Univerzita Hradec Králové
Fakulta informatiky a managementu
Katedra informatiky a kvantitativních metod

Teorie her v managementu

Bakalářská práce

Autor: Robert Míča
Studijní obor: Aplikovaná informatika

Vedoucí práce: doc. RNDr. Pavel Pražák, Ph.D.

Hradec Králové Srpen 2020

Prohlášení:

Prohlašuji, že jsem bakalářskou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 14.8.2020 Robert Míča

Míča

Poděkování:

Děkuji vedoucímu bakalářské práce doc. RNDr. Pavlu Pražákovi, Ph.D. za jeho ochotu, čas a cenné rady, které mi pomohly ke zpracování této práce.

Anotace

Tato bakalářská práce se zaměřuje na problematiku určitých typů teorie her. Teoretická část práce se věnuje studiu základních poznatků z teorie her a analýze modelů, které mohou pomoci ke zvolení té nejlepší strategie v konfliktních a rozhodovacích situacích. V praktické části jsou k těmto modelům vytvořeny programy ve vývojovém prostředí MATLAB a je popsáno jejich ovládání a možnosti využití. V práci je také zahrnuta část seznamující čtenáře s prostředím MATLAB. Vytvořené programy je možno využít pro libovolné zadání od uživatele a jejich funkčnost je posléze ověřena na praktických příkladech.

Klíčová slova:

teorie her, matice, program, strategie, sedlový bod

Annotation

Game Theory in Management

This bachelor thesis focuses on the phenomenon of certain types of game theory. The theoretical part of this thesis is dedicated to the study of basic knowledge of the game theory and analysis of models that can help to choose the right decision in conflict situations and decision-making. In the practical part the programs in the MATLAB development environment are created for these models their control, and possibilities of use are described. The work includes a part acquainting the reader with the MATLAB environment. Developed programs can be used for any user's input and their functionality is verified on the practical examples.

Key words:

game theory, matrix, program, strategy, saddle point.

Obsah

1 Úvod

V dnešním světě se lidé každý den dostávají do situací, kdy se musí rozhodnout, jak se v daném okamžiku zachovat. V zájmu každého člověka, který se do rozhodovací situace dostane, je zvolit takové rozhodnutí, které pro něj bude nejvýhodnější a přinese mu největší užitek. Při volbě strategie se může člověk rozhodnout úplně bezmyšlenkově, tedy bez promyšlení toho, jaké bude mít jeho rozhodnutí dopad, avšak rozumný člověk si nejdříve rozebere všechny možnosti volby a na základě dostupných informací a faktů se snaží vybrat takové rozhodnutí, které pro něj bude nejlepší a nejpřínosnější.

V ekonomické praxi jsou takové rozhodovací situace klíčové pro zisk a úspěch firmy. Pokud se firmy špatně rozhodují, mohou ztratit část možného zisku a oslabit svou pozici na trhu. Firmy mezi sebou svádí každý den boj a vzájemně reagují na rozhodnutí jejich konkurence. Pro určení správného rozhodnutí, jak se v takových situacích zachovat, byla vytvořena vědní disciplína, která je dnes nazvaná jako teorie her, viz [1], [3], [5], [11].

2 Cíl práce a metodika zpracování

Tato práce se zaměřuje na seznámení čtenáře se základy z teorie her a s vybranými strategiemi, které mohou být prakticky využité v různých firemních sporech a konfliktních situacích. Nejdříve budou rozebrány základní poznatky z teorie her a následně se určité strategie z těchto poznatků převedou do algoritmu, který bude mít zdrojový kód vytvořený v programu MATLAB. Finální program bude srozumitelný pro koncového uživatele a bude možné zadávat různé hodnoty do vstupů. Program bude vyhodnocovat výsledky šetření podle zadaných vstupů a celý bude prezentován na několika různorodých řešeních, která budou ověřovat metody z teoretické části.

