Обзор сложности задач в области математики: различные головоломки и видеоигры

Расолько Юлия (Б05-923)

July 29, 2022

Abstract

В этой работе я привожу основные результаты изучения сложности современных видеоигр. Демонстрирую основные правила, по которым формируется изучающийся класс игр и струкуры, благодаря которым можно доказать их принадлежность различным классам.

Введение

Огромная работа по изучению сложности игр была проделана и сейчас проводится выдающимися учёными мира. В этом проекте я собрала основные результаты о старых компьютерных играх, пазлах и некоторых играх двух игроков. Этот проект - компиляция многих статей на эту тему. Для компьютерных игр открытия собраны в хронологическом порядке.

1 2000-2002, вариации игры Push. Их описание.

В этой головоломке робот движется по конечной сетке. Каждая позиция сетки либо пуста, либо либо занята одним блоком препятствий. Во время движения робот может толкать блоки препятствий в направлении своего движения, с учетом определенных ограничений. Задача состоит в том, чтобы решить, сможет ли робот пройти от заданного начального положения до заданной целевой позиции на сетке.
В игре Push-1 блок толкается на одну клетку, в PushPush-1 как только блок препятствия начинает движение, он продолжает его до тех пор, пока не столкнется с другим препятствием или границей сетки, а в играх PushPush-k для фиксированного k $\geq$ 1 робот может толкать не более k блоков одновременно.

1.1 PushPush и Push-1. [1]

Теорема 1.1. *Игры PushPush и Push-1 -* **NP***-трудные.*

1.2 PushPush-k. [2]

Теорема 1.2. *Игры PushPush-k -* **NP***-полные.*

2 2013. Разбор классической структуры игры. Метатеоремы. Некоторые результаты б/д. [3]

Чаще всего действия игрока связаны с управлением неким аватаром. Набор игровых элементов включает в себя стены, которые не может преодолеть аватар, а также различные игровые интерактивный блоки. В общем случае экземпляр задачи будет представлять собой "уровень" данной игры. Описание уровня включает в себя расположение всех элементов игры, таких как стены, предметы, начальное местоположение аватара и т.д. Вопрос всегда заключается в том, можно ли "решить" данный уровень при определенных условиях, например, не потеряв ни одной жизни, и т.д. Точное определение "решаемости" сильно зависит от игры и может варьироваться от достижения места выхода, сбора некоторых предметов, убийства некоторых врагов, выживания в течение определенного времени и т.д.
Следующие метатеоремы предлагают доказательства сложности игр в предположениях наличия в них некоторые игровых элементов.

2.1 Обход локаций и одноразовые пути

В сценарии **обхода локаций** каждая локация должна быть посещена хотя бы один раз, аватар может посетить их несколько раз и в любом порядке. Однако первая локация обычно фиксирована (начальная локация), иногда также и последняя (место выхода). Примером такого сценария может являться сбор предметов, разбросанных по разным локациям.
В сценарии с **одноразовыми путями** существуют игровые элементы, действующие как пути, которые аватар может пройти не более обного раза. Типичным примером являются плитки, которые исчезают, как только аватар проходит по ним.

Теорема 2.1 (Метатеорема 1)**.** *Любая игра с обходом локаций (с или без начальной и конечной) и одноразовыми путями -* $\mathbf{NP} - hard$*.*

Proof. Сведем язык Гамильтоных циклов в неориентированных планарных графах со степенями вершин 3 (**NP**-полный) к уровням. Рассмотрим граф с выделенной вершиной v. Если нужна конечная вершина, добавим вершину u и соединим ее ребром с v. Теперь вершины будем считать локациями, а ребра - одноразовыми путями. Легко заметить, что полученный уровень будет "решаемым" тогда и только тогда, когда в графе есть гамильтонов цикл. □

Следствие 2.1.1. *Существует* **NP***-полная игра с обходом локаций и одноразовыми путями.*

Proof. Рассмотрим игру на неориентированном графе, в которой некоторые выделенные ребра реализуют одноразовые пути, а некоторые выделенные вершины должны быть посещены аватаром для того, чтобы выиграть. Тогда по метатеореме 1 эта игра является **NP**-трудной, а сертификат - это инъективная последовательность выделенных вершин и выделенных ребер. □

