

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Прикладная математика и информатика

Исследование операций и принятие решений в задачах оптимизации, управления и  
экономики

Грецова Мария Вадимовна

## **Модели формирования коалиций**

Бакалаврская работа

Научный руководитель:

канд. физ.-мат. наук Наумова Н.И.

Рецензент:

канд. физ.-мат. наук Бухвалова В.В.

Санкт-Петербург

2016

SAINT-PETERSBURG STATE UNIVERSITY  
Applied Mathematics and Computer Science  
Operation Research and Decision Making in Optimisation, Control and Economics  
Problems

Gretcova Mariia

## **Models of forming coalitions**

Bachelor's Thesis

Scientific supervisor:  
ass. prof. Natalia Naumova

Reviewer:  
ass. prof. Vera Bukhvalova

Saint-Petersburg

2016

# Оглавление

Введение . . . . .	2
1. Основные определения . . . . .	4
1.1. Определение гедонистической игры . . . . .	4
1.2. Свойства решений . . . . .	5
1.3. Концепции решения основанная на групповом отклонении . . . . .	6
1.4. Концепции решений основанные на одиночном отклонении игроков . . . . .	7
2. Задача «об устойчивом браке» . . . . .	11
3. Аддитивно-сепарабельные гедонистические игры . . . . .	14
4. Игры с общим порядком предпочтений . . . . .	17
5. $\mathcal{B}, \mathcal{W}$ и $TOP$ -игры . . . . .	19
5.1. Определение $\mathcal{B}$ и $\mathcal{W}$ -игр . . . . .	19
5.2. $TOP$ -игры . . . . .	21
5.3. Алгоритм «Top covering» . . . . .	25
Заключение . . . . .	30
Литература . . . . .	31
Приложение . . . . .	33

## Введение

Теория игр – это математическая дисциплина, в которой изучаются ситуации столкновения интересов, их свойства и особенности, и разрабатываются методы оптимального в том или ином смысле поведения игроков в таких ситуациях. Кооперативной называется игра, в которой группы игроков – коалиции – могут объединять свои усилия.

Люди часто объединяются в группы из соображений личной выгоды. Например, когда они организуют фирму с целью производства каких-то товаров, или клуб по интересам. В таких ситуациях каждый индивид имеет личные предпочтения относительно того, какой группе он хочет принадлежать. Обычно его интересует размер группы, другие входящие в нее индивиды и то чем группа занимается. Подобные ситуации неоднократно анализировались в рамках теории кооперативных игр.

В 1962 году Дэвид Гейл и Ллойд Шепли сформулировали задачу распределения студентов по колледжам [8]. Она заключается в том, чтобы каждого студента из заданного множества определить в один из нескольких колледжей, если известны предпочтения студентов среди колледжей и колледжей среди студентов с учетом того, что каждый колледж может принять ограниченное число абитуриентов и каждый абитуриент может попасть только в один колледж. При рассмотрении этой задачи Гейл и Шепли ввели понятия устойчивого и неустойчивого разбиения, а также рассмотрели задачу «об устойчивом браке».

Игры, в которых задача каждого игрока – попасть в наиболее приятную для себя коалицию, называются гедонистическими [7]. Они были определены в самостоятельный класс игр Жаком Дзенем и Джозефом Гринбергом в 1980 году. Такое название было выбрано не случайно. Гедонизм(греч. hedone–наслаждение) – это этико-психологическое учение, возникшее в античности, утверждающее, что наслаждение является высшим благом, целью личности, критерием истинности и целесообразности, основным мотивом ее поведения [20].

За последние 30 лет гедонистические игры стали объектом изучения многих ученых. Наиболее практически-значимые результаты были получены в работах Элвина Рота, сумевшего связать задачу о поиске устойчивого паросочетания с теорией коалиционных игр. В США при его участии были изменены многие общественные механизмы распределения, в том числе механизмы распределения интернов по больницам, система распределения учащихся по школам округов Нью-Йорк и Бостон. Помимо этого, Э.Рот совместно с Тайфуном Сомезом построили научный фундамент системы обмена

донорскими органами. В 2012 году Ллойд Шепли и Элвин Рот были удостоены Нобелевской премии по экономике «за теорию устойчивого распределения и практики дизайна рынка».

Целью данной работы является изучение принципов устойчивости для коалиционных структур в гедонистических играх с использованием практических примеров. На сегодняшний день, не существует материалов на русском языке по теории гедонистических игр. В качестве основы для исследования была взята обзорная работа [2].

В первой главе приведены основные понятия, связанные с гедонистическими играми, и разобраны поясняющие примеры.

Вторая глава посвящена задаче «об устойчивом браке» [8], поставленной Гейлом и Шепли в 1962 году – приведена ее формулировка, а также алгоритм решения и практические примеры реализации этого алгоритма.

Далее рассмотрены игры с различными принципами формирования предпочтений.

В третьей главе рассказывается о гедонистических играх с аддитивно-сепарабельными предпочтениями. Для симметричной аддитивно-сепарабельной игры объясняется способ нахождения устойчивого по Нэшу решения, приведенный в работе Анны Богомольной и Мэттью Джексона [4], приводятся примеры нахождения таких решений.

В четвертой главе разобран алгоритм нахождения решения из ядра, для игр с общим порядком предпочтений, созданный Варутом Саксомпонгом [14].

Пятая глава рассказывает о  $\mathcal{B}$ ,  $\mathcal{W}$  и  $\mathcal{TOP}$ -играх. Помимо основных определений, взятых из [2], в ней выводится взаимосвязь между этими видами игр. Приводится алгоритм «Top covering», который находит разбиения из ядра для  $\mathcal{TOP}$ -игр [1], а также доказательство его корректности [5].

Алгоритм «Top covering» был реализован на языке  $C\#$ . Текст программы приведен в приложении.

# 1. Основные определения

## 1.1. Определение гедонистической игры

**Определение 1.1.** Пусть  $N$  – множество игроков.

**Коалицией** называется непустое подмножество  $N$ .

Пусть  $N_i = \{S \subseteq N : i \in S\}$  – множество всех коалиций, в которых содержится  $i \in N$ .

**Коалиционной структурой** называется разбиение множества игроков  $N$  на непересекающиеся коалиции.

**Отношением предпочтения (или предпочтением)**  $\succeq_i$  игрока  $i$  на множестве  $N_i$  называется полное, транзитивное бинарное отношение, заданное на  $N_i$ .

**Гедонистической игрой формирования коалиций** называется пара  $(N, \succeq)$ , где  $\succeq = (\succeq_1, \dots, \succeq_{|N|})$  – набор предпочтений всех игроков.

**Решением гедонистической игры** будем называть коалиционную структуру, удовлетворяющую определенным свойствам.

Пусть  $S, T \in N_i$

$S \succ_i T$  означает, что для игрока  $i$  строго предпочтительнее  $S$ , чем  $T$ ;

$S \sim_i T$  означает, что для игрока  $i$  коалиции  $S$  и  $T$  одинаково предпочтительны;

$S \succeq_i T$  означает, что для игрока  $i$  коалиция  $S$  нестрого предпочтительнее  $T$ , т.е. возможны оба вышеперечисленные варианта;

Для коалиционной структуры  $\pi$  и игрока  $i$ ,  $\pi(i)$  – коалиция из  $\pi$ , которая содержит  $i$ .

**Пример 1.1.** В гедонистической игре участвует три игрока 1, 2, и 3, их предпочтения среди коалиций, перечислены ниже.

Все игроки предпочитают коалицию из двух человек, а не из одного или трех.

Для каждого игрока предпочтительнее состоять в коалиции одному, чем втроем.

Для игрока 1 предпочтительнее быть с 2, чем с 3

Для игрока 2 предпочтительнее быть с 3, чем с 1

Для игрока 3 предпочтительнее быть с 1, чем с 2

Результатами игры могут быть разбиения всех игроков на коалиции:

$\{123\}, \{12\}\{3\}, \{13\}\{2\}, \{23\}\{1\}$  и  $\{1\}\{2\}\{3\}$ .

Игрок может покинуть коалицию, чтобы присоединиться к другой коалиции или, чтобы играть самостоятельно. Это явление называют **одиночным отклонением** иг-

рока. Говорят, что у игрока есть *причины* для одиночного отклонения, если существует коалиция, которая для него строго предпочтительнее текущей. Такое одиночное отклонение называют *предпочтительным*. Какие-то одиночные отклонения могут быть запрещены и называться *запрещенными*. Не запрещенные одиночные отклонения называют *разрешенными*.

**Определение 1.2.** Коалиционная структура  $\pi$  называется *устойчивой* относительно класса разрешенных одиночных отклонений, если ни один из игроков не имеет предпочтительного разрешенного одиночного отклонения.

Допустим, что в Примере 1.1. разрешены любые одиночные отклонения. Очевидно, что если коалиция не двухточечная, то её игроки имеют причины для отклонения. Получается, что все коалиционные структуры содержат хотя бы одного игрока, имеющего причины для отклонения, то есть не являются устойчивыми.

При сокращении класса разрешенных одиночных отклонений устойчивость коалиционной структуры слабеет, т.е. если в результате запрета некоторых отклонений какая-то коалиционная структура стала устойчивой, то ее устойчивость считается слабее устойчивости тех коалиционных структур, которые уже были устойчивы до запрета.

Допустим, что в Примере 1.1 игрокам запрещено выходить из коалиции без разрешения(одобрения) всех её участников. Тогда все разбиения, содержащие двухточечные коалиции будут устойчивы, так как оставшемуся в одиночестве игроку из двухточечной коалиции хуже быть одному, чем вдвоем, и он запретит отклонение второго. При одноэлементном разбиении любое отклонение разрешено, потому что оно приводит к разбиению на одно и двухточечные коалиции. Получается, что одноэлементное разбиение является неустойчивым. Если же разбиение состоит из одноточечных множеств, то разрешено любое одиночное отклонение, и все игроки хотят создать двухточечную коалицию, поэтому это разбиение тоже является неустойчивым.

В игре могут быть запрещены некоторые разбиения. В этом случае, решение выбирается из класса разрешенных разбиений.

## 1.2. Свойства решений

### 1) *Индивидуальная рациональность.*

**Определение 1.3.** Разбиение  $\pi$  является *индивидуально рациональным (ИР)*, если для каждого игрока  $i \in N$  коалиция  $\pi(i)$  не менее предпочтительна, чем коалиция  $\{i\}$ , т.е.  $\forall i \in N, \pi(i) \succeq_i \{i\}$ .

Индивидуальная рациональность весьма слабое свойство. Оно включается во многие концепции решения, потому что является необходимым условием устойчивости.

В Примере 1.1 все разбиения, кроме  $\{123\}$ , являются индивидуально рациональными.

## 2) Совершенство

**Определение 1.4.** Разбиение  $\pi$  является **совершенным**, если каждый из игроков  $i \in N$  находится в одной из своих наиболее предпочтительных коалиций.

Совершенное разбиение устойчиво по отношению к любому отклонению, но существует редко.

В Примере 1.1 для каждого из игроков множество наиболее предпочтительных игроков одноточечно:  $\{12\}, \{23\}$  и  $\{31\}$  для 1, 2 и 3 - соответственно. Объединение этих коалиций не является разбиением, поэтому для этой игры нет совершенного разбиения.

## 3) Оптимальность по Парето

**Определение 1.5.** Разбиение  $\pi$  **оптимально по Парето (ПО)**, если не существует разбиения  $\pi'$ , такого, что  $\pi'(j) \succeq_j \pi(j)$  для каждого  $j$  и  $\pi'(i) \succ_i \pi(i)$  хотя бы для одного  $i$ .

(Т.е. не существует разбиения которое было бы не хуже для всех игроков и лучше хотя бы для одного.)

В Примере 1.1 разбиение  $\pi = \{12\}\{3\}$  является Парето-оптимальным.

Действительно,

если  $\pi' = \{13\}\{2\}$ , то  $\{13\} \not\succeq_1 \{12\}$ ;

если  $\pi' = \{23\}\{1\}$ , то  $\{1\} \not\succeq_3 \{12\}$ ;

если  $\pi' = \{123\}$ , то  $\{123\} \not\succeq_1 \{12\}$ .

По-анalogии, Парето-оптимальными являются решения  $\{13\}\{2\}$  и  $\{23\}\{1\}$ .

В литературе свойство оптимальности по Парето иногда называют свойством групповой стабильности.

## 1.3. Концепции решения основанная на групповом отклонении

**Определение 1.6.** **Сильно блокирующей коалицией** называется  $S \subseteq N, S \neq \emptyset$  такое, что  $\forall i \in S : S \succ_i \pi(i)$ .

**Разбиение  $\pi$  находится в слабом ядре**, если для него не существует сильно блокирующей коалиции, т.е.  $\forall S : S \subseteq N, S \neq \emptyset \exists i \in S : \pi(i) \succeq_i S$ .



То есть коалиция  $S$  сильно блокирует разбиение  $\pi$ , если для каждого игрока  $i \in S$  коалиция  $S$  строго предпочтительнее его текущей коалиции  $\pi(i)$  в разбиении  $\pi$ , а разбиение находится в ядре, если для него нет сильно блокирующей коалиции. В дальнейшем, слабое ядро будем называть просто ядром.

Примере 1.1 не существует разбиения с сильным ядром. Каждая двухточечная коалиция является наиболее предпочтительной только для одного из ее участников, поэтому коалиция, более предпочтительная для второго будет блокировать двухэлементное разбиение, содержащее вышеназванную коалицию. Любая двухточечная коалиция сильно блокирует одноэлементное разбиение и разбиение, состоящее из одноточечных коалиций.

**Определение 1.7.** *Говорят, что коалиция  $S \subseteq N$  слабо блокирует разбиение  $\pi$ , если для каждого игрока  $i \in S$  коалиция  $S$  слабо предпочтительнее его текущей коалиции  $\pi(i)$  в разбиении  $\pi$ , и существует хотя бы один игрок  $j \in S$  для которого коалиция  $S$  строго предпочтительнее  $\pi(j)$ , т.е.  $\forall i \in S : S \succeq_i \pi(i), \exists j \in S : S \succ_j \pi(j)$ .*

*Разбиение  $\pi$  находится в сильном ядре, если для него не существует слабо блокирующей коалиции, т.е.  $\forall S : S \subseteq N, S \neq \emptyset \exists j \in S : \pi(j) \succ_j S$ .*

**Замечание 1.1.** *Каждая сильно блокирующая коалиция является слабо блокирующей, а каждая коалиция из сильного ядра содержится в слабом.*

## 1.4. Концепции решений основанные на одиночном отклонении игроков

*Разбиение  $\pi$  устойчиво по Нэшу*, когда  $\forall i \in N \quad \pi(i) \succeq_i S \cup \{i\} \forall S \in \pi \cup \{\emptyset\}$ , то есть ни одному из игроков не выгодно отклониться.

В Примере 1.1 ни одно разбиение не является устойчивым по Нэшу, так как во всех разбиениях есть игроки, которые не состоят в двухточечных коалициях, но хотят состоять.

*Разбиение  $\pi$  индивидуально-устойчиво*, когда

$\forall i \in N$  если  $\exists S \in \pi \cup \{\emptyset\} : S \neq \pi\{i\}, S \cup \{i\} \succ_i \pi(i)$ , то  $\exists j \in S : S \succ_j S \cup \{i\}$ , то есть ни одним из игроков не может отклониться в более выгодную для себя коалицию, если это не выгодно хотя бы одному из участников этой коалиции.

Индивидуальная устойчивость слабее устойчивости по Нэшу, так как достигается на более узком классе разрешенных одиночных отклонений.

**Разбиение  $\pi$  контрактно-устойчиво**, когда

$\forall i \in N$ , если  $\exists S \in \pi \cup \{\emptyset\} : S \neq \pi(i), S \cup \{i\} \succ_i \pi(i)$ , то  $\exists j' \in \pi(i) : \pi(i) \succ_{j'} \pi(i) \setminus \{i\}$ , то есть ни один из игроков не может покинуть коалицию, если это невыгодно хотя бы одному из ее участников.

Контрактная устойчивость по Нэшу является более слабой, чем устойчивость по Нэшу, так как достигается на более узком классе разрешенных одиночных отклонений. Если в Примере 1.1 запретить выход из коалиции без одобрения одного из ее участников, то, как уже было разобрано, все разбиения содержащие двухточечные коалиции будут контрактно-устойчивыми.

**Разбиение  $\pi$  контрактно-индивидуально устойчиво**, когда

$\forall i \in N$ , если  $\exists S \in \pi \cup \{\emptyset\} : S \neq \pi(i), S \cup \{i\} \succ_i \pi(i)$ , то  $\exists j \in S : S \succ_j S \cup \{i\}$  или  $\exists j' \in \pi(i) : \pi(i) \succ_{j'} \pi(i) \setminus \{i\}$

Очевидно, что контрактно-индивидуальная устойчивость еще слабее чем индивидуальная и контрактная устойчивости и слабее устойчивости по Нэшу.

**Пример 1.2.** *Различные виды устойчивости удобно иллюстрировать на брачных отношениях. В этом случае класс разрешенных разбиений состоит из одиноких игроков и разнополых пар.*

*Рассмотрим гедонистическую игру, в которой игроками будут герои романа Михаила Шолохова «Тихий Дон» – Григорий, Аксинья, Наталья и Степан.*

*Обозначим игроков по первым буквам имен.*

$$N = \{Г, А, С, Н\}$$

*Будем считать, что Григорий, в характерной для себя манере, не может выбрать между Аксиньей и Натальей и не хочет быть один. Аксинья не хочет жить со Степаном и мечтает соединиться с Григорием. Наталья и Степан пытаются сохранить свои семейные отношения с Григорием и Аксиньей соответственно и не рассматривают друг друга в качестве возможных спутников жизни.*

*Таким образом, предпочтения игроков среди коалиций выглядят следующим образом:*

$$\{ГА\} \sim_G \{ГН\} \succ_G \{Г\};$$

$$\{ГА\} \succ_A \{А\} \succ_A \{СА\};$$

$$\{СА\} \succ_C \{С\} \succ_C \{СН\};$$

$$\{ГН\} \succ_H \{Н\} \succ_H \{СН\}.$$

*Класс разрешенных разбиений состоит из следующих разбиений:*

$$\{ГА\}\{СН\}, \{СА\}\{ГН\}, \{Г\}\{А\}\{СН\}, \{ГА\}\{С\}\{Н\}, \{СА\}\{Г\}\{Н\}, \{С\}\{А\}\{ГН\}, \{Г\}\{А\}\{С\}\{Н\}.$$

Оценим устойчивость этих разбиений.

1) В разбиении  $\{ГА\}\{СН\}$  Степан и Наталья имеют причины для отклонения в одноточечные коалиции, поэтому оно **не устойчиво по Нэшу**. Так как отклонение в одноточечную коалицию означает присоединение к пустой коалиции, в которой нет игроков, которые могли бы этому препятствовать, поэтому разбиение **не индивидуально-устойчиво**. Степан и Наталья – единственные, кто хочет отклониться – состоят в одной коалиции, и они не будут препятствовать отклонению друг друга, поэтому разбиение **не контрактно-устойчиво**.

Коалиции  $\{С\}$  и  $\{Н\}$  сильно блокируют разбиение, поэтому оно **не принадлежит слабому ядру**.

2) При разбиении  $\{СА\}\{ГН\}$ , Аксинья хочет отклониться в одноточечную коалицию, поэтому оно **не устойчиво по Нэшу** и **не индивидуально-устойчиво**. Отклонение Аксиньи не желательно для Степана, поэтому он будет ему препятствовать и разбиение будет **контрактно-устойчиво**.

Коалиция  $\{А\}$  сильно блокирует разбиение, что означает, что оно **не принадлежит слабому ядру**.

3) Разбиение  $\{Г\}\{А\}\{СН\}$  **не устойчиво по Нэшу** по целому ряду причин. Во-первых, Аксинья и Григорий хотят объединиться, поэтому разбиение **не индивидуально устойчиво**. Во-вторых, Степан и Наталья не хотят быть вместе, поэтому разбиение **не контрактно-устойчиво**.

Для этого разбиения есть целый ряд сильно блокирующих коалиций:  $\{Г, А\}$ ,  $\{Г, Н\}$ ,  $\{С\}$ ,  $\{Н\}$ .

Оно **не принадлежит слабому ядру**.

4) При разбиении  $\{ГА\}\{С\}\{Н\}$  никто из игроков не имеет причин для одиночного отклонения, поэтому оно **устойчиво по Нэшу**. Сильно блокирующих коалиций нет, но коалиция  $\{Г, Н\}$  будет слабо блокирующей, поэтому разбиение **лежит в слабом ядре**, но не в сильном.

5) В разбиении  $\{СА\}\{Г\}\{Н\}$  у Аксиньи, Григория и Натальи есть причины для отклонения, причем и Григорий, и Наталья имеют причины для соединения, поэтому разбиение **не будет устойчиво по Нэшу, не индивидуально устойчиво и не контрактно устойчиво**.

Причем, если Степан и может помешать Аксинье уйти к Григорию, то никто не мешает объединиться Григорию и Наталье, которые оба этого хотят.

Коалиции  $\{ГА\}$ ,  $\{ГН\}$  и  $\{А\}$  сильно блокируют разбиение, поэтому оно не принадле-

*жсит слабому ядру.*

6) При разбиении  $\{C\}\{A\}\{ГН\}$  только Степан имеет причины для одиночного отклонения. Он хочет присоединиться к Аксинье, а она этому препятствует, поэтому разбиение **индивидуально устойчиво**. Коалиция  $\{C\}$  **одноточечная**, поэтому разбиение **не контрактно устойчиво**.

Нет сильно блокирующих коалиций, но есть слабо блокирующая –  $\{ГН\}$ , значит **разбиение принадлежит слабому ядру**.

7) В разбиении  $\{Г\}\{A\}\{C\}\{Н\}$  все игроки состоят в одноточечных коалициях и хотят отклониться, поэтому оно **не устойчиво по Нэшу** и **не контрактно-устойчиво**. И Аксинья, и Наталья хотели бы соединиться с Григорием, который не будет этому препятствовать, поэтому разбиение **не индивидуально-устойчиво**. Есть две сильно блокирующие коалиции –  $\{ГА\}$  и  $\{ГН\}$ , поэтому разбиение **не лежит в слабом ядре**.

## 2. Задача «об устойчивом браке»

Задачей «об устойчивом браке» называется обобщение задачи о разбиении игроков на пары.

Пусть множество игроков делится на два непересекающихся подмножества –  $M = \{m_1, \dots, m_k\}$  и  $N = \{n_1, \dots, n_l\}$ . Требуется разбить игроков по парам, в которых один из игроков принадлежит первому подмножеству, а другой – второму. У игроков есть предпочтения среди тех, кто будет с ними в паре. Может быть так, что некоторым парам игрок предпочитает одиночество. Каждый игрок может находиться одновременно в ограниченном количестве пар.

Таким образом, отношение предпочтения  $\geq$  для игрока  $m_i \in M$  задаётся на некотором множестве  $N_i \subseteq N$ , а для игрока  $n_j \in N$  – на множестве  $M_j \subseteq M$ .

Каждый игрок  $k \in M \cup N$  может принадлежать не более  $q_k$  парам.

### **Теорема Гейла-Шепли (1962)[8]**

*Для задачи «об устойчивом браке» со строгими предпочтениями игроков существует решение из ядра.*

Для доказательства этой теоремы был предложен алгоритм, который и получил название «алгоритм Гейла–Шепли». Он представлен ниже.

**Алгоритм 1.** *1 шаг. Каждый игрок  $m_i$  из множества  $M$  делает предложения  $q_{m_i}$  наиболее предпочтительным для него игрокам из множества  $N$ . Если игроку  $m_i$  не нравится никто из  $N$ , то он выходит из игры. Каждый игрок  $n_j$  из множества  $N$  рассматривает предложения игроков из  $M$  и говорит «может быть» не более  $q_{n_j}$  раз тем, предложение которых ему кажутся наиболее привлекательными. Если все сделанные ему предложения, для него менее предпочтительны чем одиночество, то он остается один.*

*2 шаг. Игроки множества  $M$ , не набравшие квоту, делают предложения наиболее предпочтительным для них игрокам из множества  $N$ , не набравшим квоту. Если для какого-то игрока  $m_i$  из  $M$  ни один игрок из  $N$  не предпочтительнее одиночества, то он не делает предложение никому. Игроки из множества  $N$  сравнивают новые предложения со старыми и говорят «может быть» наиболее предпочтительным игрокам, количество которых не превышает их квоту. Предложения, менее предпочтительные чем одиночество, игроки не рассматривают.*

*3 шаг. Шаг 2 повторяется до тех пор, пока остаются игроки из  $M$ , которые не набрали квоту и не предпочитают одиночество всем не набравшим квоту игрокам из  $N$ , и*

игроки из  $N$  со свободными местами, не предпочитающие одиночество всем игрокам из  $M$ , не набравшим квоту.

4 шаг. Все «помолвленные» пары сходятся, а все остальные остаются одиночками.

**Пример 2.1.** Игру примера 1.2 можно переформулировать в виде задачи «об устойчивом браке».

$$M = \{\Gamma, C\}, N = \{A, H\}.$$

$$N_{\Gamma} = \{A, H\} \quad A \dot{\succ}_{\Gamma} H$$

$$N_C = \{A\} \quad N_A = \{\Gamma\} \quad N_H = \{\Gamma\}$$

$$q_{\Gamma} = q_C = q_A = q_H = 1$$

1 шаг. Григорий и Степан делают предложение Аксинье. Аксинья говорит Григорию «может быть», и они становятся помолвлены.

2 шаг. Единственный непомолвленный мужчина – Степан. Для него одиночество предпочтительнее брака с Натальей и он выходит из игры.

3 шаг. Не осталось мужчин, которые хотели бы сделать предложение. Переходим к следующему шагу.

4 шаг. Григорий и Аксинья сходятся, Степан и Наталья остаются одни.

Таким образом, в результате алгоритма Гейла-Шепли мы получили разбиение  $\{\Gamma A\}\{C\}\{H\}$ .

Нетрудно заметить, что оно действительно лежит в ядре.

**Пример 2.2.** С помощью алгоритма Гейла-Шепли можно решить задачу распределения шестерых студентов некоторой кафедры по четырем научным руководителям.

Фамилии педагогов – Сорокин, Дроздов, Грачев и Чиждова.

Фамилии студентов – Петров, Федоров, Васильев, Назаров, Борисов и Алексеева.

Обозначим  $M = \{\Pi, \Phi, B, H, B, A\}$  и  $N = \{C, D, \Gamma, \Psi\}$ , как множество студентов и преподавателей кафедры.

Будем считать, что студент может быть прикреплен к одному научному руководителю, поэтому  $q_{\Pi} = q_{\Phi} = q_B = q_H = q_{\text{Б}} = q_A = 1$ .

Пусть, научный руководитель может иметь не более двух студентов.  $q_C = q_D = q_{\Gamma} = q_{\Psi} = 2$ .

Индивидуальные предпочтения игроков выглядят следующим образом:

$$N_{\Pi} = N_{\Phi} = N_B = N_H = N_{\text{Б}} = N_A = \{C, D, \Gamma, \Psi\}$$

$$C \succ_{\Pi} D \succ_{\Pi} \Gamma \succ_{\Pi} \Psi$$

$$C \succ_{\Phi} D \succ_{\Phi} \Psi \succ_{\Phi} \Gamma$$

$$C \succ_B \Psi \succ_B \Gamma \succ_B D$$

$Д >_H C >_H Ч >_H Г$

$Г >_B Д >_B C >_B Ч$

$Д >_A Г >_A Ч >_A C$

$M_C = \{П, Ф, В, Б, А\}$

$П >_C Н >_C В >_C А >_C Ф >_C Б >_C C$

$M_D = \{Ф, Б, П, В, Н\}$

$Ф >_D Б >_D П >_D В >_D Н$

$M_G = \{П, Ф, В\}$

$П >_G Ф >_G В$

$M_C = \{Ф, П, Н, В, Б, А\}$

$Ф >_C П >_C Н >_C В >_C Б >_C А >_C Ч$

*Реализуем алгоритм:*

*1 шаг. Петров, Федоров и Васильев делают предложение Сорокину, Назаров и Алексеева – Дроздову, Борисов – Грачеву. Сорокин говорит «может быть» Петрову и Васильеву и отказывает Федорову. Дроздов говорит «может быть» Назарову. Грачев отказывает Борисову, а Дроздов – Алексеевой.*

*2 шаг. Федоров и Борисов делают предложение Дроздову, Алексеева – Грачеву. Дроздов говорит «может быть» Федорову и Борисову и отказывает Назарову. Грачев отказывает Алексеевой.*

*3 шаг. Назаров делает предложение Сорокину, Алексеева – Чижевцу и те говорят «может быть», Сорокин отказывает Васильеву.*

*4 шаг. Васильев делает предложение Чижевцу и тот соглашается.*

*Студенты распределяются по научным руководителям следующим образом:*

$\{HC\}, \{AЧ\}, \{ФД\}, \{БД\}, \{ПC\}, \{BЧ\}$ .

### 3. Аддитивно-сепарабельные гедонистические игры

**Определение 3.1.** Аддитивно-сепарабельной гедонистической игрой (АСГИ) называется игра, в которой каждый игрок  $i$  оценивает то, насколько ему важно состоять в одной коалиции с игроком  $j \in N$  с помощью **веса игрока**  $v_i(j)$ , и то, насколько он хочет состоять в коалиции  $S \in N_i$  с помощью **веса коалиции**  $v_i(S) = \sum_{j \in S \setminus \{i\}} v_i(j)$ ,  $v_i(\{i\}) = 0$ ,  $S \succeq_i T \Leftrightarrow \sum_{j \in S \setminus \{i\}} v_i(j) \geq \sum_{j \in T \setminus \{i\}} v_i(j)$ . [2]

Нетрудно заметить, что АСГИ могут быть представлены в виде взвешенного ориентированного графа, в котором игроки представлены вершинами, а вес дуги  $(i, j)$  равен  $v_i(j)$ .

**Определение 3.2.** Аддитивно-сепарабельные предпочтения называются **симметричными**, если  $v_i(j) = v_j(i) = v_{ij}$  для всех  $i, j \in N$ .

АСГИ, все предпочтения которой симметричны, называется **симметричной АСГИ**.

**Утверждение 3.1.** Если для некоторой симметричной АСГИ разбиение  $\pi$  является решением задачи  $\sum_{ij \in \pi, ij \in S} v_{ij} \rightarrow \max$ , то это разбиение устойчиво по Нэшу. [4]

*Доказательство.* (От противного)

Пусть, для некоторой симметричной АСГИ  $(N, \succeq)$ , разбиение  $\pi$  является решением задачи  $\sum_{ij \in \pi, ij \in S} v_{ij} \rightarrow \max$  и не устойчиво по Нэшу.

Т.е.  $\exists k \in N, \exists T \in \pi \cup \emptyset: T \cup \{k\} \succ_k \pi(k)$ .

По определению АСГИ это означает, что  $\sum_{ij \in T \cup \{k\}} v_{ij} > \sum_{ij \in \pi(k)} v_{ij}$ .

$$\sum_{ij \in T \cup \{k\}} v_{ij} = \sum_{ij \in T} v_{ij} + \sum_{i \in T} v_{ik} + \sum_{j \in T} v_{kj} > \sum_{ij \in \pi(k)} v_{ij} = \sum_{ij \in \pi(k), i \neq k} v_{ij} + \sum_{j \in \pi(k)} v_{kj} + \sum_{i \in \pi(k)} v_{ik}$$

$$\sum_{ij \in \pi, ij \in S} v_{ij} =$$

$$\sum_{ij \in \pi, S \neq \pi(k), S \neq T, ij \in S} v_{ij} + \sum_{ij \in \pi(k)} v_{ij} + \sum_{ij \in T} v_{ij} =$$

$$\sum_{ij \in \pi, S \neq \pi(k), S \neq T, ij \in S} v_{ij} + \sum_{ij \in \pi(k) \setminus \{k\}} v_{ij} + \sum_{j \in \pi(k)} v_{kj} + \sum_{i \in \pi(k)} v_{ik} + \sum_{ij \in T} v_{ij} <$$

$$\sum_{ij \in \pi, S \neq \pi(k), S \neq T, ij \in S} v_{ij} + \sum_{ij \in \pi(k) \setminus \{k\}} v_{ij} + \sum_{j \in T} v_{kj} + \sum_{i \in T} v_{ik} + \sum_{ij \in T} v_{ij} =$$

$$\sum_{ij \in \pi, S \neq \pi(k), S \neq T, ij \in S} v_{ij} + \sum_{ij \in \pi(k) \setminus \{k\}} v_{ij} + \sum_{ij \in T \cup \{k\}} v_{ij}$$

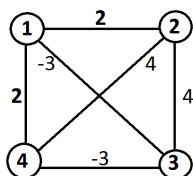
Получается, что на  $\pi$  сумма  $\sum_{ij \in \pi, ij \in S} v_{ij}$  не достигает максимума и это противоречит начальному условию.

Значит наше предположение было неверно и разбиение  $\pi$  устойчиво по Нэшу. □

**Замечание 3.1.** Для симметричной АСГИ всегда существует разбиение, устойчивое по Нэшу.



**Пример 3.1.** Рассмотрим симметричную АСГИ четырех игроков, порожденную следующим графом.



Решением задачи  $\sum_{ij: \exists S \in \pi, ij \in S} v_{ij} \rightarrow \max$  является разбиение  $\pi = \{1, 2, 4\} \{3\}$ .

Коалиционная структура  $\pi$  устойчива по Нэшу.

**Пример 3.2.** Известно, что усвояемость пищи зависит от сочетания входящих в нее компонентов, поэтому далеко не все продукты питания полезно употреблять вместе. Допустим, в магазине был куплен следующий набор продуктов: яйца, сыр, кабачки, хлеб, рис и банка варенья. Согласно исследованиям, яйца, сыр и варенье нельзя сочетать с хлебом и рисом. Кабачки плохо перевариваются с вареньем. Чтобы понять, в каких сочетаниях следует употреблять купленные продукты, составим симметричную АСГИ, в которой продукты будем считать игроками. Веса дуг, соединяющих «сочетаемые» продукты, назначим равными 1, а веса дуг соединяющих «несочетаемые» продукты будут -1.

$N = \{\text{яйца, сыр, кабачок, хлеб, рис, варенье}\}$

(Продукты будем обозначать по первым буквам названия.)

$$v_{яс} = 1, v_{як} = 1, v_{ях} = -1, v_{яр} = -1, v_{яв} = 1,$$

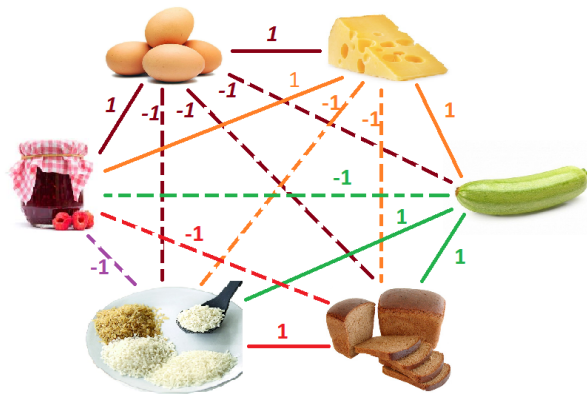
$$v_{ск} = 1, v_{сх} = -1, v_{ср} = -1, v_{св} = 1,$$

$$v_{кх} = 1, v_{кр} = 1, v_{кв} = -1,$$

$$v_{хр} = 1, v_{хв} = -1, v_{рв} = -1.$$

Чтобы составить правильные сочетания продуктов, нужно решить задачу

$$\sum_{ij: \exists S \in \pi, ij \in S} v_{ij} \rightarrow \max.$$



Получается, что разбиение  $\pi = \{\text{яйцо, сыр, варенье}\} \{\text{кабачок, хлеб, рис}\}$  устойчиво по

*Нэш. Такое разбиение продуктов наиболее полезно.*

## 4. Игры с общим порядком предпочтений

**Определение 4.1.** *Говорят, что гедонистическая игра  $(N, \succeq)$  имеет **общий порядок предпочтений**, если существует функция  $w : 2^N \setminus \{\emptyset\} \rightarrow \mathbb{R}$ , такая, что для любого игрока  $i$  и всех содержащих его коалиций  $C$  и  $D$  верно следующее:  $C \succeq_i D \Leftrightarrow w(C) \geq w(D)$ . [14]*

Для игры с общим порядком предпочтений, был найден алгоритм нахождения устойчивой по Нэшу коалиционной структуры из ядра [14].

**Алгоритм 2.** *(В  $\pi$  будем записывать коалиции. Множество игроков, коалиции которых еще не найдены, обозначим как  $R$ .)*

1.  $\pi := \emptyset$ ,  $R := N$ .

2. Пока  $R \neq \emptyset$  выполняем следующие операции.

-Выбираем подкоалицию  $T \subseteq R$  с наибольшим значением  $w(T)$ . Если таких коалиций несколько, то выбираем ту, в которой больше игроков. Среди коалиций одинакового размера можно выбирать любую.

-Добавляем  $T$  в  $\pi$ .

- $R := R \setminus T$ .

**Утверждение 4.1.** *Алгоритм 2 находит для гедонистической игры с общим порядком предпочтений индивидуально-устойчивую коалиционную структуру из ядра.*

*Доказательство.* В начале докажем, что полученное разбиение содержится в ядре.

(От противного)

Предположим, что разбиение  $\pi$ , полученное в результате работы алгоритма, не содержится в ядре.

Тогда существует коалиция  $S$ , такая что  $S \succ_i \pi(i)$  для всех игроков  $i \in S$ . Это означает, что  $w(S) > w(\pi(i))$  для всех игроков  $i \in S$ .

Рассмотрим игрока  $i' \in S$  такого, что  $w(\pi(i')) > w(\pi(i)) \forall i \in S$ . Значит, коалиция  $\pi(i')$  была выбрана раньше остальных коалиций  $\pi(i)$ , содержащих игроков из  $S$ , а значит  $\pi(i')$  – то подмножество  $\cup_{i \in S} \pi(i)$ , на котором функция  $w$  принимает наибольшее значение.  $w(S)$  не может быть больше  $w(\pi(i'))$ .

Значит, наше предположение было неверно, и разбиение  $\pi$  содержится в ядре.

Теперь докажем, что  $\pi$  индивидуально-устойчиво.

(От противного)

Допустим, что  $\pi$  не индивидуально-устойчиво.

Тогда, существует игрок  $i$  и коалиция  $S \in \pi : S \cup \{i\} \succ_i \pi(i)$ .

Т.е.  $w(S \cup \{i\}) > w(\pi(i))$ . Получается, что коалиция  $S$  была выбрана раньше чем  $\pi(i)$  и  $w(S) > w(S \cup \{i\}) > w(\pi(i))$ . Таким образом, игрокам из  $S$  не выгодно отклонение  $i$  и разбиение  $\pi$  индивидуально-устойчиво.

Предположение было неверно. □

**Пример 4.1.** Пусть в гедонистической игре  $(N, \succ)$  с общим порядком предпочтений  $N = \{1, 2, 3\}$  и функция  $w$  задана следующим образом:

$$w(\{1\}) = 3, w(\{2\}) = 4, w(\{3\}) = 2, w(\{12\}) = 4, w(\{13\}) = 4, w(\{23\}) = 1,$$

$$w(\{123\}) = 3.$$

Тогда общий порядок предпочтений игроков среди коалиций выглядит следующим образом:  $\{2\} \sim \{12\} \sim \{13\} > \{1\} \sim \{123\} > \{3\} > \{23\}$ .

Реализуем алгоритм.

Начальные данные:  $S = N \quad \pi = \emptyset$

1 шаг.  $T := \{12\} \quad S := S \setminus T = \{3\} \quad \pi := \{12\}$

2 шаг.  $T := \{3\} \quad S := S \setminus T = \emptyset \quad \pi := \{12\}\{3\}$

Нетрудно проверить, что коалиционная структура  $\{12\}\{3\}$ , полученная в результате реализации алгоритма, индивидуально-устойчива и находится в ядре.

## 5. $\mathcal{B}, \mathcal{W}$ и $\mathcal{TOP}$ -игры

### 5.1. Определение $\mathcal{B}$ и $\mathcal{W}$ -игр

Опишем игры, в которых предпочтения каждого игрока  $i$  среди коалиций  $S \in N_i$  однозначно определяется через список всех игроков  $N$ , упорядоченный по предпочтению для этого игрока  $i$ . В этом случае мы говорим, что на множестве игроков  $N$  задано отношение предпочтения  $\succeq$ .

Если для каждого игрока  $i \in N$  и каждой коалиции  $S \in N_i$  задать множество наиболее ( $\max_{\succeq_i}(S)$ ) или наименее ( $\min_{\succeq_i}(S)$ ) предпочтительных игроков из коалиции  $S$  для игрока  $i$ , то для каждого  $i \in N$  можно упорядочить по предпочтительности множество коалиций  $N_i$ .

**Определение 5.1.** *Игра называется  $\mathcal{B}$ -игрой, или игрой с  $\mathcal{B}$ -предпочтениями, если  $S \succeq_i T$  т.и т.т. когда выполнено одно из следующих условий:*

$$1) \forall s \in \max_{\succeq_i}(S \setminus \{i\}), \forall t \in \max_{\succeq_i}(T \setminus \{i\}), s \succ_i t$$

$$2) \forall s \in \max_{\succeq_i}(S \setminus \{i\}), \forall t \in \max_{\succeq_i}(T \setminus \{i\}), s \dot{=} t, |S| < |T|$$

**Определение 5.2.** *Игра называется  $\mathcal{W}$ -игрой, или игрой с  $\mathcal{W}$ -предпочтениями, если  $S \succ_i T$  т.и т.т. когда выполнено одно из следующих условий:*

$$1) \forall s \in \min_{\succeq_i}(S \setminus \{i\}), \forall t \in \min_{\succeq_i}(T \setminus \{i\}), s \succ_i t$$

$$2) \forall s \in \min_{\succeq_i}(S \setminus \{i\}), \forall t \in \min_{\succeq_i}(T \setminus \{i\}), s \dot{=} t, |S| < |T| \quad [2]$$

Считаем, что  $\max_{\succeq_i}(\{i\}) = \min_{\succeq_i}(\{i\}) = i$ .

**Замечание 5.1.** *В  $\mathcal{B}$  и  $\mathcal{W}$ -играх  $S \sim_i T$  т. и т.т. когда*

$$\forall s \in \min_{\succeq_i}(S \setminus \{i\}), \forall t \in \min_{\succeq_i}(T \setminus \{i\}), s \dot{=} t, |S| = |T|$$

**Пример 5.1.** *Рассмотрим игру пяти лиц  $N = \{H, L, \Phi, Z, T\}$ , для которых задан следующий список предпочтений:*

для  $H$ :  $L \succ_H \Phi \succ_H H \succ_H T \succ_H Z$

для  $L$ :  $\Phi \succ_L Z \succ_L H \succ_L L \succ_L T$

для  $\Phi$ :  $L \succ_\Phi H \succ_\Phi Z \succ_\Phi \Phi \succ_\Phi T$

для  $Z$ :  $T \succ_Z L \succ_Z \Phi \succ_Z Z \succ_Z H$

для  $T$ :  $Z \succ_T L \succ_T H \succ_T \Phi \succ_T T$

Если по нему построить  $\mathcal{B}$ -игру, то список предпочтений среди коалиций будет выглядеть следующим образом:

для  $H$   $\{ЛН\} >_H \{ЛФН\} \sim_H \{ЛЗН\} \sim_H \{ЛТН\} >_H \{ЛФЗН\} \sim_H \{ЛФТН\} \sim_H \{ЛЗТН\} >_H \{ЛФЗТН\} >_H \{ФН\} >_H \{ФЗН\} \sim_H \{ФТН\} >_H \{ФЗТН\} >_H \{H\} >_H \{ТН\} >_H \{ЗТН\} >_H \{ЗН\}$

для  $L$   $\{ФЛ\} >_L \{ФЛН\} \sim_L \{ФЛЗ\} \sim_L \{ФЛТ\} >_L \{ФЛНЗ\} \sim_L \{ФЛНТ\} \sim_L \{ФЛЗТ\} >_L \{ФЛНЗТ\} >_L \{ЗЛ\} >_L \{НЛЗ\} >_L \{ЗЛТ\} >_L \{ЗЛТН\} >_L \{НЛ\} >_L \{НЛТ\} >_L \{Л\} >_L \{ЛТ\}$

для  $\Phi$   $\{ЛФ\} >_\Phi \{ЛФН\} \sim_\Phi \{ЛФЗ\} \sim_\Phi \{ЛФТ\} >_\Phi \{ЛФНЗ\} \sim_\Phi \{ЛФНТ\} \sim_\Phi \{ЛФЗТ\} >_\Phi \{ЛФНЗТ\} >_\Phi \{НФ\} >_\Phi \{НФЗ\} >_\Phi \{НФТ\} >_\Phi \{НФЗТ\} >_\Phi \{ЗФ\} >_\Phi \{ЗФТ\} >_\Phi \{Ф\} >_\Phi \{ТФ\}$

для  $Z$   $\{ЗТ\} >_Z \{ТЗЛ\} \sim_Z \{ТЗН\} \sim_Z \{ТЗФ\} >_Z \{ТЗЛН\} \sim_Z \{ТЗЛФ\} \sim_Z \{ТЗНФ\} >_Z \{ТЗНФЛ\} >_Z \{ЛЗ\} >_Z \{ЛЗН\} \sim_Z \{ЛЗФ\} >_Z \{ЛЗФН\} >_Z \{ФЗ\} >_Z \{ФЗН\} >_Z \{З\} >_Z \{ЗН\}$

для  $T$   $\{ЗТ\} >_T \{ЗТЛ\} \sim_T \{ЗТН\} \sim_T \{ЗТФ\} >_T \{ЗТЛФ\} \sim_T \{ЗТЛН\} \sim_T \{ЗТНФ\} >_T \{ЗТНФЛ\} >_T \{ЛТ\} >_T \{ЛТФ\} \sim_T \{ЛТН\} >_T \{ЛТНФ\} >_T \{НТ\} >_T \{НТФ\} >_T \{ФТ\} >_T \{Т\}$

**Пример 5.2.** Если взять тот же список предпочтений среди игроков, что и в примере 5.1, но игру считать  $\mathcal{W}$ -игрой, то предпочтения игроков среди коалиций будут выглядеть следующим образом:

для  $H$   $\{ЛН\} >_H \{ФН\} >_H \{ФНЛ\} >_H \{H\} >_H \{ТН\} \sim_H \{ФТН\} \sim_H \{ЛТН\} >_H \{ЛФТН\} >_H \{ЗН\} >_H \{ЗНФ\} \sim_H \{ЗНЛ\} \sim_H \{ЗТН\} >_H \{ЗНФТ\} \sim_H \{ЗНЛТ\} \sim_H \{ЗНЛФ\} >_H \{ЗНЛФТ\}$

для  $L$   $\{ФЛ\} >_L \{ЗЛ\} >_L \{ЗЛФ\} >_L \{НЛ\} >_L \{НЛФ\} \sim_L \{ЗЛН\} >_L \{ЗЛНФ\} >_L \{Л\} >_L \{ТЛ\} >_L \{ТЛЗ\} \sim_L \{ТЛФ\} \sim_L \{ТЛН\} >_L \{ТЛНЗ\} \sim_L \{ТЛФЗ\} \sim_L \{ТЛФН\} >_L \{ТЛФНЗ\}$

для  $\Phi$   $\{ЛФ\} >_\Phi \{НФ\} >_\Phi \{НФЛ\} >_\Phi \{ЗФ\} >_\Phi \{ЗФН\} \sim_\Phi \{ЗФЛ\} >_\Phi \{ЗФЛН\} >_\Phi \{Ф\} >_\Phi \{ТФ\} >_\Phi \{ТФЗ\} \sim_\Phi \{ТФЛ\} \sim_\Phi \{ТФН\} >_\Phi \{ТФЛЗ\} \sim_\Phi \{ТФНЗ\} \sim_\Phi \{ТФНЛ\} >_\Phi \{ТФНЛЗ\}$

для  $\exists$   $\{T\exists\} \succ_{\exists} \{L\exists\} \succ_{\exists} \{L\exists T\} \succ_{\exists} \{\Phi\exists\} \succ_{\exists} \{\Phi\exists T\} \sim_{\exists} \{\Phi\exists L\} \succ_{\exists} \{\Phi\exists LT\} \succ_{\exists} \{\exists\} \succ_{\exists} \{\exists L\} \succ_{\exists} \{\exists L\Phi\} \sim_{\exists} \{\exists L T\} \succ_{\exists} \{\exists L T\Phi\} \sim_{\exists} \{\exists L LT\} \sim_{\exists} \{\exists L L\Phi\} \succ_{\exists} \{\exists L L\Phi T\}$

для  $T$   $\{T\exists\} \succ_T \{LT\} \succ_T \{LT\exists\} \succ_T \{HT\} \succ_T \{HT\exists\} \sim_T \{HTL\} \succ_T \{HTL\exists\} \succ_T \{\Phi T\} \succ_T \{\Phi T H\} \sim_T \{\Phi T\exists\} \sim_T \{\Phi T L\} \succ_T \{\Phi T H\exists\} \sim_T \{\Phi T L H\} \sim_T \{\Phi T L\exists\} \succ_T \{\Phi T L\exists H\} \succ_T \{T\}$

## 5.2. $TOP$ -игры

Обозначим как  $Ch(i, S)$ , ту подкоалицию коалиции  $S$ , которую игрок  $i$  предпочитает больше других.

Для обозначения наиболее предпочтительных подкоалиций коалиции  $S$  для игрока  $i$  введем обозначение  $Ch(i, S)$ :

$$Ch(i, S) := \{S' \subseteq S : (i \in S') \wedge (S' \succeq_i S'' \forall S'' \subseteq S)\}$$

Если  $|Ch(i, S)| = 1$ , то мы обозначаем единственное подмножество  $S$ , которое наиболее предпочтительно для  $i$ , как  $ch(i, S)$ .

**Определение 5.3.** *Игра  $(N, \succeq)$  называется  $TOP$ -игрой, если каждая коалиция имеет одну наиболее предпочтительную подкоалицию и для любого игрока  $i$  и любых двух коалиций  $S$  и  $T$ , его содержащих, верно, что:*

- 1)  $S \succ_i T$ , если  $ch(i, S) \succ_i ch(i, T)$
- 2)  $S \succ_i T$ , если  $ch(i, S) = ch(i, T)$ ,  $S \subset T$  [2]

Таким образом, считается, что чем меньше множество, содержащее «кумира», тем оно лучше.

Если  $(N, \succeq)$  –  $TOP$ -игра, то  $\forall S, T \subseteq N, i \in (S \cap T)$  верны следующие утверждения.

**Утверждение 5.1.** *Если  $S \succ_i T$ , то  $ch(i, S) \succeq_i ch(i, T)$ .*

*Доказательство.* (От противного) Пусть  $ch(i, T) \succ_i ch(i, S)$ . Тогда по определению  $TOP$ -игры  $T \succ_i S$ , что противоречит исходным данным. Таким образом, наше предположение было не верно.  $\square$

**Утверждение 5.2.** *Если  $T \subseteq N$ , то  $ch(i, S) \succeq_i ch(i, T) \forall i$*

**Утверждение 5.3.** *Если  $T \subset S, T \succeq_i S$  то  $ch(i, S) = ch(i, T)$ .*

*Доказательство.* (От противного) Пусть  $ch(i, S) \succ_i ch(i, T)$ . Тогда  $S \succ_i T$ , что противоречит исходным данным. Значит наше предположение было неверно и  $ch(i, S) \sim_i ch(i, T)$ , то есть  $ch(i, S) = ch(i, T)$ .  $\square$

**Утверждение 5.4.** *Если  $T \subseteq S, S \succeq_i T, ch(i, S) = ch(i, T)$ , то  $S = T$ .*

*Доказательство.* (От противного) Пусть  $T \subset S$ , тогда по определению  $TOP$ -игры  $T \succeq_i S$ , что противоречит начальным условиям. Значит наше предположение было неверно и  $S = T$ .  $\square$

**Замечание 5.2.** *Если  $\mathcal{B}$  или  $\mathcal{W}$ -игра является  $TOP$ -игрой, то для каждого игрока существует только один наиболее предпочтительный игрок.*

*Доказательство.* Допустим, что у игрока  $i \in N$  существует два наиболее предпочтительных игрока  $j$  и  $k$ .

$Ch(i, \{i, j, k\}) = \{\{i, j\}, \{i, k\}\}$ , значит  $|Ch(i, \{i, j, k\})| \neq 2$  и игра не является  $TOP$ -игрой, что противоречит нашему изначальному предположению.

Таким образом, если  $\mathcal{B}$  или  $\mathcal{W}$ -игра является  $TOP$ -игрой, то у каждого игрока существует только один наиболее предпочтительный игрок.  $\square$

**Замечание 5.3.** *Обратное утверждение не верно.*

*$\mathcal{B}$  или  $\mathcal{W}$ -игра, в которой каждый игрок имеет одного наиболее предпочтительного игрока, не всегда является  $TOP$ -игрой.*

**Пример 5.3.** *Рассмотрим  $\mathcal{B}$  - игру со следующими упорядоченными по предпочтению списками игроков:*

$$2 \succ_1 3 \doteq_1 4 \succ_1 1$$

$$3 \succ_2 4 \doteq_2 1 \succ_2 2$$

$$1 \succ_3 2 \doteq_3 4 \succ_3 3$$

$$3 \succ_4 2 \doteq_4 1 \succ_4 4$$

*В этом случае, предпочтения игроков среди коалиций выглядят следующим образом:*

$$\{12\} \succ_1 \{123\} \sim_1 \{124\} \succ_1 \{1234\} \succ_1 \{13\} \sim_1 \{14\} \succ_1 \{134\} \succ_1 \{1\}$$

$$\{23\} \succ_2 \{123\} \sim_2 \{234\} \succ_2 \{1234\} \succ_2 \{24\} \sim_2 \{12\} \succ_2 \{124\} \succ_2 \{2\}$$

$$\{13\} \succ_3 \{123\} \sim_3 \{134\} \succ_3 \{1234\} \succ_3 \{23\} \sim_3 \{34\} \succ_3 \{234\} \succ_3 \{3\}$$

$$\{34\} \succ_4 \{234\} \sim_4 \{134\} \succ_4 \{1234\} \succ_4 \{24\} \sim_4 \{14\} \succ_4 \{124\} \succ_4 \{4\}$$

$|Ch(1, \{134\})| = |\{13\}\{14\}| = 2$  и игра не является  $TOP$ -игрой.



**Утверждение 5.5.** Если в  $\mathcal{B}$  или  $\mathcal{W}$ -игре каждый игрок имеет строгий порядок предпочтений среди игроков, то  $|Ch(i, S)| = 1$  для любого игрока  $i$  и содержащей его коалиции  $S$ .

*Доказательство.* (От противного)

Пусть в некоторой  $\mathcal{B}$ -игре каждый игрок имеет строгий порядок предпочтений среди игроков, но для некоторого игрока  $i$  найдется коалиция  $S$  такая, что  $|Ch(i, S)| > 1$ .

Это означает, что в  $S$  найдется хотя бы два подмножества  $S_1$  и  $S_2$  такие что  $S_1, S_2 \in Ch(i, S)$ .

Так как  $S_1, S_2 \in Ch(i, S)$ , значит  $S_1 \sim_i S_2$ .

Из замечания мы знаем, что в игре две коалиции  $S_1$  и  $S_2$  могут быть одинаково предпочтительны для игрока  $i$ , только если

$$\forall s_1 \in \max_{\succeq_i}(S_1 \setminus \{i\}), \forall s_2 \in \max_{\succeq_i}(S_2 \setminus \{i\}), s_1 \sim_i s_2, |S_1| = |S_2|.$$

В нашей игре для  $i$  задан строгий порядок предпочтений среди игроков, значит

$$|\max_{\succeq_i}(S_1 \setminus \{i\})| = |\max_{\succeq_i}(S_2 \setminus \{i\})| = 1 \text{ и } \max_{\succeq_i}(S_1 \setminus \{i\}) = \max_{\succeq_i}(S_2 \setminus \{i\}) = \{s_{max}\}$$

Обозначим множество  $\{s_{max}, i\}$  как  $S_m$

Покажем, что  $S_1 = S_m$ .

Допустим, что это не так и  $S_m \subset S_1$ , а  $|S_m| < |S_1|$ .

Мы знаем, что  $\max_{\succeq_i}(S_1 \setminus \{i\}) = \{s_{max}\} = \max_{\succeq_i}(S_m \setminus \{i\})$ .

По определению  $\mathcal{B}$ -игры получается, что  $S_m \succ_i S_1$ , но этого быть не может, так как  $S_m \subset S_1 \subseteq S$ , а  $S_1 \in Ch(i, S)$ .

Значит наше предположение было неверно и  $S_1 = S_m$ .

Аналогично доказывается, что  $S_2 = S_m$ .

Таким образом, получается, что  $S_1 = S_m = S_2$  и предположение о том, что найдутся такие  $i$  и  $S$ , что  $|Ch(i, S)| > 1$  было неверно.

Доказательство для  $\mathcal{W}$ -игры полностью аналогично. □

**Утверждение 5.6.** Если в  $\mathcal{B}$ -игре каждый игрок имеет строгий порядок предпочтений среди игроков, то для любого игрока  $i$  и содержащей его коалиции  $S$  множества  $\max_{\succeq_i}(S \setminus \{i\})$  и  $\max_{\succeq_i}(ch(i, S) \setminus \{i\})$  равны.

*Доказательство.* Мы знаем, что  $ch(i, S) \succeq_i S$ .

Тогда по определению  $\mathcal{B}$ -игры выполняется одна из двух альтернатив:

$$1) \forall cs \in \max_{\succeq_i}(ch(i, S) \setminus \{i\}), \forall s \in \max_{\succeq_i}(S \setminus \{i\}) : cs \succ_i s,$$

$$2) \forall cs \in \max_{\succeq_i}(ch(i, S) \setminus \{i\}) \forall s \in \max_{\succeq_i}(S \setminus \{i\}) : s \dot{\succeq}_i cs, |Ch(i, S)| < |S|.$$

Первая альтернатива не может быть выполнена, так как  $\max_{\succeq_i}(ch(i, S) \setminus \{i\}) \subseteq ch(i, S) \subseteq S$

$S$ , значит всегда выполняется вторая.

Игрок  $i$  имеет строгий порядок предпочтений среди игроков, поэтому множества  $\max_{\geq_i}(S \setminus \{i\})$  и  $\max_{\geq_i}(ch(i, S) \setminus \{i\})$  одноточечные, а значит  $\max_{\geq_i}(S \setminus \{i\}) = \max_{\geq_i}(ch(i, S) \setminus \{i\})$ .  $\square$

**Утверждение 5.7.** *Если в  $\mathcal{W}$ -игре каждый игрок имеет строгий порядок предпочтений среди игроков, то для любого игрока  $i$  и содержащей его коалиции  $S$  множества  $\min_{\geq_i}(S \setminus \{i\})$  и  $\min_{\geq_i}(ch(i, S) \setminus \{i\})$  равны.*

*Доказательство.* Доказывается аналогично предыдущему утверждению.  $\square$

**Утверждение 5.8.** *Если в  $\mathcal{B}$  или  $\mathcal{W}$ -игре каждый игрок имеет строгий порядок предпочтений среди других игроков, то для любого игрока  $i$  и любых двух коалиций, которым он принадлежит, из того что  $ch(i, S) \succ_i ch(i, T)$  следует, что  $S \succ_i T$ .*

*Доказательство.* Возьмем  $i, S, T$  такие, что  $ch(i, S) \succ_i ch(i, T)$ .

По определению  $\mathcal{B}$ -игры получается, что выполняется одна из двух альтернатив:

$$1) \forall s \in \max_{\geq_i}(ch(i, S) \setminus \{i\}), \forall t \in \max_{\geq_i}(ch(i, T) \setminus \{i\}) : s \succ_i t$$

$$2) \forall s \in \max_{\geq_i}(ch(i, S) \setminus \{i\}) \forall t \in \max_{\geq_i}(ch(i, T) \setminus \{i\}) : s \dot{\simeq}_i t, |Ch(i, S)| < |Ch(i, T)|.$$

Из предыдущего замечания мы знаем, что  $\max_{\geq_i}(S \setminus \{i\}) = \max_{\geq_i}(ch(i, S) \setminus \{i\})$  и  $\max_{\geq_i}(T \setminus \{i\}) = \max_{\geq_i}(ch(i, T) \setminus \{i\})$ . Выполним замену множеств в 1), 2) и получим, что выполнена одна из альтернатив

$$1') \forall s \in \max_{\geq_i}(S \setminus \{i\}), \forall t \in \max_{\geq_i}(T \setminus \{i\}) : s \succ_i t$$

$$2') \forall s \in \max_{\geq_i}(S \setminus \{i\}) \forall t \in \max_{\geq_i}(T \setminus \{i\}) : s \dot{\simeq}_i t, |S| < |T|.$$

Это равносильно тому, что  $S \succ_i T$ .

Доказательство для  $\mathcal{W}$ -игры аналогично.  $\square$

**Утверждение 5.9.** *Если в  $\mathcal{B}$  или  $\mathcal{W}$ -игре каждый игрок имеет строгий порядок предпочтений среди других игроков, то для любого игрока  $i$  и любых двух коалиций  $S$  и  $T$ , которым он принадлежит, из того что  $ch(i, S) \sim_i ch(i, T)$  и  $S \subseteq T$  следует, что  $S \succ_i T$ .*

*Доказательство.* Рассмотрим  $i, S, T$  такие, что  $ch(i, S) \sim_i ch(i, T)$  и  $S \subseteq T$ .

Так как  $S \subseteq T$ , значит  $|S| < |T|$ .

Из Замечания 5.1 мы знаем, что  $ch(i, S) \sim_i ch(i, T)$  равносильно тому, что

$$\forall s \in \max_{\geq_i}(ch(i, S) \setminus \{i\}), \forall t \in \max_{\geq_i}(ch(i, T) \setminus \{i\}), s \succ_i t, |ch(i, S)| = |ch(i, T)|$$

Из Утверждения 6.6 мы знаем, что  $\max_{\geq_i}(S \setminus \{i\}) = \max_{\geq_i}(ch(i, S) \setminus \{i\})$  и  $\max_{\geq_i}(T \setminus \{i\}) =$

$max_{\geq_i}(ch(i, T) \setminus \{i\})$

Получается, что

$\forall s \in max_{\geq_i}(S \setminus \{i\}), \forall t \in max_{\geq_i}(T \setminus \{i\}), s \dot{=} t, |S| < |T|,$

Значит,  $S >_i T$ . □

**Следствие 5.1.** *В или  $\mathcal{W}$ -игра, в которой каждый игрок имеет строгий порядок предпочтений среди игроков является  $\mathcal{TOP}$ -игрой.*

### 5.3. Алгоритм «Top covering»

Опишем алгоритм нахождения разбиения из сильного ядра, для  $\mathcal{TOP}$ -игры.

Для каждого  $X \subseteq N$ , определим отношение достижимости  $\curvearrowright_X$ , как бинарное отношение на  $X \times X$ , где  $i \curvearrowright_X j$  т. и т.т., когда  $j \in ch(i, X)$ . В этом случае  $j$  называется соседом  $i$  в  $X$ . Коалицию  $CC(i, X)$ , достижимую игроком  $i$  в  $X$ , определим следующим образом:

$$CC(i, X) = \{k \in X : \exists j_1, \dots, j_l \in X : i = j_1 \curvearrowright_X \dots \curvearrowright_X j_l = k\}$$

**Алгоритм 3.** «Top covering»[2]

(В  $\pi$  будем записывать коалиции. Множество игроков, коалиции которых еще не найдены, обозначим как  $R$ .)

1.  $\pi := \emptyset, R := N$

2. Пока  $R \neq \emptyset$  выполняем следующие операции:

- Из множества  $R$  находим игрока  $i: CC(i, R) \leq CC(j, R) \forall j \in R$ .

- Добавляем в  $\pi$  найденное множество  $CC(i, R)$ .

-  $R := R \setminus CC(i, R)$ .

**Утверждение 5.10.** *Алгоритм «Top covering», примененный к  $\mathcal{TOP}$ -игре, дает разбиение из ядра.[5]*

*Доказательство.* Применим алгоритм к  $\mathcal{TOP}$ -игре. Множество  $\pi$ , полученное на выходе, является разбиением, так как состоит из непересекающихся множеств, объединение которых образует  $N$ .

Докажем, что разбиение  $\pi$  лежит в ядре.

(От противного)

Очевидно, что алгоритм циклический.

Множество тех игроков, среди которых ведется поиск  $k$ -ого элемента разбиения(во время  $k$ -го прохождения цикла) обозначим как  $R^k$ .

Будем обозначать как  $k(j)$  порядковый номер того цикла, на котором была найдена коалиция игрока  $j$ .

Пусть для разбиения  $\pi$ , полученного с помощью алгоритма, найдется блокирующая коалиция  $S$ , т.е.

$$S \succ_j \pi(j) \forall j \in S \quad (1)$$

Множество коалиций из разбиения  $\pi$ , которые содержат игроков из  $S$  обозначим как  $\pi_S$ .

Пусть коалиция  $\pi(j')$  игрока  $j' \in S$ , была найдена раньше остальных коалиций из  $\pi_S$ .  $k(j')$  будем обозначать как  $k'$ .

Тогда для любого  $j \in S$ , коалиция  $CC(j, R^{k(j)})$  содержится в  $R^{k'}$ .

$$CC(j, R^{k(j)}) \subseteq R^{k'} \forall j \in S \quad (2)$$

Так как для любого  $j$  коалиция  $\pi(j)$ , была выделена из множества  $R^{k(j)}$ , то

$$CC(j, R^{k(j)}) = \pi(j) \forall j \in S \quad (3)$$

Из 3 следует, что для всех игроков  $j$  из  $S$  верно, что  $ch(j, \pi(j)) = ch(CC(j, R^{k(j)})) = ch(i, R^{k(j)})$

$$ch(i, \pi(j)) = ch(i, R^{k(j)}) \forall j \in S \quad (4)$$

Коалиция  $S$ -блокирующая, поэтому  $S \succ_j \pi(j) \forall j \in S$ .

По Утверждению 6.1 получается, что

$$ch(j, S) \succeq_j ch(j, \pi(j)) = ch(j, R^{k(j)}) \forall j \in S \quad (5)$$

$$ch(j, S) \succeq_j ch(j, R^{k(j)}) \forall j \in S \quad (6)$$

Все игроки из  $S$  содержатся в  $R^{k'}$ , поэтому по Утверждению 6.1

$$ch(j', R^{k'}) \succeq_{j'} ch(j', S) \quad (7)$$

Из (7),(6) получаем, что  $ch(j', R^{k'}) \sim_{j'} ch(j', S)$ . У нас  $TOP$ -игра, поэтому

$$ch(j, R^{k'}) = ch(j, S) \forall j \in S \quad (8)$$

В частности,  $ch(j', R^{k'}) = ch(j', S)$ .

Докажем, что  $\pi(j') \subseteq S$

Пусть  $t \in \mathbb{N}$ . Для каждого игрока  $i$  и коалиции  $X$ , его содержащей, определим функцию  $C^t : N * 2^N \rightarrow 2^N$  следующим образом :

$$C^1(i, X) = ch(i, X)$$

$$C^{t+1}(i, X) = \cup_{i \in C^t(i, X)} ch(i, X)$$

Докажем по индукции, что  $C^t(j', R^{k'}) \subseteq S \quad \forall t \in \mathbb{N}$

База

$$C^1(j', R^{k'}) = ch(j', R^{k'}) = ch(j', S) \subseteq S$$

Переход

(Покажем, что если  $C^t(j', R^{k'}) \subseteq S$ , то и  $C^{t+1}(j', R^{k'}) \subseteq S$ )

Очевидно, что  $C^t(j', R^{k'}) \subseteq CC(j', R^{k'}) \forall t \in \mathbb{N}$

Поэтому для любого  $j$  из  $C^t(j', R^{k'})$  верно, что  $\pi(j) = \pi(j') = CC(j', R^{k'})$ .

Очевидно, что для любого игрока  $j$  из  $C^t(j', R^{k'})$  верно, что  $CC(j, R^{k'}) \subseteq CC(j', R^{k'})$ .

$$ch(j, R^{k'}) \subseteq CC(j, R^{k'}) \subseteq CC(j', R^{k'}) = \pi(j) \forall j \in C^t(j', R^{k'})$$

Получается, что

$$ch(j, R^{k'}) = ch(j, \pi(j)) \forall j \in C^t(j', R^{k'}) \quad (9)$$

Из (8),(9) получается, что  $ch(i, \pi(i)) = ch(i, S) = ch(j, R^{k'}) \forall i \in C^t(j', R^{k'})$

Тогда  $C^{t+1}(j', R^{k'}) = \cup_{i \in C^t(j', R^{k'})} ch(i, R^{k'}) \subseteq S$

$$CC(j', X^{k'}) = \cup_{t \in \mathbb{N}} C^t(j', R^{k'}) \Rightarrow \pi(j') = CC(j', X^{k'}) \subseteq S$$

Очевидно, что  $S = \pi(j')$  □

**Замечание 5.4.** Из предыдущего утверждения вытекает, что для игр обладающих высокой восприимчивостью существует разбиение из ядра.

**Пример 5.4.** (Пример реализации алгоритма для  $B$ -игры, в которой каждый из игроков имеет одного наиболее предпочтительного игрока)

Рассмотрим гедонистическую игру в которой  $N = \{1, 2, 3, 4\}$  и для каждого игрока существуют следующие предпочтения среди игроков:

$$2 \succ_1 3 \succ_1 4 \succ_1 1$$

$$3 \succ_2 4 \succ_2 1 \succ_2 2$$

$$1 \succ_3 2 \succ_3 4 \succ_3 3$$

$$3 \succ_4 2 \succ_4 1 \succ_4 4$$

Тогда предпочтения игроков на множестве коалиций выглядят следующим образом:

$$\begin{aligned}
& \{12\} \succ_1 \{123\} \sim_1 \{124\} \succ_1 \{1234\} \succ_1 \{13\} \succ_1 \{134\} \succ_1 \{14\} \succ_1 \{1\} \\
& \{23\} \succ_2 \{123\} \sim_2 \{234\} \succ_2 \{1234\} \succ_2 \{24\} \succ_2 \{124\} \succ_2 \{12\} \succ_2 \{2\} \\
& \{13\} \succ_3 \{123\} \sim_3 \{134\} \succ_3 \{1234\} \succ_3 \{23\} \succ_3 \{234\} \succ_3 \{34\} \succ_3 \{3\} \\
& \{34\} \succ_4 \{234\} \sim_4 \{134\} \succ_4 \{1234\} \succ_4 \{24\} \succ_4 \{124\} \succ_4 \{14\} \succ_4 \{4\}
\end{aligned}$$

Данная игра является *ТОР*-игрой. Реализуем алгоритм «Top covering».

Начальные данные:  $R = \{1234\}$ ,  $\pi = \emptyset$

1 шаг.

- $4 \succ_{\{1234\}} 3 \succ_{\{1234\}} 1 \succ_{\{1234\}} 2 \succ_{\{1234\}} 3$
- $S := \{123\}$
- $\pi := \{123\}$
- $R := \{1234\} \setminus \{123\} = \{4\}$

2 шаг.

- $CC(4, \{4\}) = 4$
- $S := \{4\}$
- $\pi : \{123\}\{4\}$
- $R := \{4\} \setminus \{4\} = \emptyset$

В итоге:  $\pi = \{123\}\{4\}$  – разбиение из ядра.

**Замечание 5.5.** Из Замечания 5.1 очевидно, что данный алгоритм нельзя применить к  $\mathcal{B}$ -игре, в которой хотя бы один из игроков имеет более одного наиболее предпочтительного игрока.

**Пример 5.5.** Рассмотрим игру  $(N, \succ)$ , где  $N = \{1, 2, 3, 4\}$  со следующим списком предпочтений среди игроков:

$$2 \doteq_1 3 \succ_1 4 \succ_1 1$$

$$3 \succ_2 4 \succ_2 1 \succ_2 2$$

$$1 \succ_3 2 \succ_3 4 \succ_3 3$$

$$3 \succ_4 2 \succ_4 1 \succ_4 4$$

Очевидно, что  $Ch(1, N) = \{12\}\{13\}$ ,  $Ch(2, N) = \{23\}$ ,  $Ch(3, N) = \{13\}$ ,  $Ch(4, N) = \{34\}$  и разбиение  $\{13\}\{24\}$  лежит в ядре, но данный алгоритм не применим к этой игре и найти его не может.

**Пример 5.6.** *Продemonстрируем работу алгоритма «Top covering» для игры примера 5.1.*

для Н: Л  $\succ_H$  Ф  $\succ_H$  Н  $\succ_H$  Т  $\succ_H$  З

для Л: Ф  $\succ_L$  З  $\succ_L$  Н  $\succ_L$  Л  $\succ_L$  Т

для Ф: Л  $\succ_F$  Н  $\succ_F$  З  $\succ_F$  Ф  $\succ_F$  Т

для З: Т  $\succ_Z$  Л  $\succ_Z$  Ф  $\succ_Z$  З  $\succ_Z$  Н

для Т: З  $\succ_T$  Л  $\succ_T$  Н  $\succ_T$  Ф  $\succ_T$  Т

Начальные данные:  $\pi = \emptyset$   $R = N$

1 шаг.

$-N \rightsquigarrow_{\{НЛФЗТ\}} Л \rightsquigarrow_{\{НЛФЗТ\}} Ф \rightsquigarrow_{\{НЛФЗТ\}} Л \quad З \rightsquigarrow_{\{НЛФЗТ\}} Т \rightsquigarrow_{\{НДФЗТ\}} З$

$-S := \{ТЗ\}$

$-\pi := \{ТЗ\}$

$-R := R \setminus \{ТЗ\} = \{НЛФ\}$

2 шаг.

$-N \rightsquigarrow_{\{НЛФ\}} Л \rightsquigarrow_{\{НЛФ\}} Ф \rightsquigarrow_{\{НЛФ\}} Л$

$-S := \{ЛФ\}$

$-\pi := \{ТЗ\} \cup \{ЛФ\}$

$-R := R \setminus \{ТЗ\} = \{Н\}$

3 шаг.

$-S := \{Н\}$

$-\pi := \{ТЗ\} \cup \{ЛФ\} \cup \{Н\}$

$-R := R \setminus \{Н\} = \emptyset$

Итог: разбиение  $\pi = \{ТЗ\} \cup \{ЛФ\} \cup \{Н\}$  находится в ядре.

## Заключение

В работе были изучены принципы устойчивости для коалиционных структур в гедонистических играх, предъявлены и продемонстрированы с помощью примеров некоторые модели игр, для которых есть алгоритм нахождения устойчивого решения.

Изучив уже опубликованные исследования в области теории гедонистических игр, приходится сделать вывод, что она разработана не достаточно широко. Обзорная статья, послужившая основой для данной работы, ссылается на более чем пятьдесят источников, но большинство из них являются пересказом друг друга. Известно очень мало моделей игр, для которых существует разбиение, обладающее достаточно сильной устойчивостью. Хочется верить, что новые исследования в этой области принесут практически-важные результаты.



# Литература

- [1] Alcalde, J., and Revilla, P., Researching with whom? Stability and manipulation. *Journal of Mathematical Economics*, 40(8), 869-887, 2004.
- [2] Aziz, H. and Savani, R., Hedonic Games. In: F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia (eds.), *Handbook of Computational Social Choice*. Cambridge University Press, 2015.
- [3] Banerjee, S., H. Konishi, and T. Sönmez, Core in a simple coalition formation game, *Social Choice and Welfare* 18, 135-153, 2001.
- [4] Bogomolnaia, A. and Jackson, M. O., The Stability of Hedonic Coalition Structures., *Games and Economic Behavior*, 38(2), 201-230, 2002.
- [5] Dimitrov, D. and Sung, S. C., Top responsiveness and Nash stability in coalition formation games., *Kybernetika*, 2006.
- [6] Dimitrov, D. and Sung, S. C., Top responsiveness and stable partitions in coalition formation games., *Universität Bielefeld*, 2005.
- [7] Dréze, J.H. and Greenberg, J., Hedonic coalitions: optimality and stability, *Econometrica*, 48(4), 987-1003, 1980.
- [8] Gale, D., and Shapley, L.S., College Admissions and the Stability of Marriage., *The American Mathematical Monthly*, 1962.
- [9] Hajduková, J., Coalition formation games: A survey. *International Game Theory Review*, 8(4), 613-641, 2006.
- [10] Roth, A.E., The Economics of Matching: Stability and Incentives, *Mathematics of Operations Research* 7, 617-628, 1982.

- [11] Roth, A.E., Sotomayor, M., Two-sided Matching: A Study in Game-Theoretic Modeling and Analysis. New York: Cambridge University Press, 1990.
- [12] Roth, A.E., The Economist as Engineer: Game Theory, Experimentation, and Computation as Tools for Design Economics, *Econometrica.*, 70(4), 1341–1378, 2003.
- [13] Roth, A.E., Sotomayor, M.A.O., Two-sided Matching: A Study in Game-theoretic Modeling and Analysis, Cambridge, N. Y.: Cambridge University Press, 1992.
- [14] Suksompong, W., Individual and Group Stability in Neutral Restrictions of Hedonic Games., Department of Computer Science, Stanford University.
- [15] Sung, S.C., and Dimitrov, D. 2007a. On core membership testing for hedonic coalition formation games. *Operations Research Letters*, 35(2), 155-158.
- [16] Sung, S. C., and Dimitrov, D. 2010. Computational Complexity in Additive Hedonic Games. *European Journal of Operational Research*, 203(3), 635-639.
- [17] Sung, S. C., and Dimitrov, Dinko. 2007b. On Myopic Stability Concepts for Hedonic Games. *Theory and Decision*, 62(1), 31-45.
- [18] Алескеров Ф.Т., Кисельгоф С.Г., Лауреаты Нобелевской премии – 2012: Ллойд Шепли и Элвин Рот., *Экономический журнал ВШЭ.*, 4, 2012.
- [19] Железова Е., Измалков С., Сонин К., Хованская И., Теория и практика двусторонних рынков., *Вопросы экономики.*, 1, 2013.
- [20] Коджаспирова Г. М., Коджаспиров А. Ю., Педагогический словарь: Для студентов высших и средних педагогических учебных заведений., Издат.центр: «Академия», 2001.

# Приложение

В данном приложении содержится текст программы, реализующей алгоритм «Top covering» для  $B$ -игры на языке  $C\#$ . Программа состоит из основного класса Program, и двух вспомогательных классов NewGame и ForList.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using ConsoleApplication1;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Здравствуйте!");
            Console.WriteLine("Данная программа реализует алгоритм Top Covering,");
            Console.WriteLine("который генерирует разбиение из ядра для B-игры со строгим списком предпочтений.");
            Console.WriteLine("Для создания игры Вам будет нужно указать ");
            Console.WriteLine("    количество игроков-участников");
            Console.WriteLine("    предпочтения всех игроков");
            Console.WriteLine();
            Boolean start = true;
            while (start)
            {
                string yes;
                Console.WriteLine("Хотите создать игру?");
                Console.WriteLine("1-создать игру на консоле, 2-игра из файла, 0-выйти");
                yes = Console.ReadLine();
                switch (yes)
                {
                    case "0":
                        start = false;
                        Console.WriteLine("Press any key to exit.");
                        Console.ReadKey();
                        break;
                    case "1":
                        NewGame.NGame();
                        break;
                    case "2":
                        NewGame.NGame2();
                        break;
                    default:
                        Console.WriteLine("Вы неправильно ввели данные");
                        break;
                }
            }
            Console.ReadLine();
        }
    }
}
```

```

    }
}

}

namespace ConsoleApplication1
{
    public static class NewGame
    {
        //класс для считывания количества игроков при введении игры из консоли
        public static void haveN(out int N)
        {
            N = 0;
            Boolean hN = false;
            while (hN == false)
            {
                N = 0;
                Console.WriteLine("Введите количество игроков");
                string sn= Console.ReadLine();
                int i = 0;
                while (i < sn.Length)
                {
                    if (sn[i] >= 48 && sn[i] < 58)
                    {
                        N = N * 10 + sn[i] - 48; hN = true;
                    }
                    else
                    {
                        hN = false;
                        Console.WriteLine("Вы не правильно ввели число");
                        break;
                    }
                    i++;
                }
            }
        }

        //считывание игры из текстового файла
        public static void NGame2()
        {
            Console.WriteLine("Введите название текстового файла.");
            Console.WriteLine("Каждая строка файла -- ПОЛНЫЙ СТРОГИЙ список предпочтения игрока.");
            Console.WriteLine("Игроки списка должны быть разделены пробелом.");
            string namefile=Console.ReadLine();
            string[] lines = System.IO.File.ReadAllLines(@namefile);
            int N=lines.Length;
            Boolean[] incoal = new Boolean[N];
            int[] Coal = new int[N];
            for (int i = 0; i < N; i++)
                Coal[i] = 0;
        }
    }
}

```

```

List<int>[] P = new List<int>[N];

string S;

for (int i = 0; i < N; i++)
{
    Boolean d = true;
    do
    {
        S = lines[i];
        P[i] = null;
        ForList.StringToList(N, S, out P[i], out d);
    } while (d == false);
}
int nk = 0;
int inCoal = 0;
while (N-inCoal > 0)
{
    int[,] C = new int[N, N];

    ForList.GenCC(N, P, Coal, out C);
    int min, imin;
    ForList.MinCC(N, C, Coal, out min, out imin);
    nk++;
    for (int i = 0; i < N; i++)
        if (C[imin - 1, i] == 1)
        {
            Coal[i] = nk;
            for (int j1 = 0; j1 < N; j1++)
                P[j1].Remove(i + 1);
        }
    inCoal = inCoal + min;
}
Console.Write("Разбиение ");
for (int j = 1; j <= nk; j++)
{
    Console.Write("{");
    for (int i = 0; i < N; i++)
        if (Coal[i] == j)
            Console.Write(i+1);
    Console.Write("}");
}
Console.WriteLine(" находится в ядре");
}
//считывание игры с консоли
public static void NGame()
{
    int N;
    haveN(out N);
    Boolean[] incoal = new Boolean[N];
    int[] Coal = new int[N];

```

```

for (int i = 0; i < N; i++)
    Coal[i] = 0;
List<int>[] P = new List<int>[N];
string S;
Console.WriteLine("Вводите список игроков ЧЕРЕЗ ПРОБЕЛ в порядке УБЫВАНИЯ предпочтительности");
Console.WriteLine();
for (int i = 0; i < N; i++)
{
    Boolean d = true;
    do
    {
        Console.WriteLine(" для игрока {0}", i + 1);
        S = Console.ReadLine();
        P[i] = null;
        ForList.StringToList(N, S, out P[i], out d);
    } while (d == false);
}
int nk = 0;
int inCoal = 0;
while (N - inCoal > 0)
{
    int[,] C = new int[N, N];
    ForList.GenCC(N, P, Coal, out C);
    int min, imin;
    ForList.MinCC(N, C, Coal, out min, out imin);
    nk++;
    for (int i = 0; i < N; i++)
        if (C[imin - 1, i] == 1)
        {
            Coal[i] = nk;
            for (int j1 = 0; j1 < N; j1++)
                P[j1].Remove(i + 1);
        }

    inCoal = inCoal + min;
}
Console.Write("Разбиение ");
for (int j = 1; j <= nk; j++)
{
    Console.Write("{");
    for (int i = 0; i < N; i++)
        if (Coal[i] == j)
            Console.Write(i + 1);
    Console.Write("}");
}
Console.WriteLine(" находится в ядре");
}

}

}
namespace ConsoleApplication1

```

```

{
    public class ForList
    {

        public static void GenCoalList(int N, List<int>[] P,out List<int[]>[] Coal )
        {
            Coal = new List<int[]>[N];
            Boolean[] notfull = new Boolean[N];
            for (int i = 0; i < N; i++)
                notfull[i] = true;
            for (int i = 0; i < N; i++)
            {
                for (int j = 1; j <= N; j++)
                {
                    Coal[i][j] = new int[N];
                }

            }

        }
    }

    //преобразование введенных данных в списки предпочтений игроков
    public static void StringToList(int N, string ss,out List<int> list,out Boolean d)
    {
        string s = ss;
        while (s[0] == ' ')
            s = s.Substring(1);
        list = new List<int>(N);
        d = true;
        string els = "";
        Boolean[] P = new Boolean[N];
        for (int i = 0; i < N; i++)
        {
            P[i] = false;
        }
        Boolean elYes = true;
        for (int i = 0; i < s.Length; i++)
        {
            if ((s[i] < 48 || s[i] > 57) && s[i] != ' ')
            {
                Console.WriteLine("Данные введены не правильно");
                elYes = false;
                break;
            }
            else
            {
                if (s[i] != ' ')
                {
                    els += s[i];
                    if (Convert.ToInt32(els) > N)
                    {
                        Console.WriteLine("Данные введены не правильно");
                        elYes = false;
                    }
                }
            }
        }
    }
}

```

```

        }

    }
    else
    {
        int e1 = Convert.ToInt32(els);
        if (P[e1-1])
        {
            d = false;
        }
        else
        {
            P[e1-1] = true;
            list.Add(e1);
        }
        els = "";
    }
}
if (elYes==false)
{
    break;
}

}
if (elYes)
{
    if (els != "")
    {
        int e1 = Convert.ToInt32(els);
        if (P[e1-1])
        {
            d = false;
        }
        else
        {
            P[e1-1] = true;
            list.Add(e1);
        }
    }

    for (int i = 0; i < N; i++)
    {
        if (P[i] == false)
        {
            d = false;
        }
    }
}
else d = false;
}

```



```

//генерация CC
public static void GenCC(int N,List <int>[] P,int[] coal, out int[,] C )
{
    int j;
    C = new int[N, N];
    int i;
    for (i = 0; i < N; i++)
    {
        for (j = 0; j < N; j++)
        {
            C[i, j] = 0;
        }
    }

    for (i = 0; i < coal.Length; i++)
    {
        C[i, i] = 1;
        j = P[i][0] - 1;
        while (C[i, j] == 0)
        {
            C[i, j] = 1;
            j = P[j][0] - 1;
        }
    }
}

//нахождение минимальной CC
public static void MinCC(int N,int[,] C,int[] coal,out int min,out int imin)
{
    min = N; imin = 0;
    int m = 0;
    for (int i = 0; i < N; i++)
    {
        if (coal[i] == 0)
        {
            for (int j = 0; j < N; j++)
            {
                m += C[i, j];
            }
            if (m < min)
            {
                imin = i; min = m;
            }
            m = 0;
        }
    }
    imin++;
}
}
}
}

```