Postup vypracování této práce se bude řídit podle následujících bodů:

- výzkum bude prováděn pomocí kvantitativních metod a bude se věnovat oboru informatiky,
- pro studium bude využita odborná literatura zaměřující se na téma teorie her a modely, které rozebírají různé strategie,
- na základě poznatků z teorie her bude vybráno několik druhů modelů, ke kterým bude vytvořen program ověřující jejich pravdivost,
- pro ověření modelů budou vytvořeny algoritmy v programu MATLAB, na které bude možno použít libovolné praktické příklady zadání,
- vytvořený program bude vyzkoušen na více variantách vstupů,
- výstupy programu budou porovnány a bude vyhodnoceno, zda program funguje správně.

3 Teoretická část – základní poznatky z teorie her

3.1 *Úvod do teorie her*

Mezi první příspěvky do teorie her lze zařadit již práci [7], ve které francouzský matematik vytvořil model pro stanovení rozsahu výroby produktu na trhu, kde existují pouze dvě konkurenční firmy (duopol) a následně také pro více konkurenčních firem (oligopol). Jeho modely se dnes vyskytují téměř v každé učebnici ekonomie.

Základní poznatky z teorie her se začaly vyskytovat v první polovině 20. století v pracích [8], [9], [10]. Podle [1] tyto práce definovaly určité postupy užívané v dnešní teorii her a dokazovaly první výsledky aplikace těchto postupů. Moderní teorie her byla však poprvé představena v [4] jako vědní disciplína.

Dnes je teorie her využívána v mnoha odvětvích lidské činnosti. Najde uplatnění v ekonomii, kdy se jedná o souboj dvou či více firem, které se snaží maximalizovat vlastní zisk a minimalizovat možnou ztrátu. Dále má své využití v oblasti společenských her, kde je díky ní možno nalézt nejlepší strategii, která dovede hráče k vítězství, nebo mu alespoň maximalizuje šance na jeho dosažení nebo například v politické sféře, kde může vypomoci ke správnému rozhodnutí při hlasování. Teorie her je tedy prostředek k řešení všemožných konfliktů a pomocí výpočtů pomáhá nalézt konkrétní strategie pro určité účastníky hry, viz [1], [11].

3.1.1 Definice hry

Podle [1] je možné teorii her definovat jako analýzu konfliktních rozhodovacích situací, které jsou řešeny využitím specifických matematických modelů. Dva nejdůležitější matematické modely pro řešení konfliktů jsou hra v normálním a hra v rozvinutém tvaru. Modely se vyznačují tím, že se zaměřují na podstatné charakteristiky konfliktu a nesledují ty nepodstatné.

3.1.2 Základní pojmy

Hra je simulací určitého konfliktu nebo rozhodovací situace z praxe.Hráč je pak účastník daného konfliktu, kterým může být buď jedinec, firma nebo třeba politická strana. Každý inteligentní hráč je ten, který se snaží nalézt svoji optimální strategii pro hru, tedy tu volbu, která je pro něj nejvýhodnější. Takovou strategii si hráč vybírá z prostoru strategií, který značí seznam všech možných alternativ, ze kterých může hráč volit. Všechny základní pojmy jsou vypsány v následující tabulce.

Teorie her	Ekonomická realita
Hra	Rozhodovací situace, konflikt
Hráč	Účastník konfliktu, firma, jedinec, politická strana
Strategie	Konkrétní alternativa, kterou si může hráč v průběhu hry zvolit
Optimální strategie	Hráčem zvolená strategie, která je pro něj nejvýhodnější
Prostor strategií	Seznam všech možných alternativ, které jsou hráči dostupné
Výplatní funkce	Výhra hráče, která se odvíjí od zvolené strategie
Inteligentní hráč	Účastník konfliktu, který má dokonalé informace o hře a maximalizuje svou výhru

Obrázek 1 Charakteristika základních pojmů, [1]

3.2 Hry s konstantním součtem

3.2.1 Hra v normálním tvaru

Hra v normálním tvaru, jindy také nazývána jako hra ve strategickém tvaru, je v [1] popsána jako hra ve strategickém tvaru a je určena následujícími třemi množinami:

