FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE

Zadání bakalářské práce

Název: Stabilita Hedonických her se strukturovanými preferencemi
Student: Jan Šmolík
Vedoucí: RNDr. Dušan Knop, Ph.D.
Studijní program: Informatika
Obor / specializace: Teoretická informatika
Katedra: Katedra teoretické informatiky
Platnost zadání: do konce letního semestru 2021/2022

Pokyny pro vypracování

Hedonické hry jsou známým a značně rozšířeným konceptem tvorby koalic. Nedávno Boehmer a Elkind zavedli koncept profilů založený na takzvaných podmínkách různodorodosti -- ve stručnosti lze říci, že agenti jako takoví ztrácejí svou identitu (jedná se tedy o anonymní hedonické hry) ale mají nějaké typy (například červený či modrý); preference agentů jsou pak založeny na počtu agentů stejného typu.Cílem práce bude prozkoumat různé koncepty stability pro výše popsanou tvorbu koalic (budeme uvažovat pevnou velikost kolic například 3 či 4) pro stukturované profily. Známe koncepty struktury profilů zahrnují, mimo jiné, tzv. Single-Peaked či Single-Crossing známé kupříkladu z teorie volebních systémů. Budeme zkoumat zda na základě těchto strukturálních omezení zle garantovat existenci stabilního rozdělení agentů a jaký mají tato omezení vliv na výpočetní složitost nalezení takového rozdělení agentů.

–

- Niclas Boehmer, Edith Elkind: Individual-Based Stability in Hedonic Diversity Games. AAAI 2020: 1822-1829- Niclas Boehmer, Edith Elkind: Stable Roommate Problem With Diversity Preferences. AAMAS 2020: 1780-1782
- Robert Bredereck, Jiehua Chen, Ugo Paavo Finnendahl, Rolf Niedermeier: Stable roommates with narcissistic, single-peaked, and single-crossing preferences. Auton. Agents Multi Agent Syst. 34(2): 53 (2020)
- David F. Manlove: Algorithmics of Matching Under Preferences. Series on Theoretical Computer Science 2, WorldScientific 2013, ISBN 978-981-4425-24-7, pp. 1-524

Elektronicky schválil/a doc. Ing. Jan Janoušek, Ph.D. dne 13. února 2021 v Praze.

Bakalářská práce

STABILITA HEDONICKÝCH HER SE STRUKTUROVANÝMI PREFERENCEMI

Jan Šmolík

Fakulta informačních technologií ČVUT v Praze
Katedra teoretické informatiky
Vedoucí: RNDr. Dušan Knop, Ph.D.
13. května 2021

Odkaz na tuto práci: Jan Šmolík. *Stabilita Hedonických her se strukturovanými preferencemi.* Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií, 2021.

Obsah

Seznam obrázků

Seznam tabulek

Chtěl bych moc poděkovat vedoucímu této práce Dušanu Knopovi za veškerou pomoc, všechny rady a všechny konzultace. Nemohl jsem si přát lepšího vedoucícho. Pod jeho vedením mě bavilo pracovat a myslím, že jsem se toho i hodně naučil. Dále bych chtěl poděkovat své rodině a také kamarádům, se kterými mám tu čest studovat na této škole.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené.

V Praze dne 8. května 2021 ....

Abstrakt

Problém stabilních spolubydlících s podmínkami různorodosti spočívá v rozdělení agentů dvou typů do koalic pevné velikosti na základě jejich preferencí, které závisí pouze na počtu agentů stejného typu. V této práci se zabýváme stabilitou takových rozdělení v závislosti na velikosti koalice a struktuře preferenčních profilů agentů. Snažíme se uchopit problém v celé jeho šíři a vytvořit nadhled nad celou problematikou představením Hedonických her a jejich zajímavých podtříd. Představujeme nový randomizovaný algoritmus na rychlé hledání core stabilních řešení instancí tohoto problému a ukazujeme, jakých výsledků jsme pomocí algoritmu dosáhli. Nalezli jsme malou single-peaked instanci s prázdným core a ověřili jsme, že každá instance tohoto problému, kde

- velikost koalic = 3, preferenční relace všech agentů jsou single-peaked a počet agentů ≤ 30,
- velikost koalic = 3 a počet agentů ≤ 15,
- velikost koalic = 4, preferenční relace všech agentů jsou single-peaked a počet agentů = 8,

má core stabilní řešení. Předkládáme argumenty, proč se domníváme, že každá instance, kde

- velikost koalic = 3,
- velikost koalic = 4 a preferenční relace všech agentů jsou single-peaked,

má core stabilní řešení.

Klíčová slova Problém stabilních spolubydlících s podmínkami různorodosti, Hedonické hry, Core stabilita, Podmínky různorodosti, Single-peaked preferenční profily, Polynomiální algoritmy, NP-úplnost

Abstract

In the Roommate diversity problem, agents that belong to one of the two types have to be allocated to fixed size rooms based on their preferences, which depend solely on the fraction of agents of their own type among their potential roommates. In this work we study the stability of outcomes with respect to the room size and structured preferences. We provide a new randomized algorithm for finding core stable outcomes and show the results which we have obtained with the algorithm. We have found a small single-peaked instance with empty core and have verified, that every instance of the Roommate diversity problem with

- room size = 3, all preferences single peaked and number of agents ≤ 30,
- room size = 3 and number of agents ≤ 15,
- room size = 4, all preferences single peaked and number of agents = 30,

admits an outcome that is core stable. We present arguments why we believe that every instance of this problem with

- room size $= 3$,
- room size $= 4$ and all preferences single peaked,

admits an outcome that is core stable.

Keywords Roommate Diversity Problem, Hedonic Games, Core stability, Diversity preferences, Single-peaked preferences, Polynomial-time algorithms, NP-completeness

Úvod

Teorie sociální volby je teoretický vědní obor zabývající se preferencemi, zájmy a názory jednotlivců, na základě kterých lze poté dosáhnout určitých kolektivních rozhodnutí. Nachází široké uplatnění napříč mnoha obory, například v teorii volebních systémů nebo v ekonomii blahobytu, která se zabývá různými problémy spojenými s dosažením maximálního ekonomického a sociálního blahobytu, a napříč staletími přitahovala některé z nejbystřejších myslitelů.

Výpočetní sociální volba je oproti tomu však velmi moderní odvětví vědy, které je v průniku klasických vědních oborů, jako jsou sociologie, ekonomie (teorie her) a teoretická informatika [4]. Nezabývá se pouze studiem paradigmat teorie sociální volby, ale snaží se také o návrh a prozkoumání složitosti různých algoritmů. Pro mnoho problémů jsou dnes algoritmy známé. Mnoho problémů je však tak těžkých, že ani neočekáváme existenci efektivního (polynomiálního) algoritmu. To může někdy přijít vhod – výpočetní složitost může například posloužit i jako bariéra proti strategickému zmanipulování voleb.

Koalice jsou významnou součástí ekonomického, politického, ale i sociálního života. Výpočetní sociální volba se jimi zabývá, podrobně studuje vytváření koalic na základě různých pravidel a preferencí a prozkoumává kvalitu vytvořených koalic podle různých kritérií. Jedním z nejrozšířenějších konceptů tvorby koalic jsou Hedonické hry, kterými se tato práce zabývá.

V Hedonických hrách vystupují tzv. agenti, což je univerzální označení pro účastníky těchto her (mohou to být lidé, roboti,...). Ti mají preference ohledně různých koalic, které mohou vytvořit společně s jinými agenty. Na základě těchto preferencí potom hledáme nějaká zajímavá rozdělení agentů, která jsou lepší než ostatní.

Můžeme se například snažit, aby co největší počet agentů byl umístěn do koalice, kterou preferuje nejvíce. Nebo obráceně, můžeme se snažit minimalizovat počet agentů, kteří jsou v koalici, kterou preferují nejméně. Tato práce se však zabývá především stabilitou takových rozdělení. Velmi neformálně řečeno, zajímají nás taková rozdělení do koalic, ve kterých neexistuje žádná skupinka agentů, kteří by raději vytvořili společně novou koalici, než aby byli ve své současné. Tomuto konceptu stability se říká core stabilita – existují však i jiné, a některé z nich si ukážeme.

Pomocí těchto her lze namodelovat mnoho zajímavých situací, kupříkladu plánování skupinových aktivit [8], formování výzkumných týmů [1], sestavování vládnoucí koalice [7] nebo distribuované přidělování úkolů pro bezdrátové agenty [13].

V roce 2020 N. Boehmer a E. Elkind publikovali článek *Stable Roommates Problem with Diversity Preferences* [3], ve kterém zavedli nový typ těchto her. Agenti mohou být jednoho ze dvou typů (např. červení a modří, učitelé a žáci, ženy a muži), smějí vyvářet pouze koalice předem dané velikosti a jejich preference jsou založeny na počtu agentů daného typu v koalici.

Tento problém je hlavním předmětem této práce. Pomocí tohoto modelu lze také zachytit mnoho zajímavých situací z reálného světa, například sdílení pokojů, rozdělování studentů do týmů na školní projekt (chlapci/dívky, česky/anglicky mluvící) nebo usazení ke stolům dané velikosti.

Stuktura práce

V Kapitole 1 si zadefinujeme obecnou Hedonickou hru. Dále se seznámíme s Problémem stabilních spolubydlících, Hedonickou hrou s pevnou velikostí koalic, Problémem stabilních manželství a Hedonickou hrou s podmínkami různorodosti.

V Kapitole 2 se blíže podíváme na Problém stabilních spolubydlících s podmínkami různorodosti (dále Roommate diversity problem), který je hlavním předmětem naší práce a srhneme si vše významné, co o tomto problému již víme.

V Kapitole 3 si představíme nový randomizovaný algoritmus na rychlé hledání core stabilního řešení pro instance Roommate diversity problému.

V Kapitole 4 si shrneme všechny nové poznatky, které jsme mohli o Roommate diversity problému i díky novému algoritmu učinit.

A nakonec v Kapitole 5 se blíže podíváme na implementaci celé této problematiky i algoritmu, tak jak se nachází v přiložených souborech.

Kapitola 1

Hedonické hry

V této kapitole si nejprve zadefinujeme obecnou Hedonickou hru, poté si ukážeme nějaké nejznámnější podtřídy Hedonických her. Začneme tedy s tím nejjednodušším modelem, postupně budeme přidávat na složitosti a propracovávat se až k hlavnímu tématu této práce – Roommate diversity problému, kterému jsou věnovány následující kapitoly. Cílem této kapitoly je vytvořit čtenáři jakýsi nadhled nad celou touto problematikou.

1.1 Hedonická hra

Nechť $N = \{1, 2, \ldots, |N|\}$ je množina agentů. Každou neprázdnou podmnožinu $S \subseteq N$ nazveme *koalicí*. *Rozdělením do koalic* nebo *řešením* (anglicky *coalition partition* nebo *outcome*) nazveme množinu $\Pi = \{S_k\}_{k=1}^{K}$ koalic, která je rozkladem množiny N, tedy $\bigcup_{k=1}^{K} S_k = N$ a $S_k \cap S_\ell = \emptyset$ pro všechna $1 \leq k < \ell \leq K$.

Preferenční profil agenta $i \in N$ je preferenční relace $\succsim_i$ nad množinou $\{S \subseteq N : i \in S\}$, což jsou koalice, které obsahují daného agenta i. Speciální značení $\succ_i$ se používá pro striktní preferenci, $\sim_i$ označuje lhostejnost. Pro dané rozdělení do koalic Π a daného agenta i, $\pi(i)$ označuje koalici $S \in \Pi$ takovou, že $i \in S$.

▶ **Definice 1.1.** Hedonická hra *je dvojice* $(N, (\succsim_i)_{i \in N})$*, kde* $N = \{1, 2, \ldots, |N|\}$ *je množina agentů a* $\succsim_i$ *každého agenta* $i \in N$ *je preferenční relace nad množinou* $\{S \subseteq N : i \in S\}$ *koalic, které obsahují agenta* i*.*

▶ **Definice 1.2.** *Řekneme, že koalice* T blokuje *rozdělení* Π*, pokud* $T \succ_i \pi(i)$ *pro všechny* $i \in T$*.*

▶ **Definice 1.3.** *Rozdělení do koalic* Π *je* core stabilní *ve hře* $(N, (\succsim_i)_{i \in N})$ *právě tehdy, když žádná koalice* $T \subseteq N$ *není blokující. Jako* core *dané hry se potom označuje množina všech core stabilních rozdělení.*

Jinak řečeno, rozdělení do koalic je core stabilní, pokud neexistuje koalice $T \subseteq N$, jejíž všichni členové ji striktně preferují před svou aktuální koalicí.

▶ **Poznámka 1.4.** Toto není jediný koncept stability - v dalších částech této práce si ukážeme ještě exchange stabilitu, závidění prostost a Pareto optimalitu.

▶ **Příklad 1.5.** Pro ilustrování nám poslouží jednoduchý příklad se třemi agenty $N = \{1, 2, 3\}$:

$$\begin{aligned}
1 &: \{1,3\} \succ_1 \{1,2,3\} \succ_1 \{1\} \succ_1 \{1,2\},\\
2 &: \{1,2\} \succ_2 \{1,2,3\} \succ_2 \{2\} \succ_2 \{2,3\},\\
3 &: \{1,3\} \succ_3 \{1,2,3\} \succ_3 \{3\} \succ_3 \{2,3\}.
\end{aligned}$$

V tomto příkladu stojí za povšimnutí, že jsou preferenční profily všech agentů striktní: žádný agent není lhostejný mezi dvěma různými koalicemi (vždy striktně preferuje první nebo druhou koalici). V tomto případě je rozdělení $\Pi_1 = \{\{1, 3\}, \{2\}\}$ core stabilní. Jediný agent, který si v tomto rozdělení může polepšit, je agent 2, který by raději do koalice $\{1, 2\}$ nebo $\{1, 2, 3\}$. Ani jedna však není blokující, protože je v obou koalicích agent 1, který by si pohoršil. Rozdělení Π_1 tedy leží v core této Hedonické hry.

Naopak rozdělení $\Pi_2 = \{\{1, 2, 3\}\}$ už core stabilní není. Oba agenti 1 a 3 striktně preferují koalici $\{1, 3\}$ před $\{1, 2, 3\}$ a koalice $\{1, 3\}$ je tedy blokující. To znamená, že Π_2 neleží v core.

Hedonická hra, kterou jsme si ukázali v tomto příkladu, má tedy neprázdné core.

▶ **Příklad 1.6.** Pojďme si nyní ukázat nějakou hru, která nemá žádné core stabilní rozdělení do koalic:

$$\begin{aligned}
1 &: \{1,2\} \succ_1 \{1,3\} \succ_1 \{1,2,3\} \succ_1 \{1\},\\
2 &: \{2,3\} \succ_2 \{1,2\} \succ_2 \{1,2,3\} \succ_2 \{2\},\\
3 &: \{1,3\} \succ_3 \{2,3\} \succ_3 \{1,2,3\} \succ_3 \{3\}.
\end{aligned}$$

V této hře není žádné možné rozdělení do koalic core stabilní.

- Rozdělení $\{\{1\}, \{2\}, \{3\}\}$, kde je každý agent sám, je blokováno koalicí $\{1, 2, 3\}$.
- Rozdělení $\{1, 2, 3\}$, kde jsou všichni agenti spolu v jedné koalici (tzv. grand coalition), je blokováno například koalicí $\{1, 2\}$.
- Rozdělení $\{\{1, 2\}, \{3\}\}$ je blokováno koalicí $\{2, 3\}$.
- Rozdělení $\{\{2, 3\}, \{1\}\}$ je blokováno koalicí $\{1, 3\}$.
- Rozdělení $\{\{1, 3\}, \{2\}\}$ je blokováno koalicí $\{1, 2\}$.

Rozdělení, která obsahují jako koalici dvojici agentů, tedy obsahují cyklus. Ukázali jsme, že všechna možná rozdělení mají nějakou blokující koalici a core této hry je tedy prázdné.

Jak výpočetně složité je rozhodnout o prázdnosti core dané Hedonické hry, tedy o tom, zda má daná hra core stabilní řešení?

▶ **Věta 1.7** (Ballester, 2004 [2])**.** *Rozhodnout o prázdnosti core pro danou Hedonickou hru* $(N, \succsim)$ *je NP-úplný problém.*

Obecně tedy můžeme říci, že najít nějaké stabilní řešení Hedonické hry je velmi těžký úkol: pouze rozhodnutí o existenci takového řešení je NP-úplný problém.

Nyní se můžeme podívat na některé nejznámější podtřídy Hedonických her.

1.2 Problém stabilních spolubydlících

V této kapitole si nejdříve zavedeme pojem *Hedonická hra s pevnou velikostí koalic* a potom se podíváme na speciální případ, kdy velikost koalice je dva – *Problém stabilních spolubydlících.*

1.2.1 Hedonická hra s pevnou velikostí koalic

Zatímco v obecné Hedonické hře, kterou jsme si zadefinovali v minulé kapitole, smějí koalice nabývat všech možných velikostí (tedy od 1 do $|N|$), zde je nějaká předem pevně daná velikost koalic s. Agenti smějí vytvářet koalice pouze této velikosti, někteří pak mohou zůstat sami. Jejich preference jsou tedy nad koalicemi velikosti s a velikosti 1 (zůstat sami klidně mohou preferovat). Formálně:

- Každou podmnožinu $S \subseteq N$, kde $|S| = s$ nebo $|S| = 1$, nazveme koalicí.

- Rozdělením nebo řešením nazveme rozdělení všech agentů do k koalic $\Pi = \{S_1, S_2, \ldots, S_k\}$, kde $|S_i| = s$ nebo $|S_i| = 1$ pro všechna $i \in \{1, 2, \ldots, k\}$.
- Pref. profil agenta $i \in N$ je pref. relace $\succsim_i$ nad množinou $\{S \subseteq N : i \in S, |S| = s \vee |S| = 1\}$.

Můžeme mít na základě tohoto omezení nějakou garanci existence stabilního řešení? Zjednodušuje nám to výpočetně problém?

▶ **Věta 1.8.** *Hedonická hra s pevnou velikostí koalic nemusí mít žádné core stabilní řešení pro* $s \geq 2$.

▶ **Věta 1.9** (Ng a Hirschberg, 1991 [12]). *Rozhodnout o prázdnosti core pro danou Hedonickou hru s pevnou velikostí koalic je NP-úplný problém, už když* $s = 3$.

▶ **Věta 1.10** (Huang, 2007 [10]). *Rozhodnout o prázdnosti core pro danou Hedonickou hru s pevnou velikostí koalic je NP-úplný problém pro* $s \geq 3$.

Jak vidíme, pevná velikost koalic nám problém příliš nezjednodušuje. Nelze ani garantovat existenci žádného stabilního řešení. Speciální případ je ovšem $s = 2$, kde ačkoli nemůžeme žádné stabilní řešení garantovat, existuje algoritmus, který dokáže v polynomiálním čase odpovědět na otázku existence takového řešení a případně nějaké stabilní řešení vrátit, pokud existuje.

Hedonická hra s pevnou velikostí koalic kde $s = 2$ není nic jiného než jiné označení pro *Problém stabilních spolubydlících*, kterému se budeme věnovat v následující kapitole.

1.2.2 Problém stabilních spolubydlících

Problém stabilních spolubydlících (anglicky *Stable roommates problem*) je velice známá podtřída Hedonických her. Můžeme si jej představit jako Hedonickou hru s pevnou velikostí koalic, ve které $s = 2$. V odborné literatuře je tento problém často zaváděn tak, že se hledá tzv. stabilní párování – to je zcela ekvivalentní tomu najít v této hře core stabilní řešení, tak jak jsme si zavedli v kapitole 1.1. Klasickou motivací pro zavedení tohoto problémů může být následující situace:

Máme kolej, na kterou chceme ubytovat $2n$ osob do n pokojů pro dvě ososby. Každá osoba má preference nad všemi ostatními, se kterými může být na pokoji (můžeme si to představit jako seřazený seznam délky $2n-1$). Lze je rozdělit do pokojů tak, aby potom neexistovaly dvě osoby, které spolu nejsou na jednom pokoji, ale navzájem se preferují více než svého spolubydlícího?

Jak již víme z Věty 1.8, takové rozdělení nemusí vždy existovat. Pojďme si ukázat nějaké příklady, abychom věděli, kde může vzniknout problém.

▶ **Příklad 1.11.** V tomto příkladu máme čtyři agenty: Alici, Annu, Boba a Bruna. V zápisu preferencí šetříme místem – dle definice jsou preference daného agenta nad koalicemi velikosti dva – logicky jedním členem každé koalice je ten daný agent. V zápisu tedy nepíšeme celé koalice, ale pouze druhé agenty.

$$\begin{aligned}
\text{Alice} &: \text{Anna} \succ \text{Bruno} \succ \text{Bob},\\
\text{Anna} &: \text{Alice} \succ \text{Bruno} \succ \text{Bob},\\
\text{Bob} &: \text{Bruno} \succ \text{Alice} \succ \text{Anna},\\
\text{Bruno} &: \text{Bob} \succ \text{Anna} \succ \text{Alice}.
\end{aligned}$$

Rozdělení $\Pi = \{\{\text{Alice, Anna}\}, \{\text{Bob, Bruno}\}\}$ je v tomto případě jediné možné core stabilní rozdělení do koalic.

Jak již ale víme, takové rozdělení nemusí vůbec existovat.

▶ **Příklad 1.12.** Mějme následující příklad tentokrát s agenty Alicí, Brunem, Cecílií a Davidem.

$$\begin{aligned}\text{Alice} &: \text{Bruno} \succ \text{Cecílie} \succ \text{David},\\ \text{Bruno} &: \text{Cecílie} \succ \text{Alice} \succ \text{David},\\ \text{Cecílie} &: \text{Alice} \succ \text{Bruno} \succ \text{David},\\ \text{David} &: \text{Alice} \succ \text{Bruno} \succ \text{Cecílie}.\end{aligned}$$

V tomto případě Alice, Bruno a Cecílie jsou pro někoho vždy tou nejlepší volbou. Naopak Davida, bohužel, nikdo nemá moc rád. V každém možném rozdělení však jeden ze tří oblíbených agentů musí být v koalici s Davidem – ale je tou nejlepší volbou pro jednoho ze dvou ostatních agentů, se kterým by byl radši než s Davidem. Pro úplnost si projdeme všechna rozdělení:

Pokud bychom měli rozdělení {{Alice, Bruno}, {Cecílie, David}}, koalice {Bruno, Cecílie} bude blokující. V rozdělení {{Bruno, Cecílie}, {Alice, David}} je blokující koalice {Alice, Cecílie}. A v rozdělení {{Alice, Cecílie}, {Bruno, David}} je blokující {Alice, Bruno}.

Každé možné rozdělení tedy má nějakou blokující koalici a core této hry je tedy prázdné!

Z Věty 1.10 víme, že pro $s \geq 3$ je NP-úplný problém rozhodnout o prázdnosti core. Platí to i pro $s = 2$?

▶ **Věta 1.13** (Irving, 1985 [11]). *Existuje algoritmus, který v čase $O(n^2)$:*

- *Rozhodne o prázdnosti core pro danou instanci Problému stabilních spolubydlících.*
- *Pokud je core neprázdné, nalezne nějaké core stabilní řešení.*

Tento algoritmus, který v roce 1985 představil Robert W. Irving, je dnes znám jako Irvingův algoritmus (anglicky Irving's algorithm).

1.3 Problém stabilních manželství

Problém stabilních manželství (anglicky *Stable marriage problem*) je dost možná nejznámější podtřída Hedonických her. Představili jej již v roce 1962 D. Gale a L. S. Shapley [9] v kontextu rozdělování uchazečů mezi vysokými školami, berouce v potaz preference uchazečů i vysokých škol.

Nejdříve si tento problém formálně zadefinujeme a ukážeme příklad, nakonec si vysvětlíme souvislost s Hedonickými hrami.

▶ **Definice 1.14.** *Mějme množiny agentů M a W: $|M| = |W| = n$ a $M \cap W = \emptyset$. Každý agent má seřazený seznam všech agentů opačné množiny podle preferovatelnosti.* Problém stabilních manželství *je vytvořit n smíšených párů tak, aby vytvořené párování bylo stabilní.*

▶ **Definice 1.15.** *Vytvořené párování není* stabilní *v případě, že existuje tzv.* blokující pár*, tj. agenti $x \in M$ a $y \in W$:*

- *x striktně preferuje y před svým stávajícím partnerem, nebo x není v páru,*
- *a y striktně preferuje x před svým stávajícím partnerem, nebo y není v páru.*

V opačném případě je párování stabilní.

Můžeme si představit, že agenti z množiny M jsou páni a agenti z množiny W jsou dámy, a úkolem je vytvořit z nich páry. Tímto způsobem bývá tento problém často zadán. Obecně lze ale říci, že nemusí jít nutně o množiny pánů a dam - může jít o libovolné disjuktní množiny.

▶ **Příklad 1.16.** Mějme množinu pánů $M = \{\text{Alan, Boris, Cyril}\}$ a množinu dam $W = \{\text{Diana, Erika, Františka}\}$. Jejich preference jsou následující:

$$\begin{aligned}
\text{Alan} &: \text{Diana} \succ \text{Erika} \succ \text{Františka},\\
\text{Boris} &: \text{Erika} \succ \text{Diana} \succ \text{Františka},\\
\text{Cyril} &: \text{Diana} \succ \text{Erika} \succ \text{Františka},\\
\text{Diana} &: \text{Boris} \succ \text{Alan} \succ \text{Cyril},\\
\text{Erika} &: \text{Alan} \succ \text{Boris} \succ \text{Cyril},\\
\text{Františka} &: \text{Alan} \succ \text{Boris} \succ \text{Cyril}.
\end{aligned}$$

V tomto případě párování {{Alan, Františka}, {Boris, Erika}, {Cyril, Diana}} není stabilní, protože existuje blokující pár {Alan, Erika} – Alan preferuje Eriku před Františkou a zároveň Erika preferuje Alana před Borisem.

Naopak kupříkladu párování {{Boris, Diana}, {Alan, Erika}, {Cyril, Františka}} už stabilní je. V tomto párování totiž neexistuje žádný pár, který by byl blokujícím. Vezměme si například pár {Alan, Diana} – ačkoliv Alan více preferuje Dianu před Erikou, která je jeho druhou volbou, Diana je v páru s Borisem, který je její první volbou a tedy ho preferuje více než Alana. Tento pár tedy není blokující, stejně jako všechny ostatní.

I na tento problém můžeme nahlížet jako na specifickou Hedonickou hru – jde vlastně o Hedonickou hru s pevnou velikostí koalic, kde $s = 2$, přičemž každý agent je jednoho ze dvou typů, počet agentů obou typů je z definice stejný a z hlediska tvorby koalic jsou přípustné pouze smíšené koalice. Potom nalézt v takové hře core stabilní řešení je ekvivalentní nalezení stabilního párování.

Jaké mají tyto omezující podmínky vliv na výpočetní složitost a můžeme na jejich základě vždy garantovat stabilní řešení? Už u Problému stabilních spolubydlících přece máme algoritmus, který v polynomiálním čase rozhodne o existenci takového řešení a případně ho nalezne – zde máme navíc agenty rozděleny na dva typy, počty agentů obou typů jsou stejné a přípustné jsou pouze smíšené koalice!

▶ **Věta 1.17** (Gale a Shapley, 1962 [9])**.** *Každá instance Problému stabilních manželství má stabilní řešení. Existuje algoritmus, který nějaké takové nalezne v čase $O(n^2)$.*

V následující podkapitole se blíže seznámíme s tímto algoritmem.

1.3.1 Gale-Shapleyho algoritmus

Gale-Shapleyho algoritmus poprvé popsali D. Gale a L. S. Shapley v roce 1962 [9]. Tento algoritmus pro danou instanci Problému stabilních manželství vždy nalezne stabilní párování v čase $O(n^2)$, kde $n = |M| = |W|$. Nejprve si popíšeme práci algoritmu a poté dokážeme, že párování, které nalezl, je *perfektní* (všichni jsou v páru) a *stabilní.* Pro názorné vysvětlení jeho práce budeme uvažovat, že M je množina pánů a W je množina dam.

Dále budeme uvažovat dva stavy pánů - zadaný a nezadaný. Celý algoritmus končí ve stavu, kdy jsou již všichni páni zadaní.

Algoritmus 1: Gale-Shapleyho

Vstup : Množina pánů M, množina dam W, $|M| = |W| = n$, jejich preference
Výstup: Stabilní párování o velikosti n

Párování P $= \emptyset$

Pro všechny pány $m \in M$:
    $stav(m) \leftarrow$ nezadaný

Dokud existuje nějaký nezadaný pán $m \in M$:
    m vybere dámu w, kterou preferuje nejvíce ze všech dam, kterým se dosud nenabídl
    m se nabídne w

    Pokud w není v páru:
        $stav(m) \leftarrow$ zadaný
        Přidej pár (m, w) do P

    Pokud w je v páru, ale preferuje m před svým současným partnerem m':
        $stav(m') \leftarrow$ nezadaný
        Odeber pár (m', w) z P
        $stav(m) \leftarrow$ zadaný
        Přidej pár (m, w) do P

Vrať P, obsahující stabilní párování

Nyní si ještě dokážeme, že algoritmus pracuje správně a opravdu nalezne stabilní párování velikosti n.

Důkaz správnosti algoritmu. P obsahuje *párování*, protože vždy vytváříme pár z pána, který není v žádném jiném páru, a dámy, která buď také není v žádném jiném páru, nebo její předchozí pár zrušíme.

Dále, P je *perfektní* párování (tedy všichni jsou v nějakém páru). Nemůže nastat situace, kdy by na konci běhu algoritmu zbyl nějaký pán m a nějaká dáma w, kteří by nebyli v páru – protože každý pán se nabídne každé dámě, pokud je to třeba – tedy v nějakém kroku se určitě m musel nabídnout w, která jej musela přijmout.

A konečně, P je určitě stabilní párování. Mějme pána m a w, kteří spolu netvoří pár: m preferuje w před svou partnerkou a w preferuje m před svým partnerem. V tomto případě by párování nebylo stabilní.

Ale m se přece musel nabídnout w dříve než své partnerce, pokud ji preferuje víc, a w jej musela přijmout, pokud jej preferuje víc než svého partnera (a kohokoliv s kým byla během běhu algoritmu zapárována)! Taková situace tedy nemůže nastat a párování, které vrátí algoritmus, je vždy stabilní. ◀

1.4 Hedonická hra s podmínkami různorodosti

Ve všech modelech, které jsme si až do této doby zavedli, mají jednotliví agenti svoji identitu. Některé modely jsme mohli omezit různými způsoby (např. pevnou velikostí koalic apod.), ale preference jednotlivých agentů jsou stále založeny na této identitě – každý agent je unikátní a agenti při formování koalic rozlišují mezi každým jednotlivým agentem.

Někdy se však v reálném světě můžeme setkat s tím, že potřebujeme zformovat koalice z agentů, kteří jsou nějakých typů, a preference těchto agentů jsou pak založeny na poměru agentů různých typů – v této práci budeme uvažovat typy pouze dva. Může to být například vytváření skupin různých velikostí ze studentů mluvících různými jazyky, rozdělování studentek a studentů do smíšených skupin, sdílené ubytování apod.

Podmínky různorodosti (anglicky *diversity preferences*) tedy znamenají, že agenti z hlediska formování koalic nerozlišují mezi jednotlivými agenty, ale pouze mezi typy těchto agentů. Nyní si formálně zavedeme *Hedonickou hru s podmínkami různorodosti* [6] (*anglicky Hedonic diversity game*), která tento typ problémů modeluje.

▶ **Definice 1.18.** Hedonická hra s podmínkami různorodosti *je trojice* $G = (R, B, (\succ_i)_{i \in R \cup B})$, *kde* R *a* B *jsou disjunktní množiny agentů a preferenční relace* $\succ_i$ *každého agenta* $i \in R \cup B$ *je lineární uspořádání nad množinou*

$$\Theta = \left\{ \frac{r}{s} : r \in \{0, 1, \dots, |R|\}, s \in \{1, \dots, |R| + |B|\} \right\}.$$

Položíme $N = R \cup B$, všechny agenty v R budeme označovat jako *červené* agenty, všechny agenty v B budeme označovat *modrými* agenty. Stejně jako u Hedonických her, každou neprázdnou podmnožinu $S \subseteq N$ nazveme *koalicí.*

Rozdělením do koalic nebo *řešením* (anglicky *coalition partition* nebo *outcome*) nazveme množinu $\Pi = \{S_k\}_{k=1}^{K}$ koalic, která je rozkladem množiny N, tedy $\bigcup_{k=1}^{K} S_k = N$ a $S_k \cap S_\ell = \emptyset$ pro všechna $1 \leq k < \ell \leq K$. Pro dané rozdělení do koalic Π a daného agenta i, $\pi(i)$ označuje koalici $S \in \Pi$ takovou, že $i \in S$.

Pro danou koalici $S \subseteq N$, řekneme že je *poměru* $\theta(S)$ a myslíme tím poměr červených agentů ku všem agentům v této koalici, tedy $\theta(S) = \frac{|S \cap R|}{|S|}$.

U každého agenta $i \in N$ budeme jeho preferenční relaci $\succ_i$ interpretovat jako preference nad poměrem červených agentů v jeho koalici; tedy například

$$\frac{2}{3} \succ_i \frac{3}{5}$$

znamená, že agent i preferuje koalice, ve kterých jsou dvě třetiny červených agentů před koalicemi, ve kterých jsou tři pětiny červených agentů.

Nyní se můžeme podívat na to, jak by mohlo vypadat nějaké korektní zadání Hedonické hry s podmínkami různorodosti:

▶ **Příklad 1.19.** Nechť $G = (\{r_1, r_2, r_3, r_4\}, \{b_1, b_2\}, (\succ_i)_{i \in R \cup B})$, kde:

$$r_1, r_2, r_3, r_4 : \frac{2}{3} \succ \frac{1}{4} \succ \frac{1}{3} \succ \cdots \qquad b_1, b_2 : \frac{2}{3} \succ \frac{4}{5} \succ \frac{5}{6} \succ \cdots$$

V této hře všech šest agentů nejvíce preferuje koalici, ve které by byly dvě třetiny červených agentů.

I zde se zavádí core stabilita, stejně jako u Hedonických her.

▶ **Definice 1.20.** *Řekneme, že koalice* $S \subseteq N$ blokuje *rozdělení* Π*, pokud každý agent* $i \in S$ *preferuje* Π *před* $\pi(i)$*. Rozdělení* Π *je* core stabilní*, pokud toto rozdělení neblokuje žádná koalice* $S \subseteq N$*.*

▶ **Pozorování 1.21.** V příkladu 1.19 máme hned několik core stabilních rozdělení. Je to například rozdělení $\Pi_1 = \{\{r_1, r_2, r_3, r_4, b_1, b_2\}\}$, kde jsou všichni agenti spolu v jedné koalici (tzv. *grand coalition*), nebo $\Pi_2 = \{\{r_1, r_2, b_1\}, \{r_3, r_4, b_2\}\}$. Logicky, jsou to core stabilní rozdělení, protože jsou v nich všichni agenti v koalicích, které nejvíce preferují – nemůže tedy existovat žádná blokující koalice, kterou by preferovali více.

Jak takové preferenční profily agentů mohou v realitě vypadat? Často se setkáváme se situacemi, kdy agenti preferují nějakou pro ně nejvíce optimální volbu, a s narůstající vzdáleností od této nejlepší volby se jejich preference snižuje. Takové preference označujeme jako single-peaked. Opravdu dává smysl zavést si takový typ preferencí, vezměme si například:

- Rozdělení kuřáků a nekuřáků ke stolům. V takovém případě lze očekávat, že agenti obou typů budou chtít být usazeni ke stolu s pokud možno co největším počtem agentů stejného typu, jako jsou oni.
- Nebo rozdělení pekařů a řezníků na různá tržiště. V tomto případě agenti stejného typu soupeří o zákazníky, lze tedy očekávat, že agenti budou preferovat tržiště s pokud možno co nejmenším počtem agentů stejného typu.

▶ **Definice 1.22.** *Řekneme, že preferenční relace agenta $i \in N$ je* single-peaked*, pokud existuje $p_i \in [0, 1]$, a pro všechna $\alpha, \beta \in [0, 1]$ takové, že $p_i \leq \alpha < \beta$ nebo $\beta < \alpha \leq p_i$, platí $\alpha \succ_i \beta$.*

Bohužel, ani takové omezení preferenčních profilů nám tento problém příliš nezjednodušuje.

▶ **Věta 1.23** (Bredereck a kol., 2019 [6])**.** *Hedonická hra s podmínkami různorodosti nemusí mít žádné core stabilní řešení, a to ani když jsou preference všech agentů single-peaked.*

▶ **Věta 1.24** (Bredereck a kol., 2019 [6])**.** *Rozhodnout o tom, zda pro danou Hedonickou hru s podmínkami různorodosti existuje core stabilní řešení, je NP-úplný problém.*

Kapitola 2

Problém stabilních spolubydlících s podmínkami různorodosti

Problém stabilních spolubydlících s podmínkami různorodosti (anglicky *Stable roommates problem with diversity preferences, Roommate diversity problem)* je hlavním předmětem této práce. Tento problém zavedli a pojmenovali v roce 2020 N. Boehmer a E. Elkind [3]. V této kapitole si nejprve ukážeme, kde se nám takový model může hodit, poté si ho řádně zadefinujeme a nakonec řekneme, co o něm již víme.

V kapitole 1.4 jsme si zavedli Hedonickou hru s podmínkami různorodosti a vysvětlili, co tyto podmínky různorodosti znamenají. Obecně můžeme říct, že se tento typ problému liší od Hedonické hry s podmínkami různorodosti pevnou velikostí koalic – zatímco v Hedonické hře mohou koalice nabývat různých velikostí, zde je pevná velikost předem dána.

2.1 Motivace

Na jedné vysoké vyučoval profesor, který na začátku semestru potřeboval rozdělit studenty do skupin po pěti. Na této škole však nebyli pouze domácí, česky mluvící studenti – studovalo zde i velké množství španělsky mluvících studentů na výjezdu.

Rozdělit správně studenty do skupin nebyl vůbec lehký úkol[1]:

- Martin řekl, že by se rád o Španělsku něco dozvěděl, proto by chtěl být ve skupině pouze se španělskými studenty.
- naopak Kryštof řekl, že nemá rád Španělsko a neumí španělsky, a proto chce být ve skupině pouze s česky mluvícími studenty.
- Tadeáš by si rád procvičil španělštinu, ale chtěl by mít ve skupině i nějaké česky mluvící studenty.
- španělsky mluvící Garcia by ráda byla ve skupině s dalším španělsky mluvícím studentem a třemi českými studenty.
- atd.

Profesor je zběhlý v teoretické informatice, takže hned vidí souvislost s Hedonickou hrou s podmínkami různorodosti (kterou jsme si zavedli v kapitole 1.4). Hned je mu jasné, že najít

[1] Jak bude ukázáno později v této kapitole, byl to úkol NP-úplný.

stabilní rozdělení studentů do skupin nebude úplně lehké – rozhodnout o existenci core stabilního řešení v takové hře je NP-úplný problém.

V jeho problému je však ještě něco navíc: velikost koalic je předem jasně daná, podobně jako v Hedonických hrách s pevnou velikostí koalic (kterou jsme zavedli v kapitole 1.2). I v těchto hrách však platí, že rozhodnout o existenci core stabilního řešení je NP-úplné.

Pokud máme v Hedonické hře současně obě dvě tyto omezující podmínky, jak to ovlivní výpočetní složitost a můžeme třeba něco garantovat? Abychom to mohli blíže prozkoumat, byl zaveden [3] *Stable roommates problem with diversity preferences* a já jsem jej do češtiny přeložil jako Problém stabilních spolubydlících s podmínkami různorodosti – dále v práci ho budu označovat zkráceně jako Roommate diversity problem.

2.2 Základní definice

▶ **Definice 2.1.** Roommate diversity problem *s velikostí koalice s je čtveřice $G = (R, B, s, (\succsim_i)_{i \in R \cup B})$, kde $N = R \cup B$ a $|N| = k \cdot s$ pro nějaké $k \in \mathbb{N}$. Preferenční relace $\succsim_i$ každého agenta $i \in N$ je neostré uspořádání nad množinou $D = \{\frac{j}{s} : j \in \{0, 1, \ldots, s\}\}$.*

Všechny agenty v R budeme označovat jako *červené* agenty, všechny agenty v B budeme označovat *modrými* agenty. Dále, $r = |R|$, $b = |B|$ a $n = |R \cup B|$. Každou množinu $S \subseteq N$ o velikosti s označíme jako *koalici*. Dále, nechť $k = \frac{n}{s}$.

Rozdělením do koalic nebo *řešením* (anglicky *outcome*) nazveme rozdělení všech agentů do k koalic $\Pi = \{S_1, S_2, \ldots, S_k\}$, kde $|S_i| = $ s pro všechna $i \in \{1, 2, \ldots, k\}$. Pro dané rozdělení do koalic Π a daného agenta i, $\pi(i)$ označuje koalici $S \in \Pi$ takovou, že $i \in S$.

Pro danou koalici $S \subseteq N$, řekneme že je *poměru* $\theta(S)$ a myslíme tím poměr červených agentů ku všem agentům v této koalici, tedy $\theta(S) = \frac{|S \cap R|}{|S|}$. Pokud $S \subseteq R$, řekneme, že S je *čistě červená* koalice, pokud $S \subseteq B$, řekneme, že S je *čistě modrá* koalice.

U každého agenta $i \in N$ budeme jeho preferenční relaci $\succsim_i$ nad D interpretovat jako preference nad poměrem červených agentů v jeho koalici; tedy například

$$\frac{2}{5} \succsim_i \frac{3}{5}$$

znamená, že agent i slabě preferuje koalici se dvěma červenými agenty před koalicí se třemi červenými agenty.

Mějme dvě koalice $S, T \subseteq N$, jejichž součástí je agent i. Napíšeme $S \succ_i T$ a řekneme že agent i striktně preferuje S před T, jestliže $\theta(S) \succsim_i \theta(T)$ a zároveň $\theta(T) \not\succsim \theta(S)$. Dále, napíšeme $S \succsim_i T$ a řekneme, že i slabě preferuje S před T, pokud $\theta(S) \succsim_i \theta(T)$ nebo $\theta(S) = \theta(T)$. Pokud agent i slabě preferuje S před T a zároveň slabě preferuje T před S, zapíšeme $S \sim_i T$ a řekneme, že agent i je k těmto koalicím lhostejný, případně že je preferuje stejně.

2.2.1 Různé koncepty stability

Nyní již známe všechno potřebné pro to, abychom si mohli zavést nějaké koncepty stability řešení. Budeme se zabývat především core stabilitou, ale pro úplnost jich zde uvedeme více.

▶ **Definice 2.2.** *Rozdělení do koalic Π je* core stabilní*, pokud neexistuje koalice $S \subseteq N$, $|S| = s$ taková, že pro všechny agenty $i \in S$ platí $\theta(S) \succ_i \theta(\pi(i))$. Jako core se potom označuje množina všech těchto core stabilních rozdělení. Pokud taková koalice S existuje, řekneme že S je blokující koalice.*

▶ **Definice 2.3.** *Rozdělení do koalic Π je* silně core stabilní*, pokud neexistuje koalice $S \subseteq N$, $|S| = s$ taková, že pro všechny agenty $i \in S$ platí $\theta(S) \succsim_i \theta(\pi(i))$ a zároveň $\exists i \in S$, pro kterého platí $\theta(S) \succ_i \theta(\pi(i))$.*

▶ **Definice 2.4.** *Rozdělení do koalic* Π *je* Pareto optimální, *jestliže neexistuje rozdělení* Π' *takové, že* $\forall i \in N\colon \theta(\pi'(i)) \succsim_i \theta(\pi(i))$ *a zároveň* $\exists i \in N\colon \theta(\pi'(i)) \succ_i \theta(\pi(i))$.

Jinak řečeno, rozdělení je Pareto optimální, pokud neexistuje nějaké jiné rozdělení, ve kterém by si nikdo nepohoršil, a alespoň jeden agent by si v něm polepšil.

▶ **Definice 2.5.** *Rozdělení do koalic* Π *je* exchange stabilní, *pokud neexistují agenti* $i, j \in N$, $\pi(i) \neq \pi(j)\colon \theta((\pi(j) \setminus \{j\}) \cup \{i\}) \succ_i \theta(\pi(i))$ *a zároveň* $\theta((\pi(i) \setminus \{i\}) \cup \{j\}) \succ_j \theta(\pi(j))$. *Pokud takoví agenti* i, j *existují, řekneme, že si* i *a* j *chtějí vyměnit místo.*

▶ **Definice 2.6.** *Rozdělení do koalic* Π *je* závidění prosté, *pokud neexistují agenti* $i, j \in N$, $\pi(i) \neq \pi(j)\colon \theta((\pi(j) \setminus \{j\}) \cup \{i\}) \succ_i \theta(\pi(i))$. *Pokud takoví agenti* i, j *existují, řekneme, že agent* i *závidí agentovi* j *jeho místo.*

Na závidění prostost tedy můžeme pohlížet jako na jakousi jednostrannou verzi exchange stability.

2.2.2 Strukturované preference

Preferenční relace jednotlivých agentů můžeme navíc různými způsoby omezit – vyžadovat po nich, aby měly určitou strukturu. V této práci se zabýváme především single-peaked profily, ale pro zajímavost si ukážeme i tzv. dichotomické preferenční relace (které, jak se dále dozvíme, nám garantují core stabilní rozdělení).

▶ **Definice 2.7.** *Řekneme, že preferenční relace agenta* i *je* single-peaked, *pokud existuje* $p_i \in D$, *a pro všechna* $\alpha, \beta \in D$ *takové, že* $p_i \leq \alpha < \beta$ *nebo* $\beta < \alpha \leq p_i$, *platí* $\alpha \succsim_i \beta$.

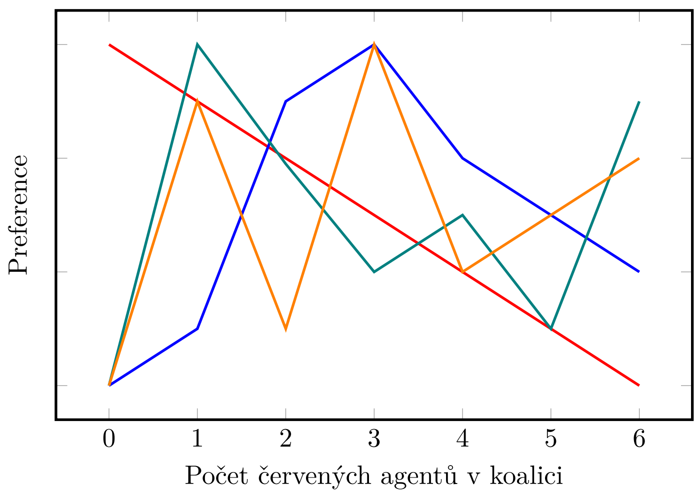
▶ **Příklad 2.8.** Preference, které jsou single-peaked, jsou například:

$$\frac{0}{6} \succ \frac{1}{6} \succ \frac{2}{6} \succ \frac{3}{6} \succ \frac{4}{6} \succ \frac{5}{6} \succ \frac{6}{6} \qquad \frac{3}{6} \succ \frac{2}{6} \succ \frac{4}{6} \succ \frac{5}{6} \succ \frac{6}{6} \succ \frac{1}{6} \succ \frac{0}{6}$$

Naopak single-peaked určitě nejsou tyto preference:

$$\frac{1}{6} \succ \frac{6}{6} \succ \frac{2}{6} \succ \frac{4}{6} \succ \frac{3}{6} \succ \frac{5}{6} \succ \frac{0}{6} \qquad \frac{3}{6} \succ \frac{1}{6} \succ \frac{6}{6} \succ \frac{5}{6} \succ \frac{4}{6} \succ \frac{2}{6} \succ \frac{0}{6}$$

Obrázek 2.1 Zobrazení ukázaných preferencí



▶ **Definice 2.9.** *Instance Roommate diversity problému je* single-peaked*, pokud preferenční relaci všech agentů v této instanci jsou single-peaked.*

▶ **Definice 2.10.** *Řekneme, že preferenční relace agenta i je* dichotomická*, pokud je možné rozdělit D do dvou podmnožin D^+ a D^-, tak aby platilo: $\forall d \in D^+, d' \in D^-$: $d \succ_i d'$, $\forall d, d' \in D^+ : d \sim_i d'$, $\forall d, d' \in D^- : d \sim_i d'$. Řekneme, že agent i schvaluje všechny poměry v D^+ a neschvaluje všechny poměry v D^-.*

▶ **Příklad 2.11.** Následující preference je dichotomická:

$$\frac{1}{6} \sim \frac{3}{6} \sim \frac{5}{6} \sim \frac{6}{6} \succ \frac{0}{6} \sim \frac{2}{6} \sim \frac{4}{6}$$

A jako poslední si uvedeme příklad preference, která je jak single-peaked, tak dichotomická:

$$\frac{3}{6} \sim \frac{4}{6} \sim \frac{5}{6} \succ \frac{0}{6} \sim \frac{1}{6} \sim \frac{2}{6} \sim \frac{6}{6}$$

2.3 Dosavadní poznatky

Vzhledem k tomu, že byl tento typ problému zavedený teprve v minulém roce, zatím nebyl podrobně prozkoumán, jako některé jiné problémy. V této kapitole si shrneme všechny důležité poznatky, které jsou dosud známy. Protože se tomuto problému zatím pořádně věnoval pouze jeden článek [3], všechny věty a důkazy, které v této kapitole uvádíme, jsou převzaty z tohoto článku.

2.3.1 Velikost koalice dva

Pro $s = 2$, Roommate diversity problem je speciálním případem Problému stabilních spolubydlících (kde je znám Irvingův algoritmus, který rozhodne v čase $O(n^2)$ zda existuje stabilní řešení, a nalezne jej, pokud existuje). Jak nám zde pomohou podmínky různorodosti? Zjednoduší nám problém natolik, že můžeme vždy garantovat existenci nějakého stabilního řešení.

▶ **Věta 2.12.** *Roommate diversity problem s velikostí koalic $s = 2$ má vždy řešení, které je Pareto optimální, core stabilní a exchange stabilní. Existuje algoritmus, který takové řešení nalezne v lineárním čase.*

Důležité je si zde uvědomit, že při této velikosti koalice mají agenti vlastně dvě možnosti volby: buď nejvíce preferují čistou barevnou koalici své barvy a na druhém místě preferencí je smíšená koalice poměru $\frac{1}{2}$ nebo naopak. Pokud obě tyto koalice agent preferuje stejně, situace se ještě zjednoduší, protože z hlediska core stability můžeme daného agenta dát do jakékoliv koalice. Logicky, červený agent nemůže být v čistě modré koalici a naopak.

Důkaz Věty 2.12. Algoritmus, který takové řešení najde, je výborně formálně popsán včetně všech důkazů stabilit v plné verzi článku [3]. Základní myšlenka algoritmu je dát dohromady co nejvíce smíšených koalic z agentů, kteří slabě preferují smíšenou koalici před čistou (jednobarevnou) koalicí. Potom se algoritmus pokusí rozmístit zbytek agentů do čistých koalic – pokud je v tomto kroku počet modrých agentů lichý, nebude to možné. V tomto případě, podle preferencí agentů ve smíšených koalicích, buď algoritmus ze zbývajících agentů vytvoří ještě jednu smíšenou koalici a ze zbytku agentů čisté koalice, nebo rozpustí jednu ze smíšených koalicí a ze zbytku včetně těchto dvou agentů vytvoří čisté koalice. ◀

Dále v této kapitole se tedy budeme věnovat tomu, co můžeme obecně říci, pokud $s > 2$.

2.3.2 Core stabilita

Ukázali jsme, že pokud $s = 2$, core stabilní řešení vždy existuje. Jak je tomu, pokud $s > 2$?

▶ **Věta 2.13.** *Roommate diversity problem nemusí mít žádné core stabilní řešení.*

▶ **Příklad 2.14.** Nechť $G = (\{r_1, r_2, r_3, r_4\}, \{b_1, b_2, b_3, b_4\}, 4, (\succsim_i)_{i\in N})$, kde:

$$r_1, r_2, r_3 : \frac{2}{4} \succ \frac{4}{4} \succ \frac{3}{4} \succ \frac{1}{4} \succ \frac{0}{4}, \qquad r_4 : \frac{4}{4} \succ \frac{1}{4} \succ \frac{2}{4} \succ \frac{3}{4} \succ \frac{0}{4},$$
$$b_1, b_2, b_3, b_4 : \frac{1}{4} \succ \frac{2}{4} \succ \frac{3}{4} \succ \frac{0}{4} \succ \frac{4}{4}.$$

Předpokládejme pro spor, že G má core stabilní řešení Π. Pokud Π obsahuje dvě koalice poměru $\frac{2}{4}$, potom je koalice $\{r_4, b_1, b_2, b_3\}$ blokující. Pokud Π obsahuje jednu koalici poměru $\frac{1}{4}$ a jednu koalici poměru $\frac{3}{4}$, potom je blokující koalice $\{r_1, r_2, r_3, r_4\}$. Pokud Π obsahuje čistě červenou a čistě modrou koalici, koalice $\{r_1, r_2, b_1, b_2\}$ je blokující. Tato instance Roommate diversity problému tedy nemá core stabilní řešení.

▶ **Věta 2.15.** *Roommate diversity problem nemusí mít žádné core stabilní řešení ani v případě, že jsou preference všech agentů single-peaked.*

Instance, ve které jsou preference všech agentů single-peaked, a má prázdné core, je ukázána v Příkladu 4.1.

▶ **Věta 2.16.** *Rozhodnout o tom, zda pro danou instanci Roommate diversity problému existuje core stabilní řešení, je NP-úplný problém.*

▶ **Věta 2.17.** *Roommate diversity problem, kde jsou prefence všech agentů dichotomické, má vždy core stabilní řešení. Takové řešení lze naleznout v polynomiálním čase.*

Důkaz. Algoritmus, který nalezne takové řešení, je následující: iterujeme přes všechny poměry $\frac{\ell}{s}$ pro $\ell \in \{0, 1, \ldots, s\}$ a vždy vytvoříme co největší počet koalic z ℓ červených agentů a $s - \ell$ modrých agentů, kteří všichni schvalují poměr $\frac{\ell}{s}$. Zbytek agentů rozdělíme jakkoliv do zbývajících koalic. ◀

2.3.3 Exchange stabilita

A jak je tomu s exchange stabilitou? Můžeme si nejdříve zavést trochu slabší verzi, tzv. *same-type-exchange stabilitu*, kterou můžeme aplikovat na nějaké příklady ze života. Například, profesor z kapitoly Motivace 2.1 může chtít, aby v každé skupině byl stejný poměr česky a španělsky mluvících studentů. Nepřipadá tedy v úvahu povolit neomezené prohazování studentů mezi skupinami, ale může povolit prohození skupin dvěma českým studentům nebo dvěma španělským studentům.

Same-type-exchange stabilita tedy znamená, že povolíme vyměňování místa v koalici pouze agentům stejného typu – poměr červených agentů v těchto koalicích tak vždy zůstane zachován.

▶ **Věta 2.18.** *Roommate diversity problem má vždy same-type-exchange stabilní řešení, a nějaké takové řešení lze nalézt v polynomiálním čase.*

Důkaz. Abychom získali nějaké same-type-exchange stabilní řešení, začneme s nějakým počátečním rozdělením Π. Dokud v Π existují agenti i a j stejného typu, kteří si z definice exchange stability chtějí vyměnit místo, vyměníme je. Tento algoritmus vždy konverguje ke stabilnímu řešení – protože prohazujeme agenty stejného typu, poměry červených agentů v koalicích se tedy nemění. Oba agenti, které prohodíme, si polepší – pro ostatní agenty se nic nemění! Algoritmus se tedy vždy zastaví v bodě, kdy už si žádní dva agenti stejného typu nechtějí vyměnit místo. ◀

Bohužel, toto se nedá říci obecně o exchange stabilitě (kde můžeme prohazovat agenty obou typů, čímž mohou být změněny poměry jednotlivých koalic).

▶ **Věta 2.19.** *Roommate diversity problem nemusí mít žádné exchange stabilní řešení, a to ani v případě, že jsou preference všech agentů single-peaked.*

▶ **Příklad 2.20.** Nechť $G = (\{r_1, r_2, r_3, r_4, r_5\}, \{b_1, b_2, b_3, b_4\}, 3, (\succsim_i)_{i \in N})$, kde:

$$r_1, r_2 : \frac{3}{3} \succ \frac{2}{3} \succ \frac{1}{3} \succ \frac{0}{3}, \qquad r_3, r_4, r_5 : \frac{2}{3} \succ \frac{1}{3} \succ \frac{3}{3} \succ \frac{0}{3},$$
$$b_1 : \frac{0}{3} \succ \frac{1}{3} \succ \frac{2}{3} \succ \frac{3}{3}, \qquad b_2, b_3, b_4 : \frac{1}{3} \succ \frac{2}{3} \succ \frac{0}{3} \succ \frac{3}{3}.$$

V této hře jsou preference všech agentů single-peaked a hra nemá žádné exchange stabilní řešení.

▶ **Věta 2.21.** *Rozhodnout o tom, zda pro danou instanci Roommate diversity problému exsituje exchange stabilní řešení, je NP-úplný problém.*

2.3.4 Závidění prostost

Stejně jako tomu bylo s exchange stabilitou, i zde si můžeme zavést tzv. same-type závidění prostost – to znamená, že agent může jinému agentovi závidět místo, pouze pokud jsou stejného typu. Bohužel existenci řešení, které by bylo stabilní z tohoto hlediska, nelze garantovat.

▶ **Věta 2.22.** *Existuje instance Roommate diversity problému s velikostí koalice dva, která nemá žádné same-type závidění prosté řešení, a tedy ani žádné závidění prosté řešení. Navíc, v této instanci jsou preference všech agentů single-peaked a dichotomické.*

▶ **Příklad 2.23.** Nechť $G = (\{r_1\}, \{b_1, b_2, b_3\}, 2, (\succsim_i)_{i \in N})$, kde:

$$r_1 : \frac{2}{2} \succ \frac{1}{2} \succ \frac{0}{2}, \qquad b_1, b_2, b_3 : \frac{0}{2} \succ \frac{1}{2} \succ \frac{2}{2}.$$

Zde jsou preference všech agentů single-peaked – pokud bychom spojili druhou a třetí preferenci každého agenta, byly by i dichotomické. Každé rozdělení do koalic obsahuje jednu čistě modrou a jednu smíšenou koalici – modrý agent, který je ve smíšené koalici, bude vždy závidět místo ostatním dvěma modrým agentům.

▶ **Věta 2.24.** *Každá instance Roommate diversity problému má same-type závidění prosté řešení, pokud je r (počet červených agentů) dělitelný s (velikostí koalic) nebo k (počtem koalic).*

Důkaz. Pokud s dělí r, existuje rozdělení do koalic takové, kde jsou pouze čistě červené a čistě modré koalice. Pokud k dělí r, existuje takové rozdělení, kde jsou všechny koalice poměru $\frac{r/k}{s}$. V obou dvou případech nezávidí žádný agent žádnému jinému agentovi stejného typu místo, protože jsou v koalicích stejného poměru. ◀

▶ **Věta 2.25.** *Rozhodnout o tom, zda pro danou instanci Roommate diversity problému existuje závidění prosté řešení, je NP-úplný problém.*

2.3.5 Pareto optimalita

Ačkoliv pro předchozí koncepty stability není vždy garantováno, že takové stabilní řešení existuje, Pareto optimální řešení vždy z definice Pareto optimality existuje. Takové řešení však není vůbec lehké najít.

▶ **Definice 2.26.** *Jazyk* $L \subseteq \Sigma^*$ *je v coNP právě tehdy, když jazyk* $\overline{L} = \Sigma^* \setminus L$ *je v NP.*

▶ **Věta 2.27.** *Rozhodnout o tom, zda je dané řešení Pareto optimální v dané instanci roommate diversity problému, je coNP-úplný problém. Navíc, vypočítat Pareto optimální řešení v polynomiálním čase by bylo možné, pouze pokud* $P = NP$.

2.3.6 Parametrizovaná složitost

▶ **Definice 2.28.** Parametrizovaný problém *je formálně definován jako jazyk* $L \subseteq \Sigma^* \times \mathbb{N}$, *kde* Σ *je konečná neprázdná abeceda. Druhý prvek je zván* parametrem *problému* L.

▶ **Definice 2.29.** *Parametrizovaný problém* L *je* parametrizovaně dostupný (fixed-parameter tractable)*, pokud existuje algoritmus, který pro danou dvojici* (x, k)

- *rozhodne, zda* $(x, k) \in L$, *a*
- *pracuje v čase* $f(k) \cdot |x|^c$, *kde* f *je nějaká algoritmicky vyčíslitelná funkce a* c *je konstanta.*

Třídu všech parametrizovaně dostupných problémů označujeme FPT.

V kapitole 2.3.1 jsme si ukázali, že pokud máme $s = 2$, tak nám to výrazně snižuje výpočetní složitost tohoto problému. Vždy existuje (core/exchange) stabilní řešení, a dokonce takové řešení lze nalézt v polynomiálním čase poměrně jednoduchým algoritmem.

▶ **Věta 2.30.** *Rozhodnout o tom, zda pro danou instanci Roommate diversity problému existuje core stabilní / exchange stabilní / závidění prosté řešení, je FPT vzhledem k velikosti koalice* s.

A právě to mě motivuje k další práci – prozkoumat, jestli nějaký takový jednoduchý algoritmus neexistuje pro $s > 2$, kupříkladu pro $s = 3$, a zda můžeme garantovat něco o existenci stabilních řešení, třeba i za předpokladu strukturovaných preferencí agentů.

Kapitola 3

Nový algoritmus

V této kapitole představujeme nový randomizovaný algoritmus, který pro danou instanci Roommate diversity problémů hledá core stabilní řešení, a pokud nějaké takové existuje, s velkou pravděpodobností ho velmi rychle nalezne.

V podkapitole 3.1 zavádíme nějaké nové definice, které nám umožní a ulehčí jeho popis, ukazujeme, jak jsme k algoritmu dospěli, a také proč by algoritmus měl fungovat. V následující podkapitole 3.2 se věnujeme popisu algoritmu, jeho implementaci a jeho analýze.

Mým hlavním cílem bylo najít algoritmus, který by pracoval rychle a efektivně, na základě jehož výsledků bychom mohli říci obecně něco o Roommate diversity problému, a pomocí kterého bychom mohli najít nějaké zajímavé malé instance, které nemají core stabilní řešení. Najít algoritmus, který by pracoval deterministicky, se mi nepodařilo, přestože jsem zkoušel více různých přístupů. Já jsem však s výsledným algoritmem spokojen, protože posloužil k obojímu, přestože je randomizovaný, a pracuje rozumně rychle a zvládá i větší instance.

3.1 Cesta k algoritmu

Cílem této podkapitoly je objasnit, jak jsem k algoritmu dospěl a z jakého důvodu funguje, případně ukázat, proč nefungují některé jiné nápady. Nejdříve je zde vysloveno tvrzení, které nám často umožní zmenšit velikost problému, potom jsou zde zavedeny nějaké nové definice, umožňující popis algoritmu, a dále jsou zde různá pozorování.

▶ **Lemma 3.1.** *Mějme nějakou instanci Roommate diversity problému $G = (R, B, s, (\succsim_i)_{i \in R \cup B})$. Pokud existuje ℓ červených agentů a $s - \ell$ modrých agentů takových, že jejich nejvíce preferované koalice jsou poměru $\frac{\ell}{s}$, tyto agenty můžeme umístit spolu do jedné koalice – nebudou nikdy v žádné blokující koalici a zbývá nám vyřešit menší instanci problému bez těchto agentů.*

Toto je velmi důležité tvrzení. Například, když rovnoměrně náhodně generujeme zadání (preference agentů) Roommate diversity problému se 180 agenty a velikostí koalice 3, lze tímto způsobem hned na začátku umístit do koalic obrovské množství agentů – většinou nám zbyde 21-42 agentů, a stačí nám pouze vyřešit tuto menší instanci.

Tedy na začátku libovolného algoritmu, který má hledat core stabilní řešení, můžeme vytvořit koalice z agentů, kteří nejvíce preferují právě tento typ koalice – a pokud najdeme core stabilní řešení zbylých agentů, máme core stabilní řešení celé instance.

A co se zbytkem? Nejdříve si zavedeme nějaké definice.

▶ **Definice 3.2.** *Mějme daného agenta i Roommate diversity problému a jeho preferenční relaci $\succsim_i$. Jako* třídu indiference *označíme množinu prvků z $\succsim_i$, mezi kterými je agent i lhostejný/indiferentní.*

▶ **Příklad 3.3.** Zde máme preferenční relaci nějakého agenta j. Třídy indiference v této relaci jsou znázorněny stejnou barvou.

$$j : \frac{0}{6} \sim \frac{1}{6} \succ \frac{2}{6} \sim \frac{3}{6} \sim \frac{4}{6} \succ \frac{5}{6} \succ \frac{6}{6}$$

▶ **Definice 3.4.** *Mějme danou instanci Roommate diversity problému $G = (R, B, s, (\succsim_i)_{i \in R \cup B})$ a dané rozdělení agentů ($N = R \cup B$) do koalic Π. Jako* nespokojenost *agenta $i \in N$ v rozdělení Π označíme počet tříd indiference, které jsou pro agenta i lepší než třída, ve které je $\theta(\pi(i))$.*

▶ **Příklad 3.5.** Zde máme preferenční relaci agenta x, červenou barvou je zvýrazněno $\theta(\pi(i))$ v daném rozdělení Π (podíl červených agentů v agentově koalici v rozdělení Π).

$$x : \frac{2}{4} \succ \frac{4}{4} \succ \frac{3}{4} \succ \frac{1}{4} \succ \frac{0}{4}$$

Nespokojenost agenta x v rozdělení Π je zde 2. Jak vidno, je v koalici, ve které jsou tři červení agenti, ale radši by byl v koalici, ve které by byli čtyři červení agenti (jeho nespokojenost by zde byla 1), a nejraději v koalici, ve které by byli dva červení agenti (zde by byla nulová).

▶ **Definice 3.6.** *Mějme danou instanci Roommate diversity problému $G = (R, B, s, (\succsim_i)_{i \in R \cup B})$. Jako* celkovou míru nespokojenosti *rozdělení Π označíme součet nespokojeností všech agentů $i \in N$ v Π.*

$$\sum_{i \in N} \text{nespokojenost } i \text{ v } \Pi$$

▶ **Definice 3.7.** *Mějme danou instanci Roommate diversity problému $G = (R, B, s, (\succsim_i)_{i \in R \cup B})$. Rozdělení do koalic Π nazveme* spokojené, *pokud neexistuje nějaké jiné rozdělení Π', které by mělo menší celkovou míru nespokojenosti.*

▶ **Pozorování 3.8.** Roommate diversity problem může mít více spokojených rozdělení.

Nechť $G = (\{r_1, r_2, r_3, r_4\}, \{\}, 2, (\succsim_i)_{i \in N})$, kde:

$$r_1, r_2, r_3, r_4 : \frac{2}{2} \succ \frac{1}{2} \succ \frac{0}{2}$$

Zde je celková míra nespokojenosti každého možného rozdělení nula, a tedy každé možné rozdělení je spokojené.

▶ **Pozorování 3.9.** Spokojené rozdělení nemusí být vždy core stabilní, a to ani když daná instance Roommate diversity problému nějaké core stabilní řešení má.

Nechť $G = (\{r_1, r_2, r_3\}, \{b_1, b_2, b_3\}, 3, (\succsim_i)_{i \in N})$, kde:

$$\begin{aligned} r_1 &: \frac{2}{3} \succ \frac{1}{3} \succ \frac{3}{3} \succ \frac{0}{3}, & b_1, b_2 &: \frac{0}{3} \succ \frac{2}{3} \succ \frac{1}{3} \succ \frac{3}{3}, \\ r_2 &: \frac{2}{3} \succ \frac{3}{3} \succ \frac{1}{3} \succ \frac{0}{3}, & b_3 &: \frac{2}{3} \succ \frac{0}{3} \succ \frac{1}{3} \succ \frac{3}{3}, \\ r_3 &: \frac{3}{3} \succ \frac{2}{3} \succ \frac{1}{3} \succ \frac{0}{3}. \end{aligned}$$

V této hře existuje jediné spokojené rozdělení $\Pi = \{\{r_1, r_2, r_3\}, \{b_1, b_2, b_3\}\}$, ve kterém je celková míra nespokojenosti rovná čtyřem. Ačkoliv toto rozdělení maximalizuje spokojenost agentů (minimalizuje celkovou míru nespokojenosti), není core stabilní, protože má blokující koalici $\{r_1, r_2, b_3\}$.

Naopak rozdělení $\{\{r_1, r_2, b_3\}, \{r_3, b_1, b_2\}\}$ je core stabilní – zde je však celková míra nespokojenosti 6.

Takže nestačí pouze najít spokojené rozdělení, protože to nemusí být vždy core stabilní. Navíc, najít nějaké takové rozdělení může být vlastně výpočetně velmi těžké – jedná se o svým způsobem nejoptimálnější rozdělení ze všech.

Zde bychom si měli ještě ukázat, co určitě nefunguje. Někoho by mohlo napadnout, že když existuje tak jednoduchý hladový algoritmus pro $s = 2$, mohl by existovat podobně jednoduchý kupříkladu pro $s = 3$, kde jsou preference všech agentů single-peaked.

▶ **Pozorování 3.10.** Algoritmus, který postupně rozděluje agenty do koalic tak, že v každém kroku vytvoří koalici s nejmenší nespokojeností, nemusí vrátit core stabilní řešení ani pro $s = 3$, kde preference všech agentů jsou single-peaked.

Nechť $G = (\{r_1, r_2, r_3, r_4\}, \{b_1, b_2\}, 3, (\succsim_i)_{i \in N})$, kde:

$$r_1, r_2, r_3 : \frac{1}{3} \succ \frac{2}{3} \succ \frac{3}{3} \succ \frac{0}{3}, \qquad r_4 : \frac{3}{3} \succ \frac{2}{3} \succ \frac{1}{3} \succ \frac{0}{3},$$
$$b_1 : \frac{2}{3} \succ \frac{1}{3} \succ \frac{0}{3} \succ \frac{3}{3}, \qquad b_2 : \frac{1}{3} \succ \frac{2}{3} \succ \frac{0}{3} \succ \frac{3}{3}.$$

Tento algoritmus v prvním kroku vytvoří koalici $\{r_1, b_1, b_2\}$, kde je celková nespokojenost 1, a ze zbytku agentů koalici $\{r_2, r_3, r_4\}$. Můžeme si rozmyslet, že toto rozdělení blokuje koalice $\{r_2, r_3, b_1\}$. Tento příklad výborně ilustruje to, na co jsem vždy narazil při pokusech o deterministický algoritmus – máme tak jednoduché zadání, tak jasnou volbu první vytvářené koalice (pro dva agenty nejlepší volba, pro třetího agenta druhá nejlepší volba), a přece to nevede ke core stabilitě.

Nyní již přímo k algoritmu. Jak již bylo řečeno, spokojené rozdělení nemusí být vždy core stabilní. Následující pozorování je ale klíčové:

▶ **Pozorování 3.11.** Je velmi vysoká pravděpodobnost, že spokojené rozdělení je core stabilní. Tabulka 3.1 ukazuje, jak je vysoká pravděpodobnost, že náhodně vybrané spokojené rozdělení z náhodně vygenerované instance Roommate diversity problému určitých parametrů je core stabilní.

Jedná se o odhad pravděpodobnosti. Byl náhodně vygenerován dostatečně velký vzorek instancí daných parametrů (řádově desítky až stovky tisíc), u každé instance bylo nalezeno nějaké spokojené rozdělení a potom bylo testováno, zda je nalezené spokojené rozdělení core stabilní.

■ **Tabulka 3.1** Pravděpodobnost, že spokojené rozdělení je core stabilní

# agentů	velikost koalic	# červených agentů	single-peaked	P
6	3	3	NE	93,8%
6	3	3	ANO	97,3%
8	4	4	NE	88%
8	4	4	ANO	93%
10	5	5	NE	96,9%
10	5	5	ANO	97,9%
12	6	6	NE	93,9%
12	6	6	ANO	96,7%

Kupříkladu šestý řádek tabulky lze chápat tímto způsobem - generujeme náhodně instance Roommate diversity problému, kde $|R \cup B| = 10$, $|R| = 5$, $s = 5$ a preference všech agentů jsou single-peaked. Potom pravděpodobnost, že náhodně vybrané spokojené rozdělení (minimalizující nespokojenosti agentů) bude core stabilní, je 97,9%. Ještě tomu přidává na váze fakt, že u této velikosti koalice již nemáme garanci existence core stabilního řešení.

Můžeme se tedy snažit nějakým způsobem minimalizovat celkovou míru nespokojenosti a sledovat, zda to vede ke core stabilnímu řešení. Nalézt spokojené rozdělení je ovšem velmi těžký úkol (nepřišel jsem na lepší způsob než vyzkoušet všechna možná rozdělení). Nabízí se ale začít s nějakým rozdělením a prohazovat agenty mezi koalicemi tak, aby v každém prohození klesla celková míra nespokojenosti – to určitě konverguje k nějakému rozdělení, protože se nespokojenost nemůže zmenšovat nekonečně dlouho. Ačkoliv tento postup nevede vždy ke spokojenému rozdělení, vrací rozdělení, která jsou z hlediska nespokojenosti agentů velmi rozumná a často také core stabilní.

Na tomto následujícím (posledním) pozorování se zakládá můj algoritmus. Jeho součástí je opět i tabulka výsledků mých měření.

▶ **Pozorování 3.12.** Mějme instanci Roommate diversity problému a následující algoritmus:

Algoritmus 2:

```
Na začátku vytvoř co nejvíce koalic dle Lemma 3.1
Zbytek agentů rozděl náhodně do koalic
Dokud existuje dvojice agentů, jejichž prohození snižuje celkovou míru nespokojenosti,
  prohoď je
Pokud je na konci algoritmu nebo v kterémkoliv jeho kroku rozdělení core stabilní,
  zahlaš jej
```

Je velmi vysoká pravděpodobnost, že takový algoritmus nalezne core stabilní řešení. Pozorování je opět podepřeno tabulkou naměřených výsledků. V Tabulce 3.2 je vidět, jak si tento algoritmus vede pro různé instance Roommate diversity problému.

Pochopitelně se opět jedná o odhad této pravděpodobnosti. Bylo náhodně vygenerováno řádově desítky až stovky tisíc instancí daných parametrů, a na nich spuštěn algoritmus popsaný pseudokódem. Přitom byl ještě měřen průměrný počet prohození agentů.

Tabulka 3.2 Pravděpodobnost nalezení core stabilního řešení

# agentů	velikost koalic	# čer. agentů	single-peaked	prům. # prohození	P
6	3	3	NE	0,2	97,3%
6	3	3	ANO	0,08	97,6%
60	3	30	NE	4,5	96,6%
60	3	30	ANO	1,75	91,1%
180	3	90	NE	12,5	98,5%
180	3	10	NE	3,7	75%
16	4	8	ANO	1,9	87%
16	4	8	NE	2,5	81,4%
40	4	20	ANO	4	87,3%
100	4	50	ANO	7	82%
100	5	50	ANO	14,5	72,5%

Například pátý řádek tabulky znamená, že náhodně generujeme instance Roommate diversity problému, kde $|R \cup B| = 180$, $|R| = 90$, $s = 3$ a preference nemusí být single-peaked. Výše popsaný algoritmus nalezne core stabilní řešení s pravděpodobností 98,5% a průměrně provede 12.5 prohození agentů. Úspěšnost dále s rostoucí velikostí koalic klesá.

Situace, kde výše popsaný algoritmus nenalezne core stabilní řešení, si vysvětluji tak, že se při prohazování agentů buď zasekl někde daleko od spokojeného rozdělení, nebo daná instance core stabilní řešení zkrátka nemá.

První variantu by mohlo vyřešit znovu rozdělit zbytek agentů náhodně do koalic a začít prohazování znovu. To je základní myšlenkou mého algoritmu, který si popíšeme v následující podkapitole.

3.2 Algoritmus

V minulé podkapitole jsme došli k nápadu na algoritmus, který by mohl fungovat. Následuje jeho popis pseudokódem:

Algoritmus 3: Rychlé hledání core stabilního řešení

Vstup : Roommate diversity problem
Výstup: Core stabilní řešení, existuje-li

1 Vytvoř co nejvíce koalic dle Lemma 3.1
2 Nepřiřazení agenti ← agenti, kteří nebyli umístěni do koalice v kroku 1
3 Rozděl všechny nepřiřazené agenty náhodně do koalic
4 Pro každou dvojici nepřiřazených agentů x,y:
5   Pokud se prohozením x a y sníží celková míra nespokojenosti:
6   Prohoď x a y
7   Začni znovu krok 4
8 Pokud rozdělení není core stabilní, *vrať se* na krok 3
9 Vrať rozdělení, které je nyní core stabilní

▶ **Poznámka 3.13.** Samozřejmě lze přidat parametr maximálního počtu vrácení (v kroku 8), aby se algoritmus necyklil nekonečně. Nastavení hodnoty parametru je potom značně závislé na parametrech řešených instancí.

Tento algoritmus se velmi osvědčil a funguje až překvapivě rychle i pro velké instance. Nejrychleji pracuje, když má na vstupu instance s podobným množstvím červených a modrých agentů.

Tabulka 3.3 ukazuje, kolikrát průměrně se algoritmus musí vrátit (v kroku 8) pro nalezení core stabilního řešení jedné instance Roommate diversity problému různých parametrů. Tyto hodnoty byly získány tím způsobem, že byly náhodně generovány instance daných parametrů a posílány na vstup algoritmu, přičemž byl počítán počet nalezených core stabilních řešení a celkový počet vrácení.

Tabulka 3.3 Průměrný počet vrácení v Algoritmu 3

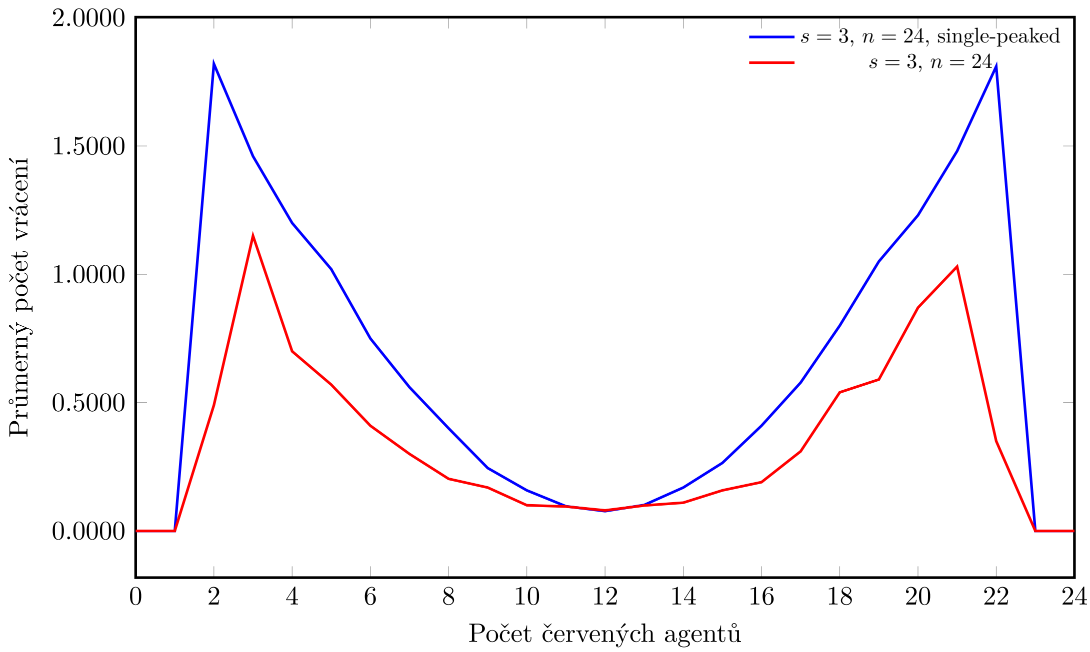
# agentů	velikost koalic	# červených agentů	single-peaked	prům. # vrácení
180	3	90	ANO	0,04
180	3	90	NE	0,7
180	3	60	ANO	3,5
180	3	30	ANO	9
30	3	15	ANO	0,09
30	3	15	NE	0,095
30	3	10	ANO	0,5
30	3	10	NE	0,3
30	3	5	ANO	1,77
30	3	5	NE	1,15
80	4	40	ANO	0,35

Jak vyplývá z tabulky, průměrný počet vrácení je často velmi nízký, a to především pokud je ve hře podobný počet červených a modrých agentů.

Obrázek 3.1 tuto skutečnost ilustruje: ukazuje, kolikrát průměrně se algoritmus musí vrátit pro nalezení core stabilního řešení jedné instance Roommate diversity problému v závislosti na počtu červených agentů.

Bereme v potaz instance, kde $s = 3$, $n = 24$ a všechny preference jsou single-peaked (modrá barva) a instance, kde $s = 3$, $n = 24$ a preference nemusí být single-peaked (červená barva). Je měněn počet červených agentů v generovaných instancích a naměřené hodnoty jsou opět průměrem.

Obrázek 3.1 Průmerný počet vrácení v Algoritmu 3 v závislosti na počtu červených agentů



Může nás to trochu překvapit, ale v tomto konkrétním případě ($s = 3$ a $n = 24$) je větší pravděpodobnost, že algoritmus nalezne core stabilní řešení na první pokus u instancí, které nemusí být nutně single-peaked.

Z obrázku je jasně vidět, že tato pravděpodobnost je u obou dvou případů nejvyšší (tedy průměrný počet vrácení je nejnižší), pokud se počet červených agentů rovná počtu modrých agentů. To je proto, že když je v instanci podobný počet červených a modrých agentů, na začátku algoritmu (v kroku 1) se pevně vytvoří průměrně větší počet koalic a dořešit danou instanci je poté značně jednodušší.

▶ **Pozorování 3.14.** Představený algoritmus na hledání core stabilních řešení pracuje nejrychleji především pro náhodně generované instance s podobným počtem agentů obou typů.

V následující kapitole si ukážeme, na co jsme mohli i díky algoritmu přijít.

Kapitola 4

Nové poznatky

V této kapitole shrneme nejdůležitější nové poznatky o Roommate diversity problému, které jsme mohli díky algoritmu představenému v minulé kapitole učinit.

4.1 Single-peaked preference

V první řadě si můžeme povšimnout, že pokud jsou v dané instanci Roommate diversity problému preference všech agentů single-peaked, algoritmus běží rychleji, protože provede znatelně menší počet prohození agentů. Je tedy velice pravděpodobné, že single-peaked preference problém výrazně zjednodušují.

U klasického Problému stabilních spolubydlících (bez podmínek různorodosti, velikost koalice dva) jsou single-peaked preference jednou z podmínek, která garantuje existenci core stabilního řešení [5]. Zaměřil jsem se tedy na nalezení pokud možno co nejmenší a nejjednodušší instance Roommate diversity problému, ve které jsou všechny preference single-peaked, a která nemá core stabilní řešení.

Zde bychom si měli připomenout, že jsme si již v Příkladu 2.14 ukázali instanci Roommate diversity problému s osmi agenty a velikostí koalice $s = 4$, která měla prázdné core. V této instanci však nebyly preference všech agentů single-peaked!

Mně se podařilo nalézt následující instanci:

▶ **Příklad 4.1.** Nechť $G = (\{r_1, r_2, r_3, r_4, r_5, r_6\}, \{b_1, b_2, b_3, b_4\}, 5, (\succsim_i)_{i \in N})$, kde:

$$\begin{aligned} r_1, r_2, r_3 &: \tfrac{5}{5} \succ \tfrac{4}{5} \succ \tfrac{3}{5} \succ \tfrac{2}{5} \succ \tfrac{1}{5} \succ \tfrac{0}{5}, & b_1, b_2, b_3 &: \tfrac{3}{5} \succ \tfrac{4}{5} \succ \tfrac{2}{5} \succ \tfrac{1}{5} \succ \tfrac{0}{5} \succ \tfrac{5}{5}, \\ r_4, r_5 &: \tfrac{3}{5} \succ \tfrac{4}{5} \succ \tfrac{5}{5} \succ \tfrac{2}{5} \succ \tfrac{1}{5} \succ \tfrac{0}{5}, & b_4 &: \tfrac{4}{5} \succ \tfrac{3}{5} \succ \tfrac{2}{5} \succ \tfrac{1}{5} \succ \tfrac{0}{5} \succ \tfrac{5}{5}, \\ r_6 &: \tfrac{4}{5} \succ \tfrac{3}{5} \succ \tfrac{5}{5} \succ \tfrac{2}{5} \succ \tfrac{1}{5} \succ \tfrac{0}{5}. \end{aligned}$$

V této hře jsou preference všech agentů single-peaked. Algoritmicky bylo ověřeno, že tato instance nemá žádné core stabilní řešení!

▶ **Poznámka 4.2.** Tato instance je, co do počtu agentů, nejmenší možnou single-peaked instancí Roommate diversity problému, která má prázdné core. Určitě neexistuje single-peaked instance s méně než deseti agenty, která by měla prázdné core. Mohla by to být například nějaká single-peaked instance, kde $s = 4$ a $n = 8$ – u těchto jsem ale ověřil, že všechny mají nějaké core stabilní řešení (4.3).

Díky tomu tedy můžeme s naprostou jistotou říci, že single-peaked preference pro velikost koalice $s > 4$ určitě negarantují existenci core stabilního řešení.

U $s = 2$ víme, že vždy existuje core stabilní řešení. Jak je tomu u velikostí koalice $s = 3$ a $s = 4$?

4.2 Pozorování

Protože algoritmus představený v minulé kapitole není deterministický a pracuje na náhodnosti, nelze jej použít jako důkaz pro následující tvrzení – proto je vysloveno pouze jako domněnka.

▶ **Domněnka 4.3.** Domníváme se, že Roommate diversity problem, kde

- s = 3 a preference všech agentů jsou single-peaked,
- s = 3 a preference mohou být libovolné,
- s = 4 a preference všech agentů jsou single-peaked,

má vždy core stabilní rozdělení do koalic.

Ačkoliv to nemůžeme říci zcela jistě, tuto domněnku máme podepřenu následujícím:

- U Roommate diversity problému s parametry, kde víme, že nemusí existovat core stabilní řešení (tedy $s = 4$, kde preference nejsou single-peaked, $s = 5$, kde preference jsou single-peaked, a potom samozřejmě dále), máme vždy triviální instanci velikosti dvou koalic, která má prázdné core.
- Dále, pokud náhodně generujeme instance Roommate diversity problému s parametry, kdy víme, že může být prázdné core, a pomocí představeného algoritmu hledáme core stabilní řešení, algoritmus velmi brzy najde instanci, která takové řešení nemá (řádově stovky až desítky tisíc vygenerovaných instancí).
- Já jsem však zkoušel generovat stovky milionů instancí daných parametrů ($s = 3$, $s = 4$ single-peaked) různými způsoby (zcela náhodně, různě systematicky apod.), a dosud jsem nenašel žádnou, která by neměla core stabilní řešení (algoritmus takové vždy našel). Generoval jsem malé instance různých poměrů červených agentů, ale i obrovské instance (až několik set agentů).
- Prošel jsem všechny možné malé instance u $s = 3$ (příští podkapitola 4.3).

Pokud by tedy měly existovat nějaké instance těchto parametrů, které by měly mít prázdné core, tak jsou pravděpodobně obrovské, a zcela jistě jsou zastoupeny velmi řídce (dosud žádná nebyla nalezena).

4.3 Hrubá síla

Díky tomu, že máme k dispozici rychlý algoritmus na hledání core stabilních řešení, můžeme vyslovit následující větu:

▶ **Věta 4.4.** *Každá instance Roommate diversity problému, kde:*

- $s = 3$, *preference všech agentů jsou single-peaked a* $n \leq 30$,
- $s = 3$ *a* $n \leq 15$,
- $s = 4$, *preference všech agentů jsou single-peaked a* $n = 8$,

má core stabilní rozdělení do koalic.

Důkaz. Pro každou takovou instanci bylo algoritmicky nalezeno core stabilní řešení. ◀

Pro úplnost popíšeme, jak vygenerovat všechny různé instance Roommate diversity problému daných parametrů (tak jak je to v přiloženém programu a jak bylo generováno pro ověření Věty 4.4).

- Nejdříve musíme vypočítat, kolik různých typů agentů při dané velikosti koalice existuje. Musíme zohlednit, zda navíc požadujeme, aby preference všech agentů byly single-peaked. Typy si uložíme.
- Vygenerování všech různých instancí potom odpovídá úloze *rozdělení stejných objektů do různých krabiček*, známé z předmětu BI-ZDM (objekty jsou agenti a různé krabičky jsou různé typy agentů).
- Celkový počet možností je potom dán vzorcem $\binom{n+k-1}{k-1}$, n je počet agentů a k počet typů agentů (můžeme si to představit jako umístění $k-1$ přepážek).

▶ **Příklad 4.5.** Chceme vygenerovat všechny instance Roommate diversity problému, s parametry $n = 30$, $s = 3$ a aby preference všech agentů byly single-peaked.

Pro velikost koalice $s = 3$ máme celkem šest různých typů červených agentů

$$\frac{1}{3} \succ \frac{2}{3} \succ \frac{3}{3} \succ \frac{0}{3}, \qquad \frac{1}{3} \succ \frac{3}{3} \succ \frac{2}{3} \succ \frac{0}{3},$$
$$\frac{2}{3} \succ \frac{1}{3} \succ \frac{3}{3} \succ \frac{0}{3}, \qquad \frac{2}{3} \succ \frac{3}{3} \succ \frac{1}{3} \succ \frac{0}{3},$$
$$\frac{3}{3} \succ \frac{1}{3} \succ \frac{2}{3} \succ \frac{0}{3}, \qquad \frac{3}{3} \succ \frac{2}{3} \succ \frac{1}{3} \succ \frac{0}{3},$$

a šest typů modrých agentů:

$$\frac{0}{3} \succ \frac{1}{3} \succ \frac{2}{3} \succ \frac{3}{3}, \qquad \frac{0}{3} \succ \frac{2}{3} \succ \frac{1}{3} \succ \frac{3}{3},$$
$$\frac{1}{3} \succ \frac{0}{3} \succ \frac{2}{3} \succ \frac{3}{3}, \qquad \frac{1}{3} \succ \frac{2}{3} \succ \frac{0}{3} \succ \frac{3}{3},$$
$$\frac{2}{3} \succ \frac{0}{3} \succ \frac{1}{3} \succ \frac{3}{3}, \qquad \frac{2}{3} \succ \frac{1}{3} \succ \frac{0}{3} \succ \frac{3}{3}.$$

Navíc chceme, aby preference byly single-peaked. Oranžovou barvou jsou vyznačeny preference, které nejsou single-peaked. Dohromady tedy máme osm různých validních typů agentů.

Počet různých instancí potom odpovídá počtu různých možností, jak rozdělit daných 30 agentů mezi těmito osmi typy:

$$\binom{n+k-1}{k-1} = \binom{30+8-1}{8-1} = \binom{37}{7} = 10295472$$

A tolik instancí zvládne propočítat algoritmus běžící na obyčejném notebooku za necelou noc (a v tomto konkrétním případě vždy nalezl core stabilní řešení).

▶ **Poznámka 4.6.** V uvažovaných typech agentů jsou preference vždy úplné, protože jsme si tak zadefinovali Roomate diversity problem. Přípustné jsou však indiference/lhostejnosti. My zde uvažujeme pouze preferenční relace bez indiferencí, protože stačí najít core stabilní řešení pro všechny instance bez indiferencí (které lze nahradit striktními preferencemi).

Kapitola 5

Přiložená implementace

V této poslední kapitole je nastíněna implementace celé problematiky a algoritmu, která se nachází v přiložených souborech. Dále je vysvětleno, jak se používá přiložený program.

5.1 Implementace

Pro implementaci celé problematiky Roommate diversity problému a hledání algoritmu byl zvolen programovací jazyk C++ bez použití dalších knihoven. Každá třída i funkce ve zdrojovém kódu je dokumentována (komentářem).

Základem implementace jsou třídy `Agent` a `Problem`.

Třída `Agent` reprezentuje agenta: zná tedy svoji barvu a své preference. Preference jsou reprezentovány počtem červených agentů v koalicích, nikoliv poměrem, jak je definováno, protože se tak s nimi pracuje lépe.

Třída `Problem` reprezentuje Roommate diversity problem – zná tedy počet agentů, velikost místnosti, obsahuje vektor `Agentů` a zná jejich rozmístění do koalic. Nejdůležitější metody třídy jsou `isCoreStable()`, která vrací `true`, pokud je rozmístění agentů do koalic core stabilní, a `algorithm()`, která implementuje algoritmus a přímo mění rozmístění agentů do koalic. Třída se také umí chytře vytvořit, a to tak, že si sama vytvoří agenty s náhodnými preferencemi. Použití může vypadat následovně:

```
Problem p(9,3,5);       // 9 agentu, velikost koalic 3, 5 cervenych agentu
p.algorithm();          // *preference jsou jiz nahodne nastaveny
cout << p << endl;
cout << boolalpha << p.isCoreStable() << endl;
```

Samozřejmě, pokud chceme, lze vytvořit instanci problému i s jednotlivými agenty a jejich preferencemi, a to následujícím způsobem:

```
vector<Agent> agents;
agents . push_back( Agent( Color::RED,  vector<int>{2,1,0} ) );
agents . push_back( Agent( Color::RED,  vector<int>{2,1,0} ) );
agents . push_back( Agent( Color::BLUE, vector<int>{0,1,2} ) );
agents . push_back( Agent( Color::BLUE, vector<int>{1,0,2} ) );
Problem p(4,2,2,agents);
```

Samotný algoritmus (metoda `Problem.algorithm()`) je implementován tak, jak je uvedeno v pseudokódu 3. Jeho zdrojový kód je podrobně popsán komentáři.

5.2 Použití programu

Spustitelný program se zkompiluje příkazem `make` a poté je spuštěn příkazem `./a.out`. Po spuštění se uživateli zobrazí interaktivní menu s následujícími možnostmi, které se vybírají napsáním příslušného čísla:

1. Slow solve and print
2. Fast instances solving
3. Brute force
4. Example s=5, n=10, single-peaked, empty core
5. Exit

Nyní se podrobně podíváme na to, co která volba provádí:

1. Slow solve and print: Program hledá core stabilní řešení pro náhodně vygenerované instance zadaných parametrů, po každé vyřešené instanci danou instanci vytiskne, včetně rozdělení agentů do místností. Touto volbou si tedy uživatel může ověřit, že algoritmus skutečně vrací core stabilní řešení. Na začátku je nutné zadat celkový počet agentů, velikost koalice, počet červených agentů a zda mají být všechny preference single-peaked. Po každé zobrazené vyřešené instanci čeká program na vstup od uživatele, zda má pokračovat, nebo se vrátit do menu. Algoritmus samozřejmě nemusí najít core stabilní řešení, pokud jej daná instance nemá.

2. Fast instances solving: Program opět hledá core stabilní řešení pro náhodně vygenerované instance zadaných parametrů, tentokrát je však netiskne jako v předchozí volbě a automaticky pokračuje. Zobrazuje pouze počet vyřešených instancí. Tímto způsobem si tedy uživatel může vyzkoušet rychlost algoritmu na různých instancích. Počítá se s ukončením programu `Ctrl-C`.

3. Brute force: Program zkouší vyřešit všechny různé instance daných parametrů. Zadávané parametry jsou počet agentů, velikost koalice a zda mají být preference single-peaked. Všech instancí je $\binom{n+k-1}{k-1}$, kde n je počet agentů a k je počet různých typů agentů. Je tedy doporučeno zkoušet to pro minimální parametry, například:

- Počet agentů = 6, velikost koalice = 3.
- Počet agentů = 9, velikost koalice = 3, single-peaked.
- Počet agentů = 8, velikost koalice = 4, sinle-peaked.
- Velikost koalice = 2.

4. Example s=5, n=10, single-peaked, empty core: Program načte instanci z Příkladu 4.1. Poté vyzkouší všechna možná rozdělení do koalic, ověří, že žádné rozdělení není core stabilní, a nakonec instanci zobrazí.

5. Exit: Ukončí běh programu.

Závěr

V této práci jsme představili nový randomizovaný algoritmus na rychlé hledání core stabilních řešení u Roommate diversity problému. Ukázali jsme single-peaked instanci Roommate diversity problému s deseti agenty a velikostí koalic pět, která má prázdné core.

Pomocí představeného algoritmu jsme ověřili, že každá instance Roommate diversity problému, kde

- velikost koalic $= 3$, preferenční relace všech agentů jsou single-peaked a počet agentů ≤ 30,
- velikost koalic $= 3$ a počet agentů ≤ 15,
- velikost koalic $= 4$, preferenční relace všech agentů jsou single-peaked a počet agentů $= 8$,

má core stabilní řešení.

Dále se domníváme, že každá instance Roommate diversity problému, kde

- velikost koalic $= 3$,
- velikost koalic $= 4$ a preferenční relace všech agentů jsou single-peaked,

má core stabilní řešení. Ačkoliv se nám toto nepodařilo dokázat, v práci uvádíme silné argumenty, proč by to tak mělo být.

Jako otevřený problém tedy zůstává formálně dokázat tuto domněnku.

Literatura

[1] Jose Alcalde and Pablo Revilla. Researching with Whom? Stability and Manipulation. *Journal of Mathematical Economics*, 40(8):869–887, 2004. URL: `https://EconPapers.repec.org/RePEc:eee:mateco:v:40:y:2004:i:8:p:869-887`.

[2] Coralio Ballester. NP-completeness in hedonic games. *Games and Economic Behavior*, 49(1):1–30, 2004. `doi:https://doi.org/10.1016/j.geb.2003.10.003`.

[3] Niclas Boehmer and Edith Elkind. Stable roommate problem with diversity preferences, 2020. `arXiv:2004.14640`.

[4] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia. *Handbook of Computational Social Choice*. Cambridge University Press, USA, 1st edition, 2016.

[5] Robert Bredereck, Jiehua Chen, Ugo Paavo Finnendahl, and Rolf Niedermeier. Stable roommates with narcissistic, single-peaked, and single-crossing preferences. *CoRR*, abs/1903.05975, 2019. URL: `http://arxiv.org/abs/1903.05975`, `arXiv:1903.05975`.

[6] Robert Bredereck, Edith Elkind, and Ayumi Igarashi. Hedonic diversity games. In Edith Elkind, Manuela Veloso, Noa Agmon, and Matthew E. Taylor, editors, *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19, Montreal, QC, Canada, May 13-17, 2019*, pages 565–573. International Foundation for Autonomous Agents and Multiagent Systems, 2019. URL: `http://dl.acm.org/citation.cfm?id=3331741`.

[7] Michel Le Breton, Ignacio Ortuño-Ortin, and Shlomo Weber. Gamson's law and hedonic games. *Soc. Choice Welf.*, 30(1):57–67, 2008. `doi:10.1007/s00355-007-0220-9`.

[8] Andreas Darmann, Edith Elkind, Sascha Kurz, Jérôme Lang, Joachim Schauer, and Gerhard J. Woeginger. Group activity selection problem. In Paul W. Goldberg, editor, *Internet and Network Economics - 8th International Workshop, WINE 2012, Liverpool, UK, December 10-12, 2012. Proceedings*, volume 7695 of *Lecture Notes in Computer Science*, pages 156–169. Springer, 2012. `doi:10.1007/978-3-642-35311-6\_12`.

[9] David Gale and Lloyd S. Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962. URL: `http://www.jstor.org/stable/2312726`.

[10] Chien-Chung Huang. Two's company, three's a crowd: Stable family and threesome roommates problems. In Lars Arge, Michael Hoffmann, and Emo Welzl, editors, *Algorithms – ESA 2007*, pages 558–569, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[11] Robert W. Irving. An efficient algorithm for the "stable roommates" problem. *Journal of Algorithms*, 6(4):577–595, 1985. `doi:https://doi.org/10.1016/0196-6774(85)90033-1`.

[12] Cheng Ng and Daniel S. Hirschberg. Three-dimensional stable matching problems. *SIAM Journal on Discrete Mathematics*, 4(2):245–252, 1991. `arXiv:https://doi.org/10.1137/0404023`, `doi:10.1137/0404023`.

[13] Walid Saad, Zhu Han, Tamer Basar, Mérouane Debbah, and Are Hjørungnes. Hedonic coalition formation for distributed task allocation among wireless agents. *IEEE Trans. Mob. Comput.*, 10(9):1327–1344, 2011. `doi:10.1109/TMC.2010.242`.

Obsah přiloženého média

- readme.txt .... návod ke spuštění programu
- thesis.pdf .... text práce ve formátu PDF
- Makefile .... soubor Makefile
- src .... zdrojové kódy implementace