2.2 Жетоны и платные дороги

Жетоны - это предметы, которые может переносить аватар, а **платные дороги** - это специальные пути, соединяющие два места. Каждый раз, когда аватар пересекает платную дорогу, он должен "потратить" жетон. Если у аватара нет жетона, то он не может пройти по платной дороге. Мы различаем два типа жетонов: **коллекционные жетоны**, которые могут размещаются дизайнером игры в определенных местах и могут быть подобраны аватаром, и **кумулятивные жетоны**, любое количество которых может носить с собой аватар одновременно. Например, в Pac-Man есть энергетические таблетки, которые можно рассматривать как коллекционные жетоны, которые не являются кумулятивными.

Теорема 2.2 (Метатеорема 2)**.** *Игра -* $\mathbf{NP} - hard$*, если имеет место одно из следующих условий:*

1. *в игре есть коллекционные жетоны, платные дороги и обход локаций.*
2. *в игре есть кумулятивные жетоны, платные дороги и обход локаций.*
3. *в игре есть коллекционные кумулятивные жетоны, платные дороги и аватар должен добратъся до места выхода.*

Proof. Еще раз приведем сведение Гамильтонова цикла для всех трех частей метатеоремы, слегка изменяя его в зависимости от наших гипотез.
Для части (1) строим граф, как в доказательстве метатеоремы 1, считаем каждую вершину как место, которое должен пройти аватар. Каждое ребро затем реализуется как платная дорога, и в каждую вершину помещается по одному жетону, за исключением конечной вершины u, где мы не помещаем никаких жетонов, и начальной вершины v, где мы размещаем два жетона. Этот обход действителен, даже если жетоны не кумулятивны: каждый раз аватар собирает один новый жетон и немедленно тратит его на платной дороге.
Конструкция для части (2) такая же, но вместо того, чтобы разбрасывать n+1 жетонов по всему уровню (где n - количество вершин без дополнительной u), мы предполагаем, что аватар уже имеет n + 1 жетонов в начале игры.
Для части (3) мы модифицируем предыдущее доказательство следующим образом: помещаем два жетона в каждую вершину, кроме u, где мы не помещаем ни одного жетона. Затем мы реализуем каждое ребро как платную дорогу, за исключением ребра между v и u, которое реализуется как последовательность из n платных дорог. На уровне есть 2n жетонов, и n из них должны быть использованы для путешествия из v в u, поэтому на других платных дорогах можно потратить не более n жетонов. Игрок должен найти такой путь в G, который начинается и заканчивается в v, проходит не более n ребер и посещает n различных вершин. Это эквивалентно нахождению гамильтонова цикла в G. □

Следствие 2.2.1. *Существует* **NP***-полная игра с коллекционными кумулятивными жетонами и платными дорогами, в которой аватар должен добратъся до места выхода.*

Proof. Рассмотрим игру на графе, в котором некоторые выделенные ребра реализуют платные дороги, каждая вершина может содержать некоторое количество коллекционных жетонов, а одна выделенная вершина является местом выхода. Действительно, сертификат для этой игры это просто инъективная последовательность платных дорог, потому что мы можем предположить, что аватар всегда собирает все жетоны, до которых он может добраться, не пересекая платные дороги, и поэтому ни одна платная дорога не должна быть пройдена дважды. □

2.3 Двери и ключи

Дверь - это игровой элемент, который может быть открыт или закрыт и может быть пройден аватаром тогда и только тогда, когда он открыт. **Ключ** - это тип жетона, который используется на дверях. Любой ключ может открыть любую дверь, но ключ "расходуется", как только он используется. Таким образом, идея "ключ-дверь" в некоторой степени похожа на идею жетонов и платных дорог, с той лишь разницей, что дверь, открытая ключом, остается открытой и может быть пройдена несколько раз после этого без использования новых ключей.
Мы снова проводим различие между **коллекционными ключами**, которые может найти аватар и подобрать, и **кумулятивными ключами**, любое количество которых можно носить с собой одновременно. Чтобы сформулировать следующий результат, нам необходимо ввести понятие **одностороннего пути**, который представляет собой путь, который может быть пройден аватаром только в одном конкретном направлении.

Теорема 2.3 (Метатеорема 3). *Игра -* $\mathbf{NP} - hard$*, если она содержит двери и односторонние пути, и имеет место одно из следующих условий:*

1. *в игре есть коллекционные ключи и обход локаций.*
2. *в игре есть кумулятивные ключи и обход локации.*
3. *в игре есть коллекционные кумулятивные ключи, а аватар должен достичь места выхода.*

Proof. Сведем язык Гамильтоных циклов в ориентированных планарных графах с двумя входящими и одним выходящим ребром или одним входящим и двумя выходящими ребрами для каждой вершины (**NP**-полный) к уровням. Все три части нашего доказательства основаны на одной и той же конструкции: дан один такой направленный граф G на n вершинах, мы выбираем вершину v с входной степенью два и выходной один (которая существует), и присоединяем к ней новое исходящее ребро, заканчивающееся новой вершиной u. Каждую вершину считаем игровой локацией, а каждое ребро - односторонним путем. Начальной локацией аватара будет вершина v, а если требуется локация выхода, то она помещается в u. Более того, мы размещаем закрытую дверь в каждом одностороннем пути, кроме пути между v и u. Чтобы доказать часть (1), поместим по одному ключу в каждой вершине, кроме u, и предположим, что аватар должен пройти через все локации (последней из которых будет u). Уровень решен если и только если G имеет гамильтонов цикл. Для части (2) мы не помещаем в уровень никаких ключей, но предполагаем, что у аватара уже есть n ключей, когда начинается игра. Наконец, для части (3) мы размещаем два ключа в каждом месте, кроме u, и размещаем n дверей на пути между v и u. □

Следствие 2.3.1. *Существует* **NP***-полная игра с дверьми, односторонними путями, коллекционными кумулятивными ключами дверьми и обходом локаций, в которой аватар должен добраться до места выхода.*

Proof. Рассмотрим игру на графе, в котором некоторые ребра могут реализовывать двери или односторонние пути, некоторые выделенные вершины должны быть пройдены, некоторые вершины содержат коллекционные кумулятивные ключи, а одна вершина является место выхода. Сертификат для этой игры - это инъективная последовательность выделяющихся вершин, вершин, содержащих ключи, и ребер, содержащих двери. Действительно, все ключи содержащиеся в вершине, могут быть взяты, как только эта вершина достигнута, а любая открытая дверь становится обычным путем и ее не нужно открывать второй раз. Следовательно, игра находится в NP, а по метатеореме 3 она NP-полна. □

2.4 Двери и нажимные пластины

Существуют и другие способы изменения состояния двери, например, нажатие на **нажимную пластину** - напольную кнопку, которая срабатывает всякий раз, когда аватар на нее наступает, и ее действие может быть направлено как на открытие, так и на закрытие определенной двери. Каждая нажимная пластина связана только с одной дверью, и каждая дверь может управляться не более чем двумя нажимными пластинами (одна открывает, другая закрывает). Конечно, все наши результаты по трудности будут справедливы и для более общего сценария, в котором любое количество дверей управляется одной нажимной пластиной, или любое количество нажимных пластин управляет одной дверью.
Мы говорим, что игра **допускает пересечения**, если существует способ предотвратить переключение аватара между двумя пересекающимися путями. Некоторые двумерные игры изначальнореализуют переходы через мосты или туннели.

Теорема 2.4 (Метатеорема 3). *Если в игре есть двери и нажимные плиты, а аватар должен достичь места выхода, чтобы победить, то:*

1. *даже если ни одна дверь не может быть закрыта нажимной плитой, и если игра допускает пересечения, то игра является* **P***-трудной.*
2. *даже если никакие две нажимные пластины не контролируют одну и ту же дверь, игра* **NP***-трудна.*
3. *если каждая дверь может контролироваться двумя нажимными пластинами, то игра является* **PSPACE***-трудной.*

Следствие 2.4.1. *Существуют игры G1, G2, G3, в которых есть двери и нажимные плиты, в которых в которых аватар должен достичь места выхода, такие, что:*

1. *В G1 нажимные пластины могут только открывать двери, игра допускает пересечения, и G1 является* **P***-полной.*
2. *В G2 никакие две нажимные пластины не управляют одной и той же дверью, и G2 -* **NP***-полный.*
3. *В G3 каждая дверь может управляться двумя нажимными пластинами, и G3 является* **PSPACE***-полной.*

Proof. Мы рассматриваем игры, проводимые на графах, вершины которых могут содержать нажимные плиты, а ребра - двери. пластины, а ребра могут содержать двери.
Тогда G1 принадлежит **P**, поскольку множество вершин, доступных аватару, может увеличиваться только всякий раз, когда нажимная плита активируется. Когда новые нажимные плиты становятся доступными, аватар немедленно активирует их, открывая новые двери, пока либо выход не станет доступным, либо не останется ни одной новой нажимной плиты.
G2 находится в **NP**, потому что ни одна нажимная пластина не может отменить действие другой нажимной пластины. Поэтому мы можем сделать вид, что нажимная пластина исчезает, как только она активирована. Отсюда следует, что сертификат для этой игры - это просто инъективная последовательность нажимных пластин.
Наконец, чтобы убедиться, что G3 находится в **PSPACE** = **NPSPACE** (теорема Савича [4]), достаточно заметить, что состояние уровня можно хранить в линейном пространстве, выделяя один бит для каждой двери и храня положение аватара в графе. Тогда сертификат - это просто путь в графе. □

2.5 Двери и кнопки

Кнопка похожа на нажимную пластину, за исключением того, что игрок может выбирать, нажимать ее или нет. Игры с кнопками в целом не сложнее, чем игры с нажимными плитами, потому что нажимная плита может тривиально имитировать кнопку (например, можно создать небольшое ответвление и поместить в него плиту). Однако, поскольку обратное утверждение не столь очевидно, мы позволим одной кнопке действовать на несколько дверей, в отличие от нажимных пластин. Кнопка, действующая на k дверей одновременно, называется **k-кнопкой**.

Теорема 2.5 (Метатеорема 3)**.** *Если в игре есть двери и k-кнопки, а аватар должен достичь места выхода, чтобы выиграть, то:*

1. *если $k \geq 1$ и игра допускает пересечения, то игра является* **P***-трудной.*
2. *если $k \geq 2$, то игра* **NP***-трудная.*
3. *если $k \geq 3$, то игра является* **PSPACE***-трудной.*

Следствие 2.5.1. *Существуют игры G1 и G2 с дверями, в которых аватар должен добраться до места выхода, причем такие, что:*

1. *G1 состоит из 1-кнопок и является* **P***-полной.*
2. *G2 имеет k-кнопки, при $k > 3$, и является* **PSPACE***-полной.*

2.6 Результаты, полученные с помощью этих метатеорем

Boulder Dash (1984) - **NP**-трудная
Deflektor (1987) - принадлежит Ł
Lemmings (1991) - **NP**-трудная
Lode Runner (1983) - **NP**-трудная
Mindbender (1989) - **NL**-трудная
Pac-Man (1980) - **NP**-трудная
Pipe Mania (1989) - **NP**-полная
Prince of Persia (1989) - **PSPACE**-полная
Puzzle Bobble 3 (1996) - **NP**-полная
Skweek (1989) - **NP**-трудная
Starcraft (1998) - **NP**-трудная
Tron (1982) - **NP**-трудная

3 2015. Классические Nintendo игры. [5]

3.1 Super Mario Bros

Super Mario Bros. разделена на восемь миров, каждый из которых состоит из четырёх уровней. Марио (или Луиджи) должен пройти до конца уровня, перепрыгивая через ямы и уворачиваясь от врагов на пути. По пути Марио встречаются различные объекты такие как платформы (некоторые из которых могут падать если Марио встанет на них), ступени из блоков и пружины. Также по пути появляются трубы, некоторые из которых

являются проходом в секретные комнаты с множеством монет.
Многие враги могут быть поражены прыжком Марио. Купы Трупы и Жуки Баззи прячутся в свой панцирь, если на них прыгнуть, а Купы Паратрупы теряют свои крылья. Растений Пираний и Спайни нельзя одолеть прыжком, но можно уничтожить огненным цветком или же просто перепрыгивать через них.
Если маленький Марио получит урон, упадёт в яму или же закончится время на таймере, игрок теряет одну жизнь и уровень начинается либо с самого начала, либо от одного из невидимых "чекпоинтов", через которые прошёл игрок.
В конце каждого уровня стоит замок с флагштоком. Когда Марио достигает флагштока, он спускает вражеский флаг и входит в замок, завершая уровень.

Теорема 3.1. *Задача достижимости цели из начала уровня в Super Mario Bros -* **NP***-трудная задача.*

3.2 Donkey King Country

В Donkey King Country также нужно передвигаться по уровню, уворачиваясь или убивая различных врагов и собирая очки.

Теорема 3.2. *Задача достижимости цели из начала уровня в Donkey King Country -* **NP***-трудная задача.*

Теорема 3.3. *Задача достижимости цели из начала уровня в Donkey King Country -* **PSPACE***-полная задача.*

3.3 The Legend Of Zelda

Теорема 3.4. *Задача достижимости цели из начала уровня в The Legend Of Zelda -* **NP***-трудная задача.*

Теорема 3.5. *Задача достижимости цели из начала уровня в The Legend Of Zelda -* **PSPACE***-полная задача.*

4 2016. Новые результаты о Super Mario Bros [6]

Теорема 4.1. *Задача достижимости цели из начала уровня в Super Mario Bros -* **PSPACE***-полная задача.*

5 Результаты для игр двух игроков

5.1 Шахматы [7]

Теорема 5.1. *Вычисление идеальной стратегии в обобщенной версии шахмат n*n -* ***EXPTIME****-полная задача.*

5.2 Шашки [8] [9]

Теорема 5.2. *Шашки, обобщенные на доску n*n -* ***EXPTIME****-полная задача.*

5.3 Го [10]

Теорема 5.3. *Го, обобщенный на доску n*n -* **PSPACE***-трудная задача.*

Я выбрала самые известные примеры, больше результатов можно найти в [11] и [12].

References

[1] Erik D. Demaine, Martin L. Demaine, and Joseph O'Rourke. "PushPush and Push-1 are NP-hard in 2D". In: (2000). URL: https://arxiv.org/pdf/cs/0007021.pdf.

[2] Erik D. Demaine, Michael Hoffmann, and Markus Holzer. "PushPush-k is PSPACE-Complete". In: (2002). URL: https://erikdemaine.org/papers/PushPush1_PSPACE_FUN2004/paper.pdf.

[3] GIOVANNI VIGLIETTA. "Gaming is a hard job, but someone has to do it!" In: (2013). URL: https://arxiv.org/pdf/1201.4995.pdf.

[4] C. H. Papadimitriou. "Computational complexity". In: (1994).

[5] Greg Aloupis, Erik D. Demaine, and Giovanni Viglietta Alan Guo. "Classic Nintendo Games are (Computationally) Hard". In: (2015). URL: https://arxiv.org/pdf/1203.1895.pdf.

[6] Erik D. Demaine, Giovanni Viglietta, and Aaron Williams. "Super Mario Bros. Is Harder/Easier than We Thought". In: (2016). URL: http://erikdemaine.org/papers/Mario_FUN2016/paper.pdf.

[7] A. S. Fraenkel and D. Lichtenstein. "Computing a perfect strategy for n*n chess requires time exponential in n". In: (1981).

[8] A. S. Fraenkel et al. “The complexity of checkers on an N*N board - preliminary report”. In: (1978).

[9] J. M. Robson. “N by N checkers is Exptime complete”. In: (1984).

[10] D. Lichtenstein and M. Sipser. “Go is polynomial-space hard”. In: (1980).

[11] “Game complexity”. In: (). URL: https://en.wikipedia.org/wiki/Game_complexity.

[12] “Computational Complexity of Games and Puzzles”. In: (). URL: https://www.ics.uci.edu/~eppstein/cgt/hard.html.