- Množina hráčů se označuje $\{1,2,\ldots,N\}$ v případě hry dvou hráčů, kdy lze pro zjednodušení použít označení hráč 1 a hráč 2,
- Množina prostorů strategií $\{X_1, X_2, \ldots, X_N\}$, kde X_i sindexovým číslem označuje prostor strategií určitého hráče. V případě hry dvou hráčů lze označení zjednodušit na X pro prostor strategií prvního hráče a Y pro prostor strategií druhého hráče. Konkrétní strategie je v případě dvou hráčů x pro hráče 1 a y pro hráče 2,
- Množina výplatních funkcí všech hráčů$\{f_1(x_1, x_2, \ldots, x_N), f_2(x_1, x_2, \ldots, x_N), \ldots, f_N(x_1, x_2, \ldots, x_N)\}$, kdy k definování výplatních funkcí lze využít kartézský součin prostoru strategií, neboť je třeba najít výsledky pro všechny možné kombinace strategií. V případě hry dvou hráčů lze použít označení $f_1(x,y)$ pro výplatní funkci prvního hráče a $f_2(x,y)$ pro výplatní funkci druhého hráče.

Jak již z názvu plyne, tato kapitola se zabývá hrami s konstantním součtem. Podle [11] lze tento typ popsat jako hru, která popisuje konflikt, ve kterém nemá za žádných okolností smysl spolupracovat se soupeřem. Platí, že co jeden hráč získá, druhý ztratí. K může být libovolné reálné číslo a platí výplatní funkce:

$$f_1(x,y) + f_2(x,y) = K$$

K můžeme ve všech případech transformovat na ekvivalentní hru s nulovým součtem kdy je $K = 0$. Přičtením libovolné konstanty ke všem hodnotám výplatní funkce zůstane řešení beze změny. V případě hry s nulovým součtem tedy platí výplatní funkce:

$$f_1(x,y) = -f_2(x,y)$$

Z výplatní funkce pro hru s nulovým součtem tedy vyplývá, že stačí sledovat pouze výhru prvního hráče, jelikož jak je řečeno v [1]: *„výhra druhého hráče je výhra prvého hráče s opačným znaménkem".*

Pro znázornění množiny všech možných výher s nulovým součtem se využívá matice. Jako jeden z příkladů matice, popisovaný v [1] pro tuto problematiku, lze uvést názorný příklad, který zobrazuje matici $\boldsymbol{A} = (aij), i = 1, 2, \ldots, m;\ j = 1, 2, \ldots, n$. Pro hráče 1 vybíráme strategie z řádků matice a pro hráče 2 vybíráme ze sloupců matice. Dostaneme souřadnice strategie, která je výplatní funkce hráče 1 a zároveň výplatní funkce hráče 2 se záporným znaménkem.

Hráči se vždy snaží nalézt nejlepší strategii, tedy tu strategii, která jim přinese výhru. Pro hráče 1 platí, že si nikdy nevybere řádek, který má všechny prvky menší než odpovídající prvky v jiném řádku (hráč 1 vítězí, pokud výplatní funkce jeho výplatní funkce je kladná) a pro hráče 2 naopak platí, že si nikdy nevybere sloupec, ve kterém jsou všechny prvky větší než v jiném sloupci (hráč 2 vítězí, pokud výplatní funkce prvního hráče je záporná). Tento pojem se nazývá silná dominovanost a platí, že hráč si nikdy nezvolí silně dominovanou strategii, jelikož ta značí jeho prohru.

$$
A = \begin{bmatrix} a_{11}\ a_{12} \ldots\ a_{1n} \\ a_{21}\ a_{22} \ldots\ a_{2n} \\ . \\ . \\ . \\ a_{m1}\ a_{m2} \ldots\ a_{mn} \end{bmatrix}
$$

3.2.2 Nashova rovnováha

K určení optimální strategie hráčů se využívá tzv. Nashova rovnováha. Pro lepší představu je možno na základě poznatků z [11] Nashovu rovnováhu charakterizovat tak, že pokud se například hráč 1 bude držet své optimální strategie a hráč 2 se nebude držet své optimální strategie, šance na výhru hráče 1 se buď zvýší nebo zůstane stejná a šance na výhru hráče 2 se buď sníží nebo zůstane stejná. Znamená to tedy, že dodržení optimální strategie nemůže hráči v žádném případě uškodit.

Nashova rovnováha nastane v případě, kdy najdeme strategie obou hráčů $x^0 \in X$ hráče 1 a $y^0 \in Y$ hráče 2, pro které platí funkce, které jsou použity v [1]:

$$f_1(x, y_0) \leq f_1(x_0, y_0)$$
$$f_2(x_0, y) \leq f_2(x_0, y_0)$$

V případě her s nulovým součtem se pak

$$f_1(x, y) = f(x, y)$$
$$f_2(x, y) = -f(x, y)$$

a výše uvedené nerovnosti můžeme zapsat jako

$$f(x, y_0) \leq f(x_0, y_0) \leq f(x_0, y)$$

Tyto strategie se nazývají rovnovážnými strategiemi.

3.2.3 Rovnovážné strategie

Pro získání Nashovy rovnováhy je potřeba nalézt v matici tzv. sedlový prvek matice. Sedlový prvek matice můžeme popsat jako prvek v matici, který je největší číslo ve svém sloupci a nejmenší ve svém řádku. Pokud nalezneme jeden nebo více sedlových prvků, můžeme využít metodu Nashovy rovnováhy a použít jeden z nalezených prvků. Pokud se v matici žádný sedlový prvek nevyskytuje, znamená to, že nemůžeme využít ryzích strategií. V tomto případě je potřeba využít jiných strategií.

Vyhledávání sedlového bodu v matici lze ukázat na třech příkladech prezentující všechny možnosti, které mohou při hledání sedlového bodu nastat. V případě matice A lze zjistit, že při vyhledání všech sloupcových maxim a řádkových minim je nalezen jeden sedlový bod, a to na souřadnicích (1,3). Tento bod značí optimální rovnovážnou strategii.

$$A = \begin{bmatrix} 6 & (8) & [(3)] \\ [-7] & -6 & -6 \\ (7) & 7 & [-2] \end{bmatrix}$$

V případě matice B je nalezeno více sedlových bodů, konkrétně jeden na pozici (1,3) a druhý na pozici (3,3). Oba tyto body mohou být označené jako optimální strategie hry.

$$B = \begin{bmatrix} (6) & 4 & [(2)] \\ [0] & 6 & 1 \\ 3 & (9) & [(2)] \end{bmatrix}$$

V případě matice C není nalezen žádný sedlový bod v matici a značí to, že tato hra bude mít řešení ve smíšených strategiích.

$$B = \begin{bmatrix} (6) & 4 & [2] \\ [0] & 3 & (4) \\ 3 & (7) & [1] \end{bmatrix}$$

Pro hledání sedlového bodu v matici je vytvořen program č. 1, který je popsaný v praktické části práce. Zadáním vstupů v podobě maticí lze ověřit, zda v matici jsou sedlové body a pokud nejsou, program vypočítá řešení ve smíšených strategiích.

3.2.4 Smíšené strategie

Smíšené (pravděpodobnostní) strategie, které se nazývají smíšené rozšířené maticové hry, se liší od těch ryzích tím, že se v nich musí náhodně vybírat z prostoru strategií, jelikož nelze nalézt strategii, která by byla výhodnější než ty ostatní. Podle [1] se pro tyto strategie se využívají vektory pravděpodobností, s jakou hráči zvolí jednu ze svých strategií. Prostory strategií hráčů jsou:

$$X = \left\{ x \in R^m; \sum_{i=1}^{m} x_i = 1, x \geq 0 \right\}$$

$$Y = \left\{ y \in R^n; \sum_{j=1}^{n} y_j = 1, y \geq 0 \right\}$$

Pokud se jedná o hru s konstantním součtem, postačí opět pouze výplatní funkce prvního hráče, která je:

$$f(x, y) = \sum_{i=1}^{m} \sum_{j=1}^{n} x_i a_{ij} y_j = x^T A y$$

Pro všechny maticové hry platí tzv. základní věta maticových her, která zní:

„Každá maticová hra má Nashovo rovnovážné řešení ve smíšených strategiích"

Tato věta značí, že v každé matici A existují dva vektory x a y (kroužkem se označuje, že jde o rovnovážné strategie), pro které platí nerovnice:

$$x^T A y^0 \leq x^{0T} A y^0 \leq x^{0T} A y^9$$

3.3 Hry s nekonstantním součtem

3.3.1 Kooperativní a nekooperativní hry

U mnoha konfliktů v praxi se může stát, že nastanou situace, kdy zájmy hráčů nejsou v úplném rozporu. Takové konflikty se rozebírají například v [5] a jsou nazývané jako neantagonistické. V případě takových problémů nastává situace, kdy výhra jednoho z hráčů nemusí znamenat prohru druhého hráče. U takových her se rozlišuje, zda se jedná o hru kooperativní nebo nekooperativní. Pokud se jedná o hru kooperativní, hráčům se vyplatí v konfliktu spolupracovat a analogicky v případě hry nekooperativní se hráčům spolupráce nevyplatí.

3.3.2 Dvoumaticové hry

Pro neantagonistické konflikty se využívá model dvoumaticové hry. Hra je určena maticemi $A\ a\ B$. Každá matice charakterizuje výplatní funkci jednoho z hráčů v konfliktu. Na rozdíl od her s konstantním součtem neplatí přímý vztah mezi hodnotami výher obou hráčů. Nelze tedy použít větu definující výhru jednoho hráče jako výhru druhého hráče s opačným znaménkem a nikdy není nalezen sedlový prvek. U nekonstantních her je vždy využito hledání řešení pomocí smíšených strategií, viz. [1].

$$\text{A} = \begin{bmatrix} a_{11}\ a_{12} \dots\ a_{1n} \\ -a_{21}\ a_{22} \dots\ a_{2n} \\ . \\ . \\ . \\ a_{m1}\ a_{m2} \dots\ a_{mn} \end{bmatrix} \qquad \text{B} = \begin{bmatrix} b_{11}\ b_{12} \dots\ b_{1n} \\ -b_{21}\ b_{22} \dots\ b_{2n} \\ . \\ . \\ . \\ b_{m1}\ b_{m2} \dots\ b_{mn} \end{bmatrix}$$

Hodnota výplatní funkce prvního hráče je rovna prvku a_{ij} a hodnota výplatní funkce druhého hráče pak prvku b_{ij}. $i\ (i = 1, 2 \dots, m)\ a\ j\ (j = 1, 2 \dots, m)$ označujícího strategie prvního a druhého hráče.

3.3.3 Nekooperativní hra

U nekooperativních her se tedy hráči nesnaží za žádné situace spolupracovat. Pro tento druh her lze použít vzorec, který byl již použit u her s konstantním součtem v kapitole 4.2.

$$f_1(x, y_0) \leq f_1(x_0, y_0)$$
$$f_2(x_0, y) \leq f_2(x_0, y_0)$$

Pro vyhledání rovnovážného řešení v ryzích strategiích je potřeba vyhledat v první matici všechna sloupcová maxima a v druhé matici všechna řádková maxima. Pokud je určitá dvojice prvků označena hráčem A i B, bylo nalezeno rovnovážné řešení. Při hledání rovnovážného řešení mohou podle [1] nastat tyto případy:

- Bylo nalezeno jedno rovnovážné řešení, které je využito jako strategie pro oba hráče,
- Bylo nalezeno více než jedno rovnovážné řešení a jedno ze všech rovnovážných řešení je pro oba hráče nejvýhodnější. Takové řešení dominuje ostatním řešením a je zvoleno jako strategie pro oba hráče,
- Bylo nalezeno více než jedno rovnovážně řešení a neexistuje žádné řešení, které by bylo pro oba hráče nejvýhodnější, tedy existují pouze taková řešení, která jsou vždy výhodná pouze pro jednoho z hráčů. V takovém případě hra nemá rovnovážné řešení v ryzích strategiích a je potřeba použít smíšeného rozšíření dvoumaticové hry.

Při aplikaci smíšeného rozšíření maticových her je možno pro nalezení řešení využít metod nelineárního programování. Prostory strategií jsou dány jako:

$$X = \left\{ x \in R^m; \sum_{i=1}^{m} x_i = 1, x \geq 0 \right\}$$

$$Y = \left\{ y \in R^n; \sum_{j=1}^{n} y_j = 1, y \geq 0 \right\}$$

Výplatní funkce pak mají tvar:

$$f_1(x, y) = \sum_{i=1}^{m} \sum_{j=1}^{n} x_i a_{ij} y_j = x^T A y$$

$$f_2(x, y) = \sum_{i=1}^{m} \sum_{j=1}^{n} x_i b_{ij} y_j = x^T B y$$

Je potřeba maximalizovat rovnici:

$$p^T (A + B) q - e^T p - f^T q$$

Při splnění podmínek:

$$A q \leq e$$

$$B^T p \leq f$$

$$p \geq 0$$

$$q \geq 0$$

Při použití programu pro výpočet rovnovážného řešení ve smíšených strategiích se do zadané rovnice opakovaně zadávají všechny přípustné hodnoty z matice a jako výsledek jsou zobrazena všechna rovnovážná řešení pro oba hráče.

3.3.4 Vězňovo dilema

I přes komplexnost Nashovy rovnováhy, je možné se dostat do situací, kdy může dojít k problémům a využití optimálního řešení nemusí být výhodné pro žádného z hráčů. Tato situace se dá dobře vysvětlit na příkladu, který je obecně nejznámější z problémů v teorii her a je známý jako vězňovo dilema. Různé rozbory příkladů, které popisující vězňovo dilema se vyskytují například v [1], [5], [11], [12].

V tomto příkladu jsou dva vězni souzeni za stejný zločin, mají možnost se přiznat nebo nepřiznat a vzájemně nevědí, jak se rozhodují. Pokud se vězeň A přizná a vězeň B nepřizná, je vězni A udělen menší trest. Pokud se ani jeden z vězňů nepřizná, bude jim udělen nižší trest, než v případě, kdy by se oba dva přiznali, jelikož pro odsouzení není dostatečný počet důkazů. Situace je znázorněna na následujícím příkladu z [12].

		Hráč 2	
		zapírat	*přiznat se*
Hráč 1	*zapírat*	$(-3,-3)$	$(-10,-1)$
	přiznat se	$(-1,-10)$	$(\mathbf{-5},\mathbf{-5})$

Obrázek 2 Příklad vězňova dilema, [12]

Čas strávený ve vězení je definován jako záporná hodnota, jelikož pro hráče nepřináší zisk a snaží se, aby jeho hodnota byla co nejblíže k nule. Vězni by tedy správně měli zvolit strategii přiznat se ke zločinu, nicméně v situaci, kdy se oba vězni přiznají je pro ně výsledná doba ve vězení delší než pokud se ani jeden nepřizná. Nepřiznat se k zločinu by však bylo v rozporu s Nashovou rovnováhou, jelikož přiznáním si vězeň může snížit délku svého trestu.

Řešení, kdy se oba vězni přiznali, je tedy v souladu s rovnovážnými strategiemi. Tedy změnou rozhodnutí si vězeň nemůže polepšit, ale není nejvýhodnější, jelikož pokud se oba hráči nepřiznají, bude pro oba čas strávený ve vězení kratší. Je tedy vidět, že pokud oba hráči jednají racionálně, dostanou se do situace, která je pro oba nevýhodná.

V praxi si lze příklad vězňova dilema představit jako spor dvou firem, které spolu během vyjednávání uzavřou dohodu (takzvané kartelové dohody), kterou mohou následně dodržet nebo porušit. Pokud jedna z firem dohodu poruší, získá ve sporu výhodu. V takových sporech platí, že nezávislá dohoda ve výsledku, je tedy bezvýznamná a nepřináší výhody oproti sporu bez dohody.

3.3.5 Kooperativní hry

Může nastat situace, kdy je pro hráče výhodné vzájemně spolupracovat. V takovém případě se jedná o kooperativní hry. Hráč, má zájem spolupracovat pouze v takovém případě, kdy pro něj spolupráce znamená větší hodnotu výhry. Pro uzavření spolupráce je potřeba, aby zvýšení výhry platilo pro všechny účastníky spolupráce. Příklady, které se věnují spolupráci hráčů jsou probírané například v [1], [5], [12].

Pro zhodnocení, zdali je ve hře spolupráce výhodná, je nejdříve potřeba zjistit hodnotu zaručené výhry v případě nekooperativní hry u každého hráče. Hodnota zaručené výhry hráče jedna označena jako $v(1)$ a hodnota zaručené výhry druhého hráče je označena jako $v(2)$. Podle [1] maximální možná hodnota výhry, která je dosažitelná spoluprací je pak označená vzorcem:

$$v(1,2) = \max_{x \in X\ y \in Y} [f_1(x, y) + f_2(x, y)]$$

Pokud pak platí:

$$v(1,2) > v(1) + v(2)$$

je pro hráče vzájemná spolupráce výhodná. Optimální strategie je pak taková, kdy při součtu hodnot v matici hráče A a hráče B dostaneme maximální hodnotu $v(1,2)$. Při spolupráci dvou hráčů tedy každý hráč dostane novou hodnotu výhry, která je označena jako a_1 pro prvního hráče, resp. a_2 pro druhého hráče a pro hodnoty platí vzorec:

$$a_1 + a_2 = v(1,2)$$

$$a_1 \geq v(1)$$

$$a_2 \geq v(2)$$

Množina všech variant rozdělení $a1, a2$, které splňují předchozí vzorce, je nazvána jako jádro hry. Podle [1] závisí rozdělení přebytečné hodnoty nad zaručenou výhru obou hráčů na vzájemné dohodě obou stran. Jako doporučený postup se uvádí rozdělení přebytečné hodnoty výhry ve stejném poměru pro oba hráče.

V praktické části je zpracován příklad č. 2, který pomáhá v rozhodnutí, zdali se hráčům vzájemná spolupráce v konfliktu vyplatí nebo ne. V programu se zadává matice A, která udává prostor strategií pro prvního hráče a matice B, která udává prostor strategií pro druhého hráče. Výstupem programu je porovnání hodnoty zaručené výhry pro oba hráče v případě, že spolu nebudou spolupracovat s možnou výhrou v případě spolupráce.

3.4 Hry v rozvinutém tvaru

3.4.1 Definice

Hry v normálním tvaru se liší oproti hrám v rozvinutém tvaru tím, že všechny konfliktní situace probíhají v po sobě jdoucích tazích. Hráč má tedy možnost svou strategii měnit na základě tahu (rozhodnutí) jeho protihráče. Hry v rozvinutém tvaru se často nazývají také jako „tahové hry" jak je řečeno v [5]. Mezi tento typ her řadíme mnoho salónních her, přičemž matematický rozbor některých salónních her lze považovat za počátky teorie her. Jako příklad her v rozvinutém tvaru lze uvést šachy, dámu, go, NIM, poker a mnoho dalších.

3.4.2 Strom hry

Pro modelování hry v rozvinutém tvaru používáme tzv. strom hry, který je jinak známý také jako „Kuhnův strom", jak se zmiňuje v [2]. Jak uvádí [5], strom hry zachycuje všechny situace, které ve hře mohou nastat.

Strom je graf, který se skládá z množiny uzlů a hran, má vždy jeden počáteční uzel, tzv. kořen, který značí první tah hry a obvykle má také několik koncových uzlů, které značí konec hry. Každý uzel značí jednu situaci a každá hrana jeden z možných tahů. Jestliže se hráč, který je na řadě, rozhodne pro nějaký tah, navodí novou situaci, v níž se rozhoduje druhý hráč – této situaci opět odpovídá jistý uzel stromu spojený s předchozí hranou. Občas se mohou vyskytnout tzv. průchozí uzly, které hráči nabízejí pouze jednu volbu, jak pokračovat ve hře, viz. [5].

3.4.3 Typy tahových her

Podle [2] tahové hry můžeme rozdělit na dvě kategorie. První z nich jsou tahové hry s dokonalou informací. Pro tyto hry platí to, že teoreticky by měl být znám výsledek již před zahájením hry, jelikož oba hráči mají na začátku hry dokonalé informace o hře, tedy již na začátku hry je možné navrhnutím stromu hry nalézt vítěznou strategii. Podle [1] pro všechny tahové hry s dokonalou informací platí věta:

„Každá konečná hra v rozvinutém tvaru s dokonalou informací má řešení v ryzích strategiích.“

Příklad, který se zabývá tahovými hrami s dokonalou informací, je rozebrán v praktické části jako příklad č. 3. V tomto příkladu vystupují dvě konkurenční firmy, z nichž jedna již nějakou dobu působí na trhu a druhá se rozhoduje, zdali na trh vstoupit nebo ne.

Tento postup lze aplikovat na některé hry, které nemají příliš složitý strom hry jako je například hra NIM. Problém nastává v situaci, kdy je ve hře příliš velké množství možností tahů.

Teoreticky, by i taková hra jako šachy, měla mít už před začátkem jasného vítěze, jelikož je to hra konečná, kdy každý hráč má ve svém tahu konečný počet strategií. Jenže již po prvních dvou tazích hry může hráč zvolit stovky různých kombinací pozic na šachovnici. Díky tomu je ve hře příliš velké množství variant k sestavení celého stromu hry. Do dnešního dne nikdo nedokázal výsledek hry vypočítat. Viz. [1], kde je uvedeno, že ve hře šachy tedy každý hráč sestavuje pouze částečný strom hry a sleduje okolní pozice, které mohou v dalších tazích nastat.

Druhá kategorie jsou hry s nedokonalou informací. V tomto typu her nemá hráč dokonalou informaci o všech možnostech soupeřových strategiích. Do této kategorie můžeme zařadit například karetní hry, ve kterých nelze vidět, jaké karty má soupeř v ruce.

3.5 Dokonalý a nedokonalý trh

3.5.1 Základní poznatky o trzích

Pro charakterizaci chování hráčů na trhu se využívá dokonalých a nedokonalých trhů. Podle [11] dokonalý trh vychází z následujících předpokladů:

- nakupující i prodávající jsou dokonale informovaní,
- při změně dodavatele jsou nulové náklady,

- jedná se o homogenní produkt,
- existuje velké množství prodávajících,
- neexistují žádné státní regulace.

Z těchto bodů je zřejmé, že v praxi není možné všechny tyto podmínky splnit, a proto se začaly vytvářet modely nedokonalých trhů.

Nejběžnější modely nedokonalých trhů jsou monopolistická konkurence a oligopol. Monopolistická konkurence narušuje pravidlo dokonalých trhů o homogennosti produktu. Hráči se snaží cenu produktu snižovat nebo zvyšovat oproti konkurenci a tím zajistit lepší zisk z výsledného počtu prodaných produktů, který je ovlivněn počtem prodaných kusů a jejich cenou.

Oligopol narušuje pravidlo o dostatečně velkém množství prodávajících. Na trhu je malé množství firem prodávajících produkt a ty pak mohou kontrolovat situaci na trhu. Mezi těmito firmami je pak často využívána teorie konfliktních rozhodovacích situací a vznikají tajné dohody o cenách, již dříve zmíněné jako kartelové dohody, při kterých firmy koordinují ceny tak, aby společně dosáhly maximálního zisku, který si pak mezi sebou rozdělují. Extrémním případem oligopolu je pak monopol kdy jedna firma ovládá celý trh, má velké množství nakupujících a může tak libovolně navyšovat cenu produktu.

3.5.2 Modely oligopolu

Modely oligopolu se rozdělují na kooperativní a nekooperativní. U kooperativních se předpokládá, že firmy budou využívat vzájemné komunikace při stanovení ceny. U nekooperativních poté, že jedna z firem změní strategii prodeje a ostatní firmy budou na změnu reagovat. Jako jeden z prvních se modely nedokonalých trhů začal zabývat [7], když vytvořil model pro duopol (oligopol pro dva hráče). Model využívá koncepce Nashovy rovