УДК 519.876.5+28.23.29

£ 10.25209/2079-3316-2022-13-3-255-273



Практики балансирования компьютерных игр

Гульнара Фаритовна $\mathbf{CaxuбrapeeBa^{1\boxtimes}}$, Влада Владимировна $\mathbf{KyrypakoBa^2}$ Казанский (Приволжский) федеральный университет, Казань, Россия $\mathbf{E^{\boxtimes}}$ gulnara.sahibgareeva42@gmail.com (подробнее об авторах на с. 269)

Аннотация. Предметом исследования является сфера разработки игровых и интерактивных проектов. В частности, разработка системы игрового баланса для них. В ходе изучения данной темы учтены актуальные тренды, сформированные как в научной, так и в коммерческой сферах. Итоговые выводы и видение дальнейшей проектной работы сформированы благодаря методам моделирования и эксперимента.

Результаты работы — два инструмента для создания и автоматического редактирования игрового баланса. Они являются частью плана разработки игрового инструмента для прототипирования компьютерных игр, сокращающего время и ресурсы на разработку за счет автоматического создания контента. (see abstract in English on p. 270)

Ключевые слова и фразы: игровой баланс, компьютерные игры, экономическая симуляция, игровой дизайн, инструменты разработки

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

Для цитирования: Сахибгареева $\Gamma.\Phi.$, Кугуракова В.В. *Практики* балансирования компьютерных игр // Программные системы: теория и приложения. 2022. **Т. 13**. № 3(54). С. 255–273. http://psta.psiras.ru/read/psta2022_3_255-273.pdf



Введение

На основании анализа известных определений игрового баланса можно прийти к выводу, что в широком смысле под ним подразумевается баланс в значении гармоничности, целостности и логичности игрового процесса, в том числе с учетом игрового повествования, дизайна уровней, аудиосопровождения и даже дизайна пользовательского интерфейса [1]. Коротко—это баланс дизайнерских решений. В узком же смысле игровой баланс—это то, чему чаще всего обучают игровых дизайнеров с прикладной точки зрения, это математический баланс всех функций и переменных, т.е. экономической системы компьютерной игры (см., например, сайт Game Balance Concept (http://gamebalanceconcepts.wordpress.com/) или сайт с обучающими курсами для разработчиков игр (https://www.devtodev.com/education/online-course/mathematics-in-game-design)).

Приведение игрового проекта в баланс является одной из актуальных и нетривиальных задач, которая стоит на стыке дизайна и математики. Ежегодно в этой области появляется все больше работ, упоминающих прямую зависимость математических функций, регулирующих игру, с контекстом, создаваемым благодаря компьютерной графике, музыке, игровому сценарию и доступным игровым механикам [2,3]. Тонкая контекстуальная настройка игрового процесса на целевую аудиторию— это субъективная оценка, отражающая суть авторского замысла и его зависимости от игроков [4,5].

Для решения задач математического баланса применяются методы комбинаторики, теории вероятностей, математической статистики и теория игр. [1,6–8].

Оценить и прочувствовать игровой баланс помогают, в свою очередь, опытность и критический взгляд на дизайн концепции — подобный тезис указан, например, в серии обучающих статей на сайте Sirlin on game design (https://www.sirlin.net/articles/balancing-multiplayer-games-part-1-definitions).

Настоящая работа посвящена анализу эффективности авторских инструментов и подходов для работы над игровым балансом игр. Для этого предпринята попытка создания двух инструментов для работы с математическим балансом.

 $^{^{1}}$ Игровая механика— это правило, которое дает игроку возможность совершать действия, которые определенным образом воздействуют на игру.

Основной результат заключается в разработке и оценке эффективности двух инструментов, раскрывающих проблематику работы над игровым балансом.

С нашей точки зрения, проведенное исследование интересно в рамках более объемной разработки авторов, касающейся автоматизации в сфере разработки игр. Короткое название сазданного инструмента – генератор игровых прототипов на основе текста на естественном языке. Функционал этого инструмента заключается в извлечении информации из текстовой документации и дальнейшей генерации различных артефактов разработки.

В первом разделе представлена симуляция, которая выводит статистические данные для оценки сбалансированности текущей системы. Во втором разделе описана модификация онлайн-инструмента Machination², доказывающая эффективность использования подходов этого инструмента, которая в разы сокращает работу над математическим балансом. В третьем разделе приведена краткая информация о генераторе игровых прототипов и описано видение того, как в общую картину встраивается функционал для работы с игровым балансом. В заключении представлены выводы и планы дальнейшего развития.

Симуляция игровой экономики

Для того, чтобы представить, как выглядит работа над экономической системой, рассмотрим студенческую 3 реализацию симуляции MMORPG 4 с упором на торговые отношения между персонажами. Для оценки сбалансированности системы здесь введена функция отображения текущих значений ряда параметров.

По существу симуляция представляет собой модель взаимодействия агентов в рамках товарообмена.

 $^{^2{\}rm Machination}$ — визуальный онлайн редактор динамических диаграмм игрового баланса (https://machinations.io/).

 $^{^3 \}mbox{Примаченко A. M. Проектирование и разработка игровой экономики / Выпускная квалификационная работа // Казанский федеральный университет, <math display="inline">2022.-48~\rm c.$

⁴MMORPG (англ., сокр. Massively multiplayer online role-playing game) – массовая многопользовательская ролевая онлайн-игра, игровой жанр.

Реализацией является проект в игровом движке Unity 5 с применением информационно-ориентированного стека технологий DOTS 6 для высокопроизводительных многопоточных вычислений.

Базовым в направлении игрового балансирования является тезис о том, что игра—это упрощенная модель реальности. Для создания достоверной экономической системы используются следующие элементы: ресурсы, хранилища, источники/стоки, конвертеры и торговцы [9].

Замкнутость игровой экономики зависит от того добывают ли ресурсы $\mathrm{NPC^7}$ или персонажи или товары появляются без причины, из надобности для игрового процесса. Важно и то, преобразовывают ли ресурсы, или же они исчезают без учета смыслового контекста. Допущения приводят к возрастанию степени нереалистичности игры. Игровые дизайнеры могут умышленно регулировать реалистичность и замкнутость в зависимости от идеи концепции.

Специфическим ресурсом в экономике является валюта, которая позволяет отказаться от бартера в пользу отложенной покупки какого-либо товара по требованию и при наличии. Также валюта определяет относительную ценность того или иного ресурса в унифицированной величине.

Для объектов, из которых состоит система, были выбраны следующие именования: юниты и здания.

Юниты— это агенты, т. е. персонажи, которые моделируют поведение людей. Их главное стремление— восполнять чувство сытости. Юнит характеризуется значениями следующих параметров: скорость передвижения и метаболизма, специализация, сытость и статус голода. Приоритет в поведении юнита по убыванию: искать еду, если голоден; работать, если устроен на работу и идет смена; находиться дома.

Здания—это место, где юниты работают и совершают различные действия: создают, преобразуют, покупают и продают товары за счет рабочей силы. За работу здание выплачивает юнитам зарплату. Сеть зданий поддерживает систему переработки и циркуляции товаров.

 $^{^5}$ Unity—игровой движок для работы над интерактивными проектами с компьютерной графикой на языке программирования C# (http://unity.com).

 $^{^6 \}rm Unity \ DOTS-$ безопасная среда для разработки многопоточного кода на С# для Unity (https://unity.com/ru/dots).

 $^{^{7}\}mathrm{NPC}$ (англ., сокр. Non-Player Character) – неигровой персонаж, которым управляет ИИ-алгоритм.

В конце концов благодаря им распространяется еда, которую покупают юниты.

Здания разделены на типы: торговая точка, производственная точка, добывающая точка. У каждого здания есть вакансии, которые определяют рабочего юнита, график работы, задачи и объем зарплаты. У каждых юнита и здания есть хранилище ресурсов.

Экономическая система, реализованная в рамках данной формализации, замкнута: все ресурсы появляются, перемещаются, преобразуются и уничтожаются по правилам внутриигрового мира, т.е. операции обоснованы и связаны между собой.

Во время работы программа показывает значение ряда переменных, которые образуют статистику проекта (рисунок 1). Например, это такие данные, как среднее значение сытости юнита, объема производства зданий одного типа, добычи каких-либо ресурсов.

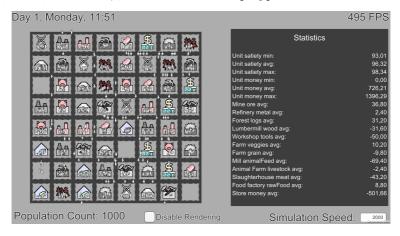


Рисунок 1. Симуляция на 1000 юнитов

В качестве эксперимента нами были сгенерированы экономические системы с одними и теми же типами зданий и пропорциональным количеством юнитов и зданий. Эффективность разработанного алгоритма иллюстрирует факт того, что при наличии одиннадцати зданий каждого типа симуляция поведения ста тысяч юнитов происходила с частотой в 45 кадров в секунду.

Приложение имеет следующий функционал: создание и запуск

симуляции, просмотр и настройка статистики для оценки сбалансированности и производительности.

Приложение, а также полученная в нем экономическая система могут быть интегрированы в игру жанра MMORPG. Функционал приложения может быть улучшен и расширен за счет добавления большего количества методов оптимизации, настраиваемых параметров и интеграции спроектированной системы заданий.

Автоматическое балансирование

Реализация экономической системы, представленной выше, осуществляется за счет программирования, а оценка сбалансированности и корректирование значений переменных достигается с помощью многократного тестирования.

Однако тестирование занимает много времени. В противовес ручному тестированию распространены алгоритмы автоматического тестирования, в том числе с применением методов машинного обучения [10-13].

В следующем примере 8 инструмента для реализации функции автоматического балансирования настроена связь с онлайн-инструментом Machination. Сделано это было для того, чтобы опустить реализацию функций визуализации динамического игрового баланса и запуска симуляции экономической системы.

Machinations—это визуальный онлайн-редактор динамического математического баланса игры. Инструмент позволяет моделировать и симулировать экономику игрового проекта благодаря динамическим диаграммам, которые показывают изменения системы в режиме реального времени.

Для эксперимента в Machination создана диаграмма баланса механики кровотечения из настольной игры Dungeons and Dragons (https://dnd.wizards.com/).

Механика состоит из урона от кровотечения, шанса ослабления или остановки кровотечения, урона от атак противника и шанса того, что атака противника попадет по игроку.

 $^{^8}$ Галимзянов Г. Р. Разработка инструмента автоматической корректировки внутриигровых параметров / Выпускная квалификационная работа // Казанский федеральный университет, 2021. – 35 с.

Сбалансированность механики оценивается по тому, сколько ходов сможет выстоять игрок, если он получает дополнительный урон из-за кровотечения. Система балансирует параметры относительно того, какое количество ходов игровой разработчик посчитает необходимым.

Рассмотрим работу компонента, непосредственно отвечающего за то, чтобы передавать информацию, сколько урона получает игрок от кровотечения (рисунок 2).

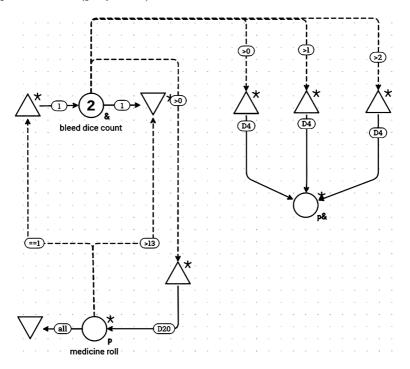


Рисунок 2. Диаграмма баланса механики кровотечения

Пул «medicine roll» проверяет медицинские навыки персонажа с кровотечением. Если проверка покажет наличие таких навыков, то кровотечение уменьшится. В противном случае, кровотечение усилится. Пул «bleed dice count» отвечает за то, сколько конкретно крови потеряет игрок.

Итоговая диаграмма содержит в себе также компоненты, которые вычисляют значения атаки и урона (рисунок 3).

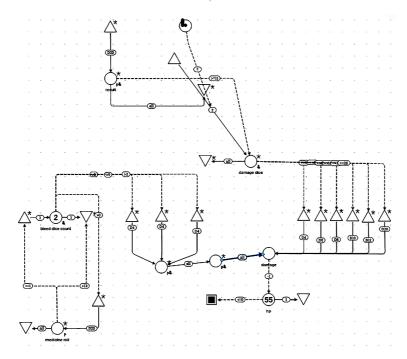


Рисунок 3. Итоговая диаграмма механики кровотечения

Конкретные численные переменные, использованные в эксперименте, следующие: уровень здоровья игрока-44, ожидаемая продолжительность боя-6-10 ходов.

В результате автоматического балансирования приложение вывело оптимальные значения остальных параметров: уровень кровотечения — 2, количество граней кости урона — 12.

Для подтверждения оптимальности сгенерированных значений проведена проверка. Например, в описанном выше эксперименте в результате 40 запусков диаграммы баланса средняя длительность боев составила семь. Эксперимент показал, что автоматический баланс значительно сокращает время на разработку: генерация и проверка полученных параметров заняла чуть больше трех часов. Ручная же работа может затянуться на дни и недели.

Интеграция в генератор игровых прототипов

Реализованные инструменты для работы с игровым балансом являются частью исследования в области разработки программного обеспечения для игровых дизайнеров и сценаристов, а именно, генератора игровых прототипов на основе текстовых документов, написанных на естественном языке.

Инструмент для реализации игрового баланса показал, каким образом создается игровая экономика на примере MMORPG-симулятора. Его логичным развитием станет составление перечня параметров, являющихся общими для игр схожего жанра.

Инструмент для автоматического балансирования является эволюционным развитием предыдущего и позволяет не только отслеживать состояние системы в каждый момент времени, но и автоматически изменять значения до необходимых пользователю. В перспективе алгоритм может быть расширен для решения задачи игрового баланса, который адаптируется под конкретного игрока [14,15].

Генератор игровых прототипов [16]—это планируемый результат обширного исследования, целью которого является разработка инструмента для игровых разработчиков и сценаристов, который призван ускорить этап прототипирования. Схема генератора отображена в виде взаимосвязи нескольких модулей (рисунок 4).

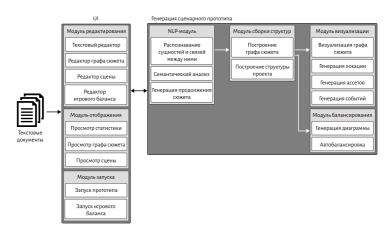


Рисунок 4. Схема инструмента генерации сценарного прототипа

Кратко опишем, каким образом складывается последовательность шагов обработки в генераторе: комплект текстовых документов анализируется; полученная статистическая и функциональная информация визуализируется для удобства пользователя; из полученных данных строится структура проекта; по структуре собирается схема, подбираются трехмерные модели, генерируются события, а также диаграмма баланса.

Для генерации визуализации игровых механик и баланса необходима избыточная информация. Кроме этого могут быть использованы методы машинного обучения [17,18]. Для этого необходимо разметить достаточный для обучения объем материалов, основываясь на котором, можно сделать однозначный вывод о характере генерации.

Попытки использования нейросетей в рамках инструмента были предприняты нами в работе с компьютерной графикой: на основе размеченных данных и текста на естественном языке были сгенерированы вариации трехмерной сцены, соответствующие текстовому описанию [19].

Однако наибольшей проблемой является составление корпуса размеченных документов, по которым программа могла бы генерировать игровые проекты надлежащего качества. Для создания больших массивов данных по требованию уместно применение технологий синтетических данных. Так, совместно с одним из авторов статьи был задокументирован подход к генерации трехмерных синтетических данных для обучения алгоритмов анализа видео с камер наблюдения [20].

Наибольшую сложность составляет работа над игровым балансом в контексте повествования. В данном направлении также планируется развитие концепции, которая учитывает тот факт, что изменение математических параметров зависит от того, какие события происходят по сюжету с персонажем игрока. В одной из работ⁹ были предприняты попытки автоматической генерации игровых механик в соответствии с нарративом, которые позволят продвинуть решение данной проблемы в будущем.

На основе полученных результатов запланированы следующие этапы работы по развитию функционала, связанного с балансированием игрового прототипа:

 $^{^9 \}mbox{Липатов В. В.}$ Автоматическая генерация игровых механик в соответствии с нарративом / Выпускная квалификационная работа // Казанский федеральный университет, 2019.-78 с.

- разметка технических документов, описывающих игровой проект, его сценарий, визуализацию и т. д.;
- генерация технических документов с применением технологии синтетических данных;
- обучение нейросети генерации запроса для генерации динамической диаграммы игрового баланса для Machination.

Полученный результат может нуждаться в корректировке.

Функции генерации и воспроизведения динамической диаграммы повторяют возможности инструмента Machination. Чтобы ускорить работу по данному направлению и не дублировать уже существующие возможности, логично настроить стабильную связь с Machination через ${\rm API}^{10}$. Это означает, что часть возможностей будет доступна во внешнем редакторе.

Концепция итогового приложения с точки зрения работы над игровым балансом имеет следующие функции:

- прием текстовых документов;
- анализ текста, извлечение данных для построения диаграммы баланса;
- формирования и отправка запроса в Machination;
- запуск страницы с сгенерированной Machination диаграммой в браузере;
- редактирование диаграммы в Machination;
- \bullet экспорт экономической системы в игровой движок $^{11},$ например, Unity.

Доступ к функциям Machination сократит время на разработку итогового инструмента.

Заключение

Определения игрового баланса касаются целого спектра задач игровой разработки. В рамках данной статьи предложено разделить широкое понимание данного термина как исключительно дизайнерского

 $^{^{10}{}m API}$ (англ., сокр. Application Programming Interface) — «программный интерфейс приложения») — описание способов, взаимодействия одной компьютерной программы с другими.

 $^{^{11}}$ Игровой движок (калька с англ. Game Engine) – общеупотребительный термин для систем разработки интерактивных приложений.

взгляда на игру и узкое понимание как математической модели внутриигровой экономики.

Инструмент автоматического создания игровых прототипов требует функции балансирования игровой экономики. Серия работ [16,21–23], раскрывающих полный спектр зависимостей технической документации и реализации, а также пытающихся наладить прочную связь между этими артефактами разработки, посвящена разработке авторского инструмента генерации игровых прототипов из текста на естественном языке. Соответственно, работа над игровым балансом должна осуществляеться на основе результата, полученного текстовым анализатором.

Амбициозность идеи о том, что игровые прототипы можно генерировать из текстовой документации, уравновешивается практическими результатами, полученными в области генерации трехмерной графики: уже сейчас автоматически можно настраивать камеру [24], расставлять предметы в пространстве [24], собирать сцены [19]. Кроме того, были предприняты попытки работы над разветвленной структурой сценария [25,26] и визуализацией информации, полученной с помощью анализа текста [27].

Выше описаны разработанные авторами инструменты, обеспечивающие работу над игровым балансом на двух этапах: разработка экономической системы и ее автоматическое балансирование.

По результатам анализа собраны и задокументированы требования к компоненту игрового баланса инструмента генерации игровых прототипов. Идеи, возникшие в результате анализа описанных инструментов для работы с игровым балансом, будут интегрированы в конвейер разрабатываемого генератора.

Список литературы

- [1] Rouse R. Game Design: Theory and Practice, Wordware Game Developer's Library, 2nd ed.— Jones & Bartlett Learning.—2004.— ISBN 978-1556229121.—704 pp. \uparrow 256
- [2] Pfau J., Liapis A., Yannakakis G. N., Malaka R. Dungeons & replicants II: Automated game balancing across multiple difficulty dimensions via deep player behavior modeling // IEEE Transactions on Games.—2022 (Early Access).
- [3] Dworak W., Filgueiras E., Valente J. Automatic emotional balancing in game design: Use of emotional response to increase player immersion, HCII 2020: Design, User Experience, and Usability. Design for Contemporary

- Interactive Environments (July 19–24, 2020, Copenhagen, Denmark), Lecture Notes in Computer Science.—vol. **12201**, Cham: Springer.— 2020.— ISBN 978-3-030-49760-6.—pp. 426–438. €↑↑256
- [4] Sylvester T. Designing games: A guide to engineering experiences, 1st ed..—Sebastopol: O'Reilly Media.—2013.—ISBN 978-1449337933.—413 pp. ↑256
- [5] Schell J. The Art of Game Design: A Book of Lenses.— Boca Raton: A K Peters/CRC Press.— 2015.— ISBN 9781466598645.—600 pp. \uparrow 256
- [7] Novak J. Game Development Essentials: An Introduction, 3rd ed.—Melbourne: Cengage Learning.—2011.— ISBN 978-1111307653.—510 pp. \uparrow 256
- [8] Rollings A., Adams E. Andrew Rollings and Ernest Adams on Game Design, 1st ed.– Indianapolis: New Riders Publishing.– 2003.– ISBN 978-1592730018.– 400 pp. ↑256
- [9] Hill G. C. E. Rethinking Economy-Building Video Games: How might designers inspire new economic models through video game mechanics? Submitted to OCAD University in partial fulfillment of the requirements for the degree of Master of Design in Strategic Foresight and Innovation.— Toronto: OCAD University.—2021.—79 pp. (PR) ↑258
- [10] Radomski S., Neubacher T. Formal verification of selected gamelogic specifications // Proceedings of the 2nd EICS Workshop on Engineering Interactive Computer Systems with SCXML (June 23, 2015, Duisburg, Germany).—2015.—pp. 30–34. 60% (R) †260
- [11] Varvaressos S., Lavoie K., Gaboury S., Halle S. Automated bug finding in video games: A case study for runtime monitoring // Computers in Entertainment.— 2017.— Vol. 15.— No. 1.— 1.— 10 pp. 60 ↑260
- [12] Pfau J., Smeddinck J. D., Malaka R. Automated game testing with ICARUS: Intelligent completion of adventure riddles via unsupervised solving // CHI PLAY '17 Extended Abstracts: Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play (15−18 October 2017, Amsterdam, The Netherlands), New York: ACM.− 2017.− ISBN 978-1-4503-5111-9.− pp. 153−164.
- [14] Andrade G., Ramalho G., Gomes A. S., Corruble V. Dynamic game balancing: An evaluation of user satisfaction, Second Artificial Intelligence and Interactive Digital Entertainment Conference // Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment.—2006.—Vol. 2.—No. 1.—pp. 3–8. (RL) ↑263

- [15] Tijs T. J. V., Brokken D., IJsselsteijn W. A. Dynamic game balancing by recognizing affect, Fun and Games 2008: Fun and Games, Lecture Notes in Computer Science. – vol. 5294, Berlin–Heidelberg: Springer. – 2008. – ISBN 978-3-540-88321-0. – pp. 89–93. €↑ ↑263
- [16] Сахибгареева Г. Ф., Кугуракова В. В. Концепт инструмента автоматического создания сценарного прототипа компьютерной игры // Электронные библиотеки. – 2018. – Т. 21. – № 3–4. – с. 235—249.
 ★ ↑263, 266
- [17] Li Z., Wang X., Yang W., Wu J., Zhang Z., Liu Z., Sun M., Zhang H., Liu S. A unified understanding of deep NLP models for text classification // IEEE Transactions on Visualization and Computer Graphics. 2022 (Early Access). 14 pp. €0 ↑264
- [18] Packowski S., Switzer W. Extracting meaning from text and creating a custom language model to optimize NLP results: NLP hands-on workshop series // CASCON '19: Proceedings of the 29th Annual International Conference on Computer Science and Software Engineering (4-6 November 2019, Toronto, Ontario, Canada), Riverton: IBM Corp. 2019. pp. 382-383. ↑264
- [19] Козар Б. А., Кугуракова В. В., Сахибгареева Г. Ф. *Модификация* подходов нейронных сетей со структуризацией сущностей для генерации трехмерных локаций на основе обработки текста // Программные продукты и системы. 2022. № 3 (в печати). ↑264, 266
- [20] Кугуракова В. В., Абрамов В. Д., Костюк Д. И., Шараева Р. А., Газизов Р. Р., Хафизов М. Р. Генерация трехмерных синтетических датасетов // Электронные библиотеки. 2021. Т. **24**. № 4. с. 622–652. *
- [21] Сахибгареева Г.Ф., Бедрин О.А., Кугуракова В.В. *Разработка компонента генерации визуализации сценарного прототипа видеоигр // Научный сервис в сети Интернет: труды XXII Всероссийской научной конференции* (21–25 сентября 2020 г., онлайн), М.: ИПМ им. М. В. Келдыша.—2020.— ISBN 978-5-98354-058-3.— с. 581–603. €0↑266
- [22] Сахибгареева Г. Ф., Бедрин О. А., Кугуракова В. В. *Раскадровка как одно из представлений сценарного прототипа компьютерных игр* // Электронные библиотеки.— 2021.— Т. **24**.— № 2.— с. 408–444. ★ む↑266
- [23] Sahibgareeva G. F., Bedrin O. A., Kugurakova V. V. Visualization component for the scenario prototype generator as a video game development tool // Proceedings of the 22nd Conference on Scientific Services & Internet, CEUR Workshop Proceedings. vol. 2784. 2020. pp. 267–282. ♠ ↑266
- [24] Кугуракова В. В., Сахибгареева Г. Ф., Нгуен А.З., Астафьев А.М. Пространственная ориентация объектов на основе обработки текстов на естественном языке для генерации раскадровок // Электронные библиотеки.— 2020.— Т. 23.— с. 1213—1238. ★ வ↑266
- [25] Сахибгареева Г. Ф., Кугуракова В. В. Прототипирование вариативности сюжета компьютерных игр // Научный сервис в сети Интернет: труды

XXIII Всероссийской научной конференции (20–23 сентября 2021 г., онлайн), М.: ИПМ им. М. В. Келдыша. – 2021. – ISBN 978-5-98354-062-0. – с. 347–360. \bigcirc \uparrow 266

[26] Sahibgareeva G. F., Kugurakova V. V. Branched structure component for a video game scenario prototype generator // Proceedings of the 23nd Conference on Scientific Services & Internet, CEUR Workshop Proceedings.—vol. 3066.— 2021.—pp. 101–111. Rp. \(\frac{1}{266}\)

 Поступила в редакцию
 28.06.2022;

 одобрена после рецензирования
 12.08.2022;

 принята к публикации
 10.09.2022.

Рекомендовал к публикации

 ∂ .ф.-м.н. А. М. Елизаров

Информация об авторах:



Гульнара Фаритовна Сахибгареева

ассистент кафедры программной инженерии Института ИТИС КФУ. Сфера научных интересов – игровая сценаристика, нарративный дизайн, изучение вопроса эффективности создания игрового прототипа и возможности автоматизации данного процесса.

(D)

0000-0003-4673-3253

e-mail: gulnara.sahibgareeva42@gmail.com



Влада Владимировна Кугуракова

к.т.н., руководитель лаборатории разработки интеллектуальных инструментов для компьютерных игр Института информационных технологий и интеллектуальных систем

(D

0000-0002-1552-4910

e-mail: vlada.kugurakova@gmail.com

Все авторы сделали эквивалентный вклад в подготовку публикации. Авторы заявляют об отсутствии конфликта интересов. UDC 519.876.5+28.23.29



10.25209/2079-3316-2022-13-3-255-273

Game Balance Practices

Gulnara Faridovna Sahibgareeva¹, Vlada Vladimirovna Kugurakova²

Kazan Federal University, Kazan, Russia

 $^{1\! extsf{Z}}$ gulnara.sahibgareeva42@gmail.com (learn more about the authors in Russian on p. 269)

Abstract. The subject of the research refers to the development of game and interactive projects—the practice of game balance of computer games. Current trends formed in both scientific and commercial spheres are taken into account. The research methods are modeling and experimentation. The main result is the formed vision on the integration of the received features into the tool of game prototypes generation. Two tools for working with game balance are described.

The presented results are the part of one big applied research aimed at developing a game tool for prototyping computer games, reducing development time and resources by automating through generation of all kinds of content based on natural language text, including, in the long run, game balance. The work is a logical development of research on creating a full-fledged game engine for game designers and screenwriters. (In Russian).

Key words and phrases: game balance, computer games, economic simulation, game design, development tools

2020 Mathematics Subject Classification: 91-08; 68T07

Acknowledgments: This paper has been supported by the Kazan Federal University Strategic Academic Leadership Program («PRIORITY-2030»)

For citation: Gulnara F. Sahibgareeva, Vlada V. Kugurakova. Game Balance Practices // Program Systems: Theory and Applications, 2022, 13:3(54), pp. 255– 273. (In Russian). http://psta.psiras.ru/read/psta2022_3_255-273.pdf

References

- [1] R. Rouse. Game Design: Theory and Practice, Wordware Game Developer's Library, 2nd ed., Jones & Bartlett Learning, 2004, ISBN 978-1556229121, 704 pp. \pdot 256
- [2] J. Pfau, A. Liapis, G. N. Yannakakis, R. Malaka. "Dungeons & replicants II: Automated game balancing across multiple difficulty dimensions via

- deep player behavior modeling", *IEEE Transactions on Games*, 2022 (Early Access). €0↑256
- [3] W. Dworak, E. Filgueiras, J. Valente. "Automatic emotional balancing in game design: Use of emotional response to increase player immersion", HCII 2020: Design, User Experience, and Usability. Design for Contemporary Interactive Environments (July 19–24, 2020, Copenhagen, Denmark), Lecture Notes in Computer Science, vol. 12201, Springer, Cham, 2020, ISBN 978-3-030-49760-6, pp. 426–438.
- [4] T. Sylvester. Designing games: A guide to engineering experiences, 1st ed., O'Reilly Media, Sebastopol, 2013, ISBN 978-1449337933, 413 pp. ↑256
- [5] J. Schell. The Art of Game Design: A Book of Lenses, A K Peters/CRC Press, Boca Raton, 2015, ISBN 9781466598645, 600 pp.↑256
- [6] W. Wang, R. Zhang. "Improved game units balancing in game design through combinatorial optimization", 2021 IEEE International Conference on e-Business Engineering (ICEBE) (12–14 November 2021, Guangzhou, China), 2021, pp. 64–69.
- [7] J. Novak. Game Development Essentials: An Introduction, 3rd ed., Cengage Learning, Melbourne, 2011, ISBN 978-1111307653, 510 pp. 1256
- [8] A. Rollings, E. Adams. Andrew Rollings and Ernest Adams on Game Design, 1st ed., New Riders Publishing, Indianapolis, 2003, ISBN 978-1592730018, 400 pp. ↑256
- [9] G. C. E. Hill. Rethinking Economy-Building Video Games: How might designers inspire new economic models through video game mechanics? Submitted to OCAD University in partial fulfillment of the requirements for the degree of Master of Design in Strategic Foresight and Innovation, OCAD University, Toronto, 2021, 79 pp. URL 1258
- [10] S. Radomski, T. Neubacher. "Formal verification of selected gamelogic specifications", Proceedings of the 2nd EICS Workshop on Engineering Interactive Computer Systems with SCXML (June 23, 2015, Duisburg, Germany), 2015, pp. 30–34. €1↑260
- [11] S. Varvaressos, K. Lavoie, S. Gaboury, S. Halle. "Automated bug finding in video games: A case study for runtime monitoring", Computers in Entertainment, 15:1 (2017), 1, 10 pp. 60 260
- [12] J. Pfau, J. D. Smeddinck, R. Malaka. "Automated game testing with ICARUS: Intelligent completion of adventure riddles via unsupervised solving", CHI PLAY '17 Extended Abstracts: Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play (15–18 October 2017, Amsterdam, The Netherlands), ACM, New York, 2017, ISBN 978-1-4503-5111-9, pp. 153–164. ♣○↑260

- [13] C. Holmgard, M. C. Green, A. Liapis, J. Togelius. "Automated playtesting with procedural personas through MCTS with evolved heuristics", *IEEE Transactions on Games*, **11**:4 (2019), pp. 352–362.
- [14] G. Andrade, G. Ramalho, A.S. Gomes, V. Corruble. "Dynamic game balancing: An evaluation of user satisfaction", Second Artificial Intelligence and Interactive Digital Entertainment Conference, Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 2:1 (2006), pp. 3–8. □R↑263
- [15] T. J. V. Tijs, D. Brokken, W. A. IJsselsteijn. "Dynamic game balancing by recognizing affect", Fun and Games 2008: Fun and Games, Lecture Notes in Computer Science, vol. 5294, Springer, Berlin–Heidelberg, 2008, ISBN 978-3-540-88321-0, pp. 89–93. €↑263
- [16] G. F. Sakhibgareyeva, V. V. Kugurakova. "The concept of automatic creation tool for computer game scenario prototype", *Elektronnyye* biblioteki, 21:3–4 (2018), pp. 235—249 (in Russian). ↑263, 266
- [17] Z. Li, X. Wang, W. Yang, J. Wu, Z. Zhang, Z. Liu, M. Sun, H. Zhang, S. Liu. "A unified understanding of deep NLP models for text classification", IEEE Transactions on Visualization and Computer Graphics, 2022 (Early Access), 14 pp. 1264
- [18] S. Packowski, W. Switzer. "Extracting meaning from text and creating a custom language model to optimize NLP results: NLP hands-on workshop series", CASCON '19: Proceedings of the 29th Annual International Conference on Computer Science and Software Engineering (4−6 November 2019, Toronto, Ontario, Canada), IBM Corp., Riverton, 2019, pp. 382–383. ↑264
- [19] B. A. Kozar, V. V. Kugurakova, G. F. Sakhibgareyeva. "Modification of neural network approaches with object structuring to create 3D locations based on text processing", Programmyye produkty i sistemy, 2022, no. 3 (to appear) (in Russian). ↑264, 266
- [20] V. V. Kugurakova, V. D. Abramov, D. I. Kostyuk, R. A. Sharayeva, R. R. Gazizov, M. R. Khafizov. "Generation of three-dimensional synthetic datasets", *Elektronnyye biblioteki*, 24:4 (2021), pp. 622–652 (in Russian).
- [21] G. F. Sakhibgareyeva, O. A. Bedrin, V. V. Kugurakova. "Development of a video games scenario prototype visualization generating component", Nauchnyy servis v seti Internet: trudy XXII Vserossiyskoy nauchnoy konferentsii (21–25 sentyabrya 2020 g., onlayn), IPM im. M. V. Keldysha, M., 2020, ISBN 978-5-98354-058-3, pp. 581–603 (in Russian).

- [22] G. F. Sakhibgareyeva, O. A. Bedrin, V. V. Kugurakova. "Storyboard as one of the representations of the scenario prototype of computer games", *Elektronnyye biblioteki*, **24**:2 (2021), pp. 408−444 (in Russian). € ↑266
- [23] G. F. Sahibgareeva, O. A. Bedrin, V. V. Kugurakova. "Visualization component for the scenario prototype generator as a video game development tool", *Proceedings of the 22nd Conference on Scientific Services & Internet*, CEUR Workshop Proceedings, vol. **2784**, 2020, pp. 267–282. (R) \(^{1}_{266}
- [24] V. V. Kugurakova, G. F. Sakhibgareyeva, A. Z. Nguyen, A. M. Astaf'yev. "Spatial orientation of objects based on processing of a natural language text for storyboard generation", *Elektronnyye biblioteki*, **23** (2020), pp. 1213–1238 (in Russian). € ↑266
- [25] G. F. Sakhibgareyeva, V. V. Kugurakova. "Prototyping the variability of computer games plots", Nauchnyy servis v seti Internet: trudy XXIII Vserossiyskoy nauchnoy konferentsii (20–23 sentyabrya 2021 g., onlayn), IPM im. M. V. Keldysha, M., 2021, ISBN 978-5-98354-062-0, pp. 347–360 (in Russian). ©↑266
- [26] G. F. Sahibgareeva, V. V. Kugurakova. "Branched structure component for a video game scenario prototype generator", *Proceedings of the 23nd Conference on Scientific Services & Internet*, CEUR Workshop Proceedings, vol. **3066**, 2021, pp. 101–111. □R)↑266
- [27] A. S. Dobrokvashina, E. A. Gazizova. "Automatization of a gaming prototype development based on the result of processing of a formalized game design document", *Uchenyye zapiski ISGZ*, 2019, no. 1(17), pp. 583–589 (in Russian). ORL 1266