позволяют протестировать придуманные концепции. В некоторых работах (например, [3–5]) предложено использовать специальные инструменты для быстрого прототипирования видеоигр, например, когда разработчик может автоматически сгенерировать раскладовку внутриигровых видеороликов (катсцен) на основе сценария или другого текста.

Таким образом, существует необходимость в создании специфичного инструмента, который помог бы геймдизайнерам выбирать оптимальные варианты игровой механики, учитывая определенные параметры. Создание такой системы принятия решений на основе экспертной информации может помочь молодым геймдизайнерам в подборе наиболее подходящих механик. Кроме этого, опытные геймдизайнеры могут получить быстрое решение, на основе которого они будут базироваться при создании видеоигры.

Учитывая все возможные методы реализации, было выбрано создать систему принятия решения, основанную на принципах нечеткой логики. Этот выбор обусловлен рядом факторов:

1. Упрощенная реализация не требует глубоких знаний в области архитектуры нейронных сетей и обладает более низким порогом входа в сравнении с другими методами.
2. Модель нечеткой логики близка к процессам человеческого мышления. Это является важной особенностью метода за счет более естественного моделирования процесса принятия решений.
3. Система на основе нечеткой логики облегчает процесс интеграции экспертных знаний и оценок в структуру принятия решений.
4. Концепция разрабатываемого инструмента не требует сложных и продвинутых алгоритмов, следовательно, использование данного подхода будет достаточным для реализации системы принятия решений.

1. Связанные работы

Теория нечетких множеств и нечеткая логика, предложенные американским ученым Лотфи Заде в 1965 году (см., например, [6]), представляют собой расширение традиционной теории множеств и формальной логики. Эти концепции возникли как ответ на потребность в описании человеческих мыслительных процессов, систем и объектов, которые часто характеризуются неопределенностью и приблизительностью.

Несмотря на упавшую популярность на фоне искусственных нейронных сетей, нечеткая логика нашла применение и активно используется в различных отраслях и промышленности:

1. Применение нечеткой логики в системе антиблокировки тормозов (ABS) [7]. Нечеткая логика может быть использована для улучшения производительности и надежности систем ABS, обеспечивая более точное и адаптивное управление тормозным процессом в различных дорожных условиях.
2. Контроль различных параметров бытовой техники [8, 9]. Стиральные машины с нечетким управлением определяют количество воды, время стирки и количество моющего средства на основе загрязненности белья и его объема. Подобные системы применяются также в современных микроволновых печах и роботах-пылесосах.
3. Нечеткая логика при установке диагноза и медицине в целом [10]. Нечеткая логика применяется для помощи в диагностике заболеваний и анализе медицинских данных, где часто присутствуют неопределенность и субъективность.

Кроме этого, нечеткие логические системы находят применение в игровой индустрии. Некоторые работы предлагают использование нечеткой логики в создании искусственного интеллекта (например, [11, 12]); несмотря на нечастое использование данного подхода, имеются примеры видеоигр с искусственным интеллектом на основе нечеткой логики. Такими примерами являются известные видеоигры The Sims или Civilization: Call to Power, использующие нечеткие конечные автоматы для управления неигровыми персонажами и другими параметрами видеоигр.

Нечеткая логика также нашла применение в разработке систем принятия решений для поиска подходящих видеоигр. В статье [13] описана возможность создания экспертной системы на основе четырех параметров: оценка критиков, оценка пользователей, глобальные продажи и год выпуска видеоигры. Кроме этого, указанные авторы подготовили шестнадцать правил для отбора наиболее подходящей видеоигры на основе выбранных параметров. По словам этих авторов, точность такой системы достигает отметки в 80 %, а в будущем, используя гибридные подходы, планируется еще повысить точность данной системы.

В некоторых работах представлена разработка адаптивных сценариев игр на основе нечеткой логики (см. [14, 15]). Так, нечеткая логика используется для более реалистичного представления уровня знаний учащегося [14]. Прогресс игрока оценивается на основе успеха в двух предыдущих тестах, результат которых влияет на соотношение сложных и простых вопросов в последующих заданиях. В другой работе [15] на основе частоты сердцебиения и уровня стресса определяется состояние больного посттравматическим стрессовым расстройством. Так как частота сердцебиения является целочисленным значением,

а уровень стресса не может быть оценен объективно, то нечеткая логика позволяет коррелировать точную и неточную переменные.

Считается (например, [16]), что применение нечеткой логики в индустрии видеоигр на данный момент невелико. Поэтому есть неисследованные знания и практики, которые могут быть эффективно применимы к различным игровым системам.

2. Разработка модели нечеткой логики

Рассмотрим универсальное множество E , содержащее элементы X , обладающие определенным свойством R . Четкое подмножество A этого универсального множества E , состоящее из элементов, соответствующих свойству R , определяется как набор упорядоченных пар $A = \{\mu A(x)/x\}$, где $\mu A(x)$ выступает в роли функции принадлежности. Значения этой функции принадлежат строго упорядоченному множеству M , что отражает меру или уровень, на котором элемент x принадлежит подмножеству A . Множество M в данном контексте именуется множеством принадлежностей. Если $M = \{0, 1\}$, то нечеткое подмножество A может быть идентифицировано как четкое множество. Основное отличие нечеткого множества заключается в том, что для элементов X из подмножества E наличие у них свойства R не определяется однозначно (т. е. не сводится к простым ответам «да» или «нет»). Следовательно, нечеткое подмножество A универсального множества E формируется как совокупность упорядоченных пар, где функция принадлежности $\mu A(x)$ выражает степень принадлежности элемента X к подмножеству A .

В нечеткой логике основную роль выполняют так называемые «лингвистические переменные», представляющие собой способ описания качественных характеристик объекта или явления с помощью языковых понятий. Примерами таких переменных могут быть даже такие абстрактные понятия, как «комфорт», «красота», «спокойствие», которые трудно оценить численно – они имеют лишь субъективное количественное определение.

Частью лингвистической переменной является понятие «терм». Каждый терм представляет собой метку, связанную с функцией принадлежности, которая определяет степень, в которой конкретное значение относится к данному терму. Например, для лингвистической переменной «температура» в качестве термов могут быть использованы понятия «пониженная», «нормальная», «высокая» (рис. 1).

Для обработки данных и получения результатов экспертной системой в нечеткой логике используется операция логического вывода на основе базы правил. База правил модели представляет собой высказывания R^k в форме «ЕСЛИ–ТО»:

R^k : ЕСЛИ (X_1 ЕСТЬ A_1^k И X_2 ЕСТЬ A_2^k И ... И X_n ЕСТЬ A_n^k), ТО (Y_j ЕСТЬ B_j^k),

где X_i ($i \in N$) – входные лингвистические переменные,

Y_j ($j \in N$) – выходные переменные,

A_i^k , B_j^k ($k \in N$) – заданные функции принадлежности нечетким множествам, термы.

При этом при задании правил должны соблюдаться следующие условия [17]:

1. Существует хотя бы одно правило для каждой лингвистической выходной переменной.
2. Для любого терма выходной переменной имеется хотя бы одно правило, в котором этот термин используется в качестве целевой части правила. В противном случае имеет место база нечетких правил.

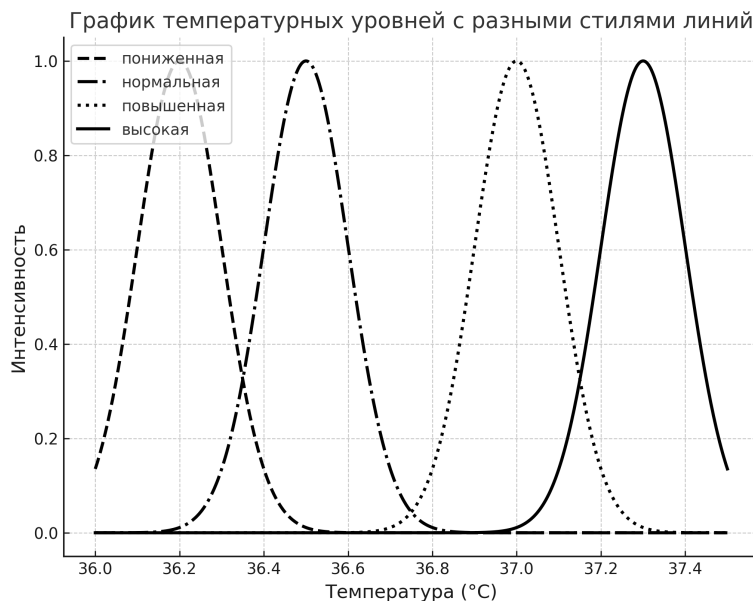


Рис. 1. График функций принадлежности четырех термов лингвистической переменной «температура»

Fig. 1. Membership functions curves for four linguistic terms of the linguistic variable “temperature”

Типовая схема работы системы принятия решений на основе нечеткой логики выглядит следующим образом (рис. 2).

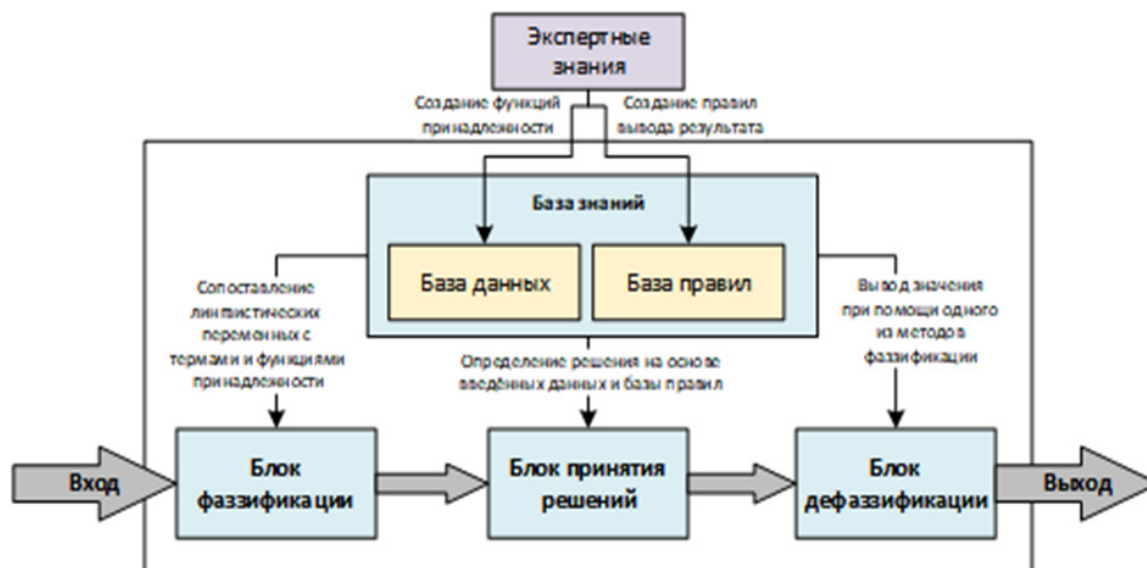


Рис. 2. Типовая архитектура нечеткой системы [18]

Fig. 2. General architecture of a fuzzy system [18]

Входные переменные проходят сначала через первый блок (фаззификатор), в котором осуществляется соответствие между численным значением входной переменной нечеткой системы и значением функции принадлежности соответствующего термина лингвистической переменной, т. е. в блоке фаззификации происходит преобразование четкого значения переменной в нечеткое.

База знаний – это блок нечеткой системы, предназначенный для хранения информации о переменных, а также необходимый для формирования решения. База знаний состоит из двух блоков:

1. База данных. Содержит представление о лингвистических переменных и значения функции принадлежности термов.
2. База правил. Хранит набор утверждений вида «Если – То», на основе которых принимается решение в блоке принятия решений.

Блок принятия решений объединяет все правила, активированные входными переменными, для формирования одного нечеткого множества, а затем агрегирует все выходные нечеткие множества для формирования.

Дефаззификация – это процесс преобразования нечеткого вывода в четкое значение. Данный процесс полностью зависит от выбранного метода фаззификации и позволяет вычислить взвешенное решение относительно всех возможных выводов.

3. Проектирование инструмента

Разработка системы на основе нечеткой логики, которая способна предлагать различные решения на основе вводимых переменных, не является сложным процессом. Учитывая архитектуру и работу системы, для реализации требуется выполнение пяти шагов:

1. Выбор нескольких параметров, которые могут хорошо описать характеристики механики, определение термов.
2. Сбор экспертной информации об игровых механиках и их характеристиках для разработки базы знаний.
3. Связывание термов с числовыми значениями, определив функцию принадлежности.
4. Выбор и реализация метода дефаззификации нечетких данных.
5. Разработка системы в среде разработки нечеткой модели.
6. Тестирование и валидация результатов.

В рамках настоящего исследования было решено остановиться на четырех параметрах, которые могут охарактеризовать игровые механики (табл. 1). Выбор конкретных механик обусловлен тем, что они оказывают непосредственное воздействие на игровой процесс и влияют на удовлетворение от игры, что делает их более ценными для игроков.

Для правильной работы системы требуются наличие в базе знаний необходимых данных для связи лингвистических переменных с конкретными термами на этапе фаззификации, а также блок правил для вывода экспертного решения. Для этого можно воспользоваться методом сбора и обработки статистической экспертной информации [17]. В основе такого метода лежит процесс заполнения каждым экспертом опросника, в котором он указывает принадлежность или непричастность (при $M = \{0, 1\}$) какого-либо значения или свойства $u_1, u_2, \dots, u_n$ относительно термов $l_1, l_2, \dots, l_n$.

Для проведения эксперимента было опрошено пять экспертов, которые отнесли термы параметра механики с конкретным представителем системы механик (рис. 3). В силу большого разнообразия игровых механик было решено ограничиться лишь тремя игровыми

системами: «Боевая система», «Головоломки», а также механики, дополняющие игровой процесс. Выбор механик пал на наиболее релевантные игровые механики из каждой системы. На основе опроса были сформированы зависимости параметров от игрового жанра, что позволило определить нечеткие правила системы (табл. 2).

Табл. 1. Выбранные лингвистические переменные и термины

Table 1. Selected linguistic variables and terms

Лингвистическая переменная	Числовые данные	Термы			
Сложность механики	Время обучения (часы) $x = [0; 6]$	Просто	Комфортно	Сложно	–
Динамичность механики	Числовая оценка (баллы) $x = [0; 10]$	Монотонно	Умеренно	Оживленно	–
Агентивность игрока	Числовая оценка (баллы) $x = [0; 10]$	Малая	Ограничена	Высокая	–
Случайность механики	Влияние случайности (%) $x = [0; 100]$	Отсутствует	Низкая	Высокая	Абсолютная

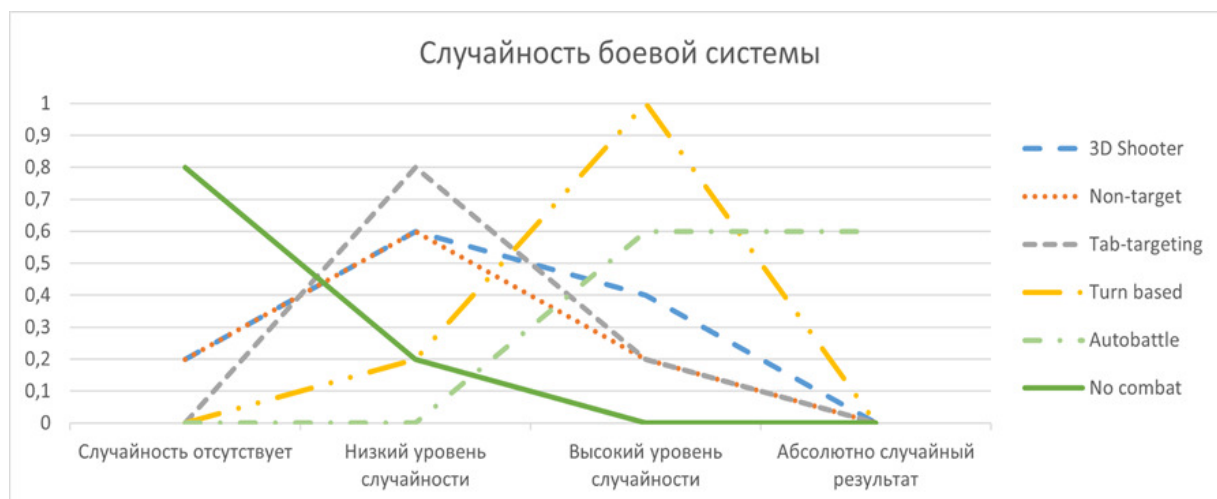


Рис. 3. Результат опроса на примере определения случайности боевой системы

Fig. 3. Survey results for randomness of combat system

Функции принадлежности (рис. 4, 5) всех термов также были получены на основе экспертной оценки пяти игровых дизайнеров. Для формирования графиков воспользуемся треугольной (первое уравнение) и трапециевидной (второе уравнение) видами функций принадлежности:

$$MF(x) = \begin{cases} 1 - \frac{b-x}{b-a}, & a \leq x \leq b; \\ 1 - \frac{x-b}{c-b}, & b \leq x \leq c; \\ 0, & \text{в других случаях,} \end{cases} \quad MF(x) = \begin{cases} 1 - \frac{b-x}{b-a}, & a \leq x \leq b; \\ 1, & b \leq x \leq c; \\ 1 - \frac{x-c}{d-c}, & c \leq x \leq d; \\ 0, & \text{в других случаях.} \end{cases}$$

Табл. 2. База правил

Table 2. Rule base

Игровая система	Игровая механика	Сложность	Динамичность	Агентивность игрока	Случайность
Боевая система	3D Shooter	Сложно	Оживленно	Высокая	Низкая
	Non-target	Комфортно	Оживленно	Ограничение	Низкая
	Tab-Targeting	Комфортно	Оживленно	Ограничение	Низкая
	Turn-based	Комфортно	Оживленно	Ограничение	Низкая
	Autobattle	Комфортно	Оживленно	Ограничение	Низкая
	No combat	Комфортно	Оживленно	Ограничение	Низкая
Головоломки	Object search	Комфортно	Монотонно	Ограничение	Низкая
	Match-three	Просто	Монотонно	Ограничение	Высокая
	Riddle	Комфортно	Умеренно	Отсутствует	Отсутствует
	Picklock	Комфортно	Умеренно	Ограничение	Низкая
	Jigsaw puzzle	Комфортно	Монотонно	Ограничение	Отсутствует
	No puzzle	Просто	Умеренно	Высокая	Отсутствует
Дополняющие механики	Card draw	Комфортно	Умеренно	Ограничение	Высокая
	Exploring	Просто	Умеренно	Высокая	Низкая
	Skill Tree	Комфортно	Умеренно	Ограничение	Низкая
	Parkour	Комфортно	Оживленно	Высокая	Отсутствует
	QTE	Просто	Умеренно	Ограничение	Отсутствует
	Dialog System	Просто	Монотонно	Ограничение	Низкая

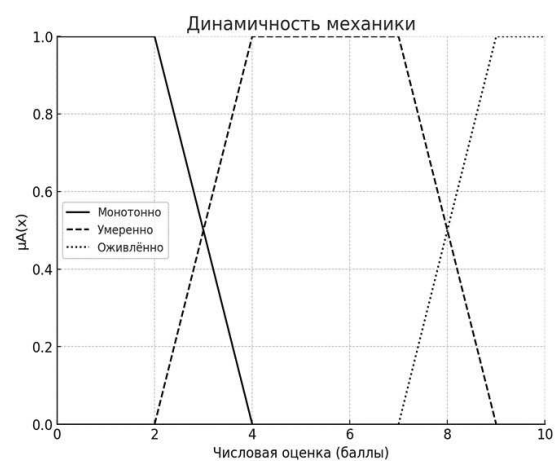
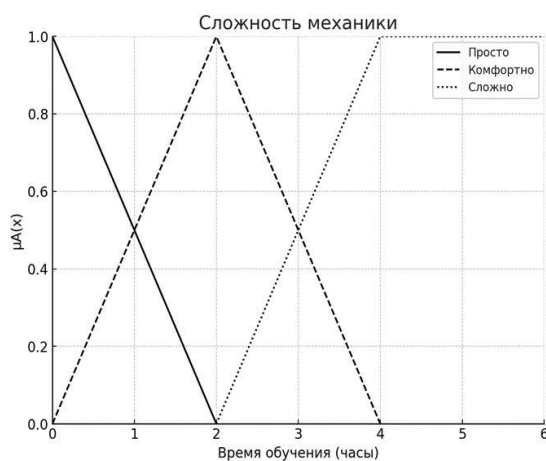


Рис. 4. Функции принадлежности термов «Сложность механики» (слева) и «Динамичность механики» (справа)

Fig. 4. Membership functions for the terms “mechanics complexity” (on the left) and “mechanics dynamics” (on the right)

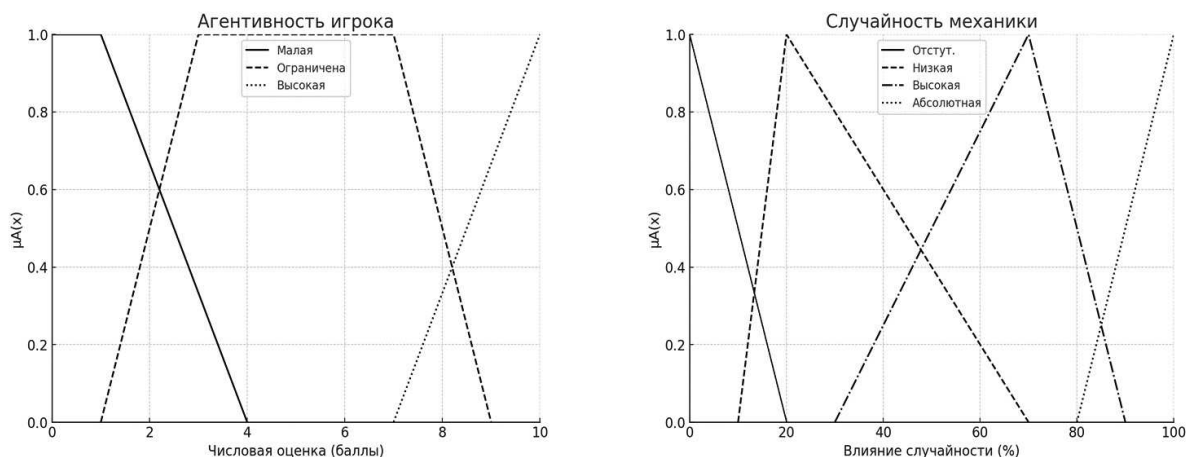


Рис. 5. Функции принадлежности термов «Агентивность игрока» (слева) и «Случайность механики» (справа)

Fig. 5. Membership functions for the terms “player agency” (on the left) and “mechanics randomness” (on the right)

Функция принадлежности для выходных переменных представляет собой распределение, равное между двумя термами, отражающими принадлежность к той или иной игровой механике. Таким образом, чем ближе выходное значение к единице, тем больше точность и релевантность предлагаемой механики относительно вводных параметров. Следовательно, можно использовать как целочисленное, так и нечеткое значение результата работы экспертной системы.

4. Обсуждение результатов

После настройки нечеткой функции принадлежности термов и правил для каждой игровой системы были получены три нечеткие системы с четырьмя входами, соответствующими характеристикам игровой механики, и шестью выходами системы (рис. 6), соответствующими конкретной механике из каждой игровой системы: боевая система, головоломки, дополняющие механики. Для удобства сделано так, что каждая нечеткая система не зависит друг от друга и имеет свой набор выходов и нечетких правил, но аналогичный набор термов для описания игровой механики. Далее происходит ввод значений, характеризующих игровую механику, для получения степени принадлежности данных значений с вариантами игровой системы. Результаты полученной работы системы отражены в табл. 3.

Таким образом, система на основе экспертной оценки может достаточно близко подбирать наиболее подходящие игровые механики на основе четырех критериев. Отметим, что полученные результаты имеют довольно низкую степень принадлежности к игровой механике, что говорит о неуверенности системы в данном решении. В то же время в некоторых случаях она полностью уверена в актуальности данной механики при всех вводных, что говорит о недочетах разработанной нечеткой системы. Кроме этого, в некоторых случаях система не может выбрать ни одну из предложенных механик и устанавливает степень принадлежности в одинаковое положение для всех игровых механик в игровой системе.

Конечно, при более тонкой настройке системы можно добиться повышения степени принадлежности: так, например, при использовании нечеткого вывода Сугено вместо

используемого метода Мамдани можно добиться повышения уверенности системы на 5–10 %, но это не решает проблему существенно. Предположительно, проблема с достаточно низкой уверенностью системы проявилась из-за недостаточного количества правил в нечеткой системе и недостатка данных. Для ее решения требуется более глубокий анализ и сбор большего объема экспертных знаний.

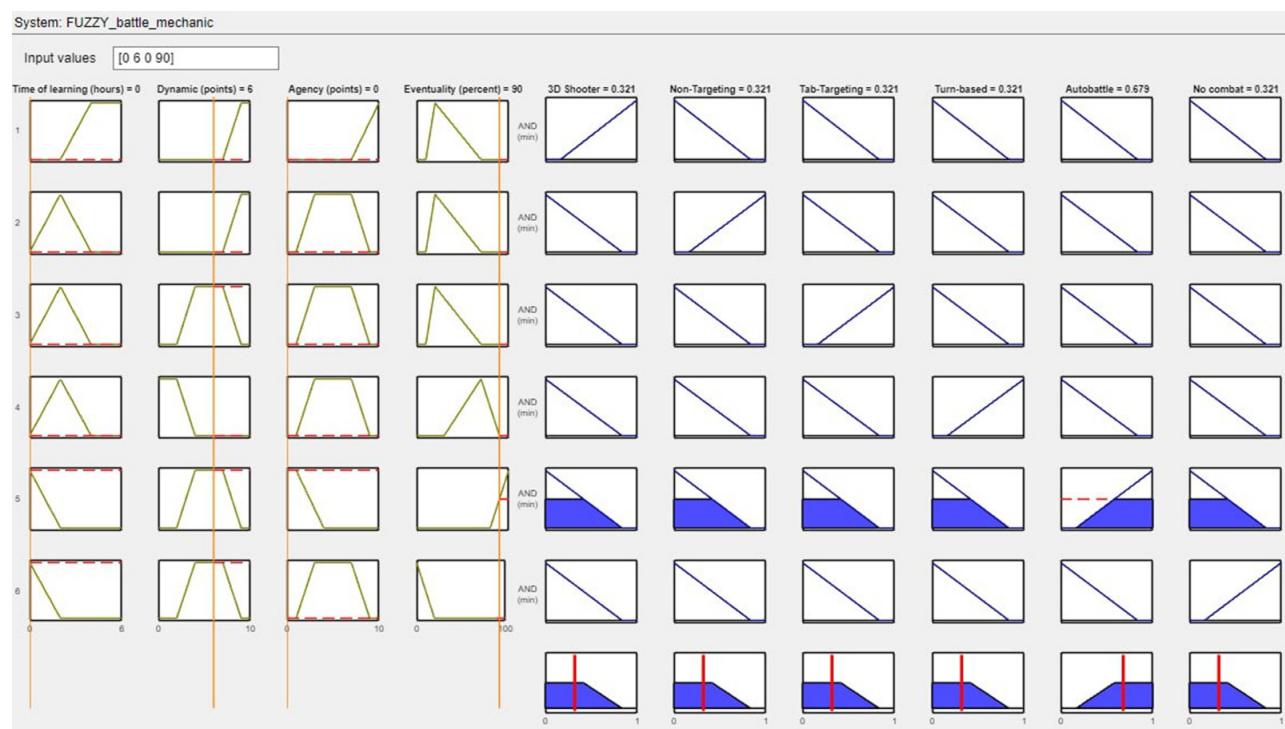


Рис. 6. Результаты работы системы принятия решения при вводе значений

Fig. 6. Output of the decision-making system based on the input values

Табл. 3. Результаты системы принятия решений на основе нечеткой логики

Table 3. Results of the decision-making system based on fuzzy logic

Сложность механики (часы)	Динамичность механики (баллы)	Агентивность игрока (баллы)	Случайность механики (%)	Боевая система (степень принадл.)	Головоломка (степень принадл.)	Доп. механика (степень принадл.)
2	3	5	60	Turn-based (0.602)	Object search (1)	Card draw (0.714)
3	5	4	30	Tab-targeting (0.679)	Любая (0.5)	Skill Tree (1)
5	8	8	50	3D Shooter (0.651)	Любая (0.5)	Любая (0.5)
0	6	4	15	No combat (0.635)	Picklock (0.67)	QTE (1)
0	6	0	90	Auto Battle (0.679)	Любая (0.5)	Любая (0.5)

Другая проблема рассматриваемой системы принятия решений заключается в том, что эта система затрагивает довольно ограниченный набор игровых механик, а их специфика не может быть описана при помощи лишь четырех критериев. Следовательно, несмотря на работоспособность теории, данная система требует доработки и расширения, что является трудной и наукоемкой задачей по определению наиболее существенных критериев, так как при наличии большого количества критериев минимально необходимый набор правил будет расти.

5. Выводы

Процесс работы геймдизайнера практически никак не изменился со времен зарождения игровой индустрии. При этом последняя шагнула далеко вперед, а комплексность и разнообразие видеоигр заметно возросли. В данных реалиях уследить за различными аспектами видеоигры и поддерживать высокое качество продукта очень сложно и требует большого опыта и времени. В связи с этим создание инструментов для поддержки работы игровых дизайнеров является актуальным.

В статье представлена система, использующая нечеткую логику для рекомендации механик видеоигр. Чтобы улучшить процесс выбора и реализации игровых механик, экспертные знания были интегрированы в экспертную систему. Основное внимание было уделено разработке системы, которая могла бы облегчить и оптимизировать этот процесс, делая его более эффективным и доступным для дизайнеров видеоигр. Система была направлена на упрощение сложных алгоритмов, чтобы обеспечить легкость в улучшении и адаптации к изменяющимся требованиям в дизайне видеоигр.

Предложенное решение может быть прототипом инструмента-ассистента для игровых дизайнеров, который в дальнейшем можно модифицировать использованием большого количества параметров для более тонкой настройки планируемых игровых механик, кроме того, необходимо значительно увеличить количество экспертов для увеличения точности экспертной оценки.

Полученные данные станут фундаментом для дальнейшего исследования, которое будет нацелено на определение взаимосвязей между игровыми механиками, нарративом и игровым опытом на основе ряда экспериментов. Определение такой взаимосвязи позволит проектировать игровой процесс более осмысленно и целесообразно, что в перспективе даст возможность ускорения процесса геймдизайна и автоматизации работы над корпусом игровых механик.

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Conflicts of Interest. The authors declare no conflicts of interest.

Литература

1. *Шараева Р.А., Кугуракова В.В., Селезнева Н.Э.* Методика упрощения таск-трекинга в проектах игровой индустрии // *Прогр. прод. и сист.* 2022. № 3. С. 374–383. <http://dx.doi.org/10.15827/0236-235X.139.374-383>.
2. *Шелл Дж.* Геймдизайн. Как создать игру, в которую будут играть все. М.: Альпина Паблшер, 2019. 640 с.
3. *Сахибгареева Г.Ф., Кугуракова В.В.* Редактор интерактивной структуры для инструмента генерации сценарных прототипов // *Электр. библ.* 2021. Т. 24, № 6. С. 1184–1202. <https://doi.org/10.26907/1562-5419-2021-24-6-1184-1202>.
4. *Козар Б.А., Кугуракова В.В., Сахибгареева Г.Ф.* Структуризация сущностей естественного текста с использованием нейронных сетей для генерации трехмерных сцен // *Прогр. прод. и сист.* 2022. № 3. С. 329–339.

5. *Sahibgareeva G.F., Kugurakova V.V.* Branched structure component for a video game scenario prototype generator // Proc. 23rd All-Russ. Sci. Conf. "Scientific Services and Internet", September 20–23, 2021. Moscow, 2021. P. 101–111. <https://doi.org/10.20948/abrau-2021-10-ceur>.
6. *Zadeh L.A.* Fuzzy logic = computing with words // IEEE Trans. Fuzzy Syst. 1996. V. 4, No 2. P. 103–111. <https://doi.org/10.1109/91.493904>.
7. *Mauer G.F.* A fuzzy logic controller for an ABS braking system // IEEE Trans. Fuzzy Syst. 1995. V. 3, No 4. P. 381–388. <https://doi.org/10.1109/91.481947>.
8. *Hungilo G.G., Emmanuel G., Maiga J., Santoso A.J.* Fuzzy logic control application: Design and simulation for washing machine // IOP Conf. Ser.: Mater. Sci. Eng. 2019. V. 662, No 2. Art. 022046. <https://dx.doi.org/10.1088/1757-899X/662/2/022046>.
9. *Raja K.* Python-based fuzzy logic in automatic washer control system // Soft Comput. 2023. V. 27. P. 6159–6185. <https://doi.org/10.1007/s00500-023-07979-3>.
10. *Ozsahin D.U., Uzun B., Ozshahin I., Mustapha M.T., Musa M.S.* Fuzzy logic in medicine // Biomedical Signal Processing and Artificial Intelligence in Healthcare / Ed. by W. Zgallai. Ser.: Developments in Biomedical Engineering and Bioelectronics. Acad. Press, 2020. P. 153–182. <http://dx.doi.org/10.1016/B978-0-12-818946-7.00006-8>.
11. *Millington I.* Artificial Intelligence for Games. The Morgan Kaufmann Ser. in Interactive 3D Technology / Ed. by D.H. Eberly. San Francisco, CA: Morgan Kaufmann Publ., 2006. xxxv, 856 p.
12. *Sweetser P., Wiles J.* Current AI in games: A review // Aust. J. Intell. Inf. Process. Syst. 2002. V. 8, No 1. P. 24–42.
13. *Calderon-Vilca H.D., Chavez N.M., Guimarey J.M.R.* Recommendation of videogames with fuzzy logic // Proc. 2020 27th Conf. of Open Innovations Association (FRUCT). Trento, 2020. P. 27–37. <https://doi.org/10.23919/FRUCT49677.2020.9211082>.
14. *Papadimitriou S., Chrysafiadi K., Virvou. M.* FuzzEG: Fuzzy logic for adaptive scenarios in an educational adventure game // Multimedia Tools Appl. 2019. V. 78, No 22. P. 32023–32053. <https://doi.org/10.1007/s11042-019-07955-w>.
15. *de Oliveira F.M., Lanzillotti R.S., da Costa R.M.E.M., Gonçalves R., Ventura P., de Carvalho L.A.V.* ARVET and SAPTEPT: A virtual environment and a system supported by fuzzy logic in virtual reality exposure therapy for PTSD patients // Proc. 2012 12th Int. Conf. on Computational Science and Its Applications. Salvador, 2012. P. 103–107. <https://doi.org/10.1109/ICCSA.2012.26>.
16. *Pirovano M.* The use of fuzzy logic for artificial intelligence in games. Milano, Univ. of Milano, Politecnico di Milano, 2012.
17. *Григорьева Д.Р., Гареева Г.А., Басыров Р.Р.* Основы нечеткой логики. Учебно-методическое пособие к практическим занятиям и лабораторным работам. Набережные Челны: Изд-во НЧИ КФУ, 2018. 42 с.
18. *Chuang K.-C., Lan T.-S., Zhang L.-P., Chen Y.-M., Dai X.-J.* Parameter optimization for computer numerical controlled machining using fuzzy and game theory // Symmetry. 2019. V. 11, No 12. Art. 1450. <https://doi.org/10.3390/sym11121450>.

References

1. Sharaeva R.A., Kugurakova V.V., Selezneva N.E. A method for simplifying task tracking in video game development projects. *Software Syst.*, 2022, no. 3, pp. 374–383. <http://dx.doi.org/10.15827/0236-235X.139.374-383>. (In Russian)
2. Schell J. Geimdzain. *Kak sozdat' igr, v kotoruyu budut igrat' vse* [The Art of Game Design: A Book of Lenses]. Moscow, Alpina Publ., 2019. 640 p. (In Russian)
3. Sahibgareeva G.F., Kugurakova V.V. Interactive structure editor for scenario prototyping tool. *Elektron. Bibl.*, 2021, vol. 24, no. 6, pp. 1184–1202. <https://doi.org/10.26907/1562-5419-2021-24-6-1184-1202>. (In Russian)
4. Kozar B.A., Kugurakova V.V., Sahibgareeva G.F. Structuring natural text entities using neural networks for 3D scene generation. *Program. Prod. Sist.*, 2022, no. 3, pp. 329–339. (In Russian)
5. Sahibgareeva G.F., Kugurakova V.V. Branched structure component for a video game scenario prototype generator. *Proc. 23rd All-Russ. Sci. Conf. "Scientific Services and Internet", September 20–23, 2021*. Moscow, 2021, pp. 101–111. <https://doi.org/10.20948/abrau-2021-10-ceur>.
6. Zadeh L.A. Fuzzy logic = computing with words. *IEEE Trans. Fuzzy Syst.*, 1996, vol. 4, no. 2, pp. 103–111. <https://doi.org/10.1109/91.493904>.
7. Mauer G.F. A fuzzy logic controller for an ABS braking system. *IEEE Trans. Fuzzy Syst.*, 1995, vol. 3, no. 4, pp. 381–388. <https://doi.org/10.1109/91.481947>.
8. Hungilo G.G., Emmanuel G., Maiga J., Santoso A.J. Fuzzy logic control application: Design and simulation for washing machine. *IOP Conf. Ser.: Mater. Sci. Eng.*, 2019, vol. 662, no. 2, art. 022046. <https://dx.doi.org/10.1088/1757-899X/662/2/022046>.
9. Raja K. Python-based fuzzy logic in automatic washer control system. *Soft Comput.*, 2023, vol. 27, pp. 6159–6185. <https://doi.org/10.1007/s00500-023-07979-3>.
10. Ozsahin D.U., Uzun B., Ozsahin I., Mustapha M.T., Musa M.S. Fuzzy logic in medicine. In: Zgallai W. (Ed.) *Biomedical Signal Processing and Artificial Intelligence in Healthcare*. Ser.: Developments in Biomedical Engineering and Bioelectronics. Acad. Press, 2020, pp. 153–182. <http://dx.doi.org/10.1016/B978-0-12-818946-7.00006-8>.
11. Millington I. *Artificial Intelligence for Games*. The Morgan Kaufmann Ser. in Interactive 3D Technology. Eberly D.H. (Ed.). San Francisco, CA, Morgan Kaufmann Publ., 2006. xxxv, 856 p.
12. Sweetser P., Wiles J. Current AI in games: A review. *Aust. J. Intell. Inf. Process. Syst.*, 2002, vol. 8, no. 1, pp. 24–42.
13. Calderon-Vilca H.D., Chavez N.M., Guimarey J.M.R. Recommendation of videogames with fuzzy logic. *Proc. 2020 27th Conf. of Open Innovations Association (FRUCT)*. Trento, 2020, pp. 27–37. <https://doi.org/10.23919/FRUCT49677.2020.9211082>.
14. Papadimitriou S., Chrysafiadi K., Virvou. M. FuzzEG: Fuzzy logic for adaptive scenarios in an educational adventure game. *Multimedia Tools Appl.*, 2019, vol. 78, no. 22, pp. 32023–32053. <https://doi.org/10.1007/s11042-019-07955-w>.
15. de Oliveira F.M., Lanzillotti R.S., da Costa R.M.E.M., Gonçalves R., Ventura P., de Carvalho L.A.V. ARVET and SAPTEPT: A virtual environment and a system supported by fuzzy logic in virtual reality exposure therapy for PTSD patients. *Proc. 2012 12th Int. Conf.*

on Computational Science and Its Applications. Salvador, 2012, pp. 103–107.
<https://doi.org/10.1109/ICCSA.2012.26>.

16. Pirovano M. The use of fuzzy logic for artificial intelligence in games. Milano, Univ. of Milano, Politecnico di Milano, 2012.
17. Grigor'eva D.R., Gareeva G.A., Basyrov R.R. *Osnovy nechetkoi logiki. Uchebno-metodicheskoe posobie k prakticheskim zanyatiyam i laboratornym rabotam* [Fundamentals of Fuzzy Logic: Educational and Methodical Manual for Practical Classes and Laboratory Work]. Naberezhnye Chelny, Izd. NChI KFU, 2018. 42 p. (In Russian)
18. Chuang K.-C., Lan T.-S., Zhang L.-P., Chen Y.-M., Dai X.-J. Parameter optimization for computer numerical controlled machining using fuzzy and game theory. *Symmetry*, 2019, vol. 11, no. 12, art. 1450. <https://doi.org/10.3390/sym11121450>.

Информация об авторах

Алексей Витальевич Шубин, ассистент, младший научный сотрудник Института информационных технологий и интеллектуальных систем, Казанский (Приволжский) федеральный университет

E-mail: shubin.aleksey.kpfu@gmail.com

ORCID: <https://orcid.org/0000-0002-6203-3268>

Author Information

Aleksey V. Shubin, Teaching Assistant, Junior Researcher, Institute of Information Technology and Intelligent Systems, Kazan Federal University

E-mail: shubin.aleksey.kpfu@gmail.com

ORCID: <https://orcid.org/0000-0002-6203-3268>

Поступила в редакцию 10.12.2024

Принята к публикации 31.12.2024

Received December 10, 2024

Accepted December 31, 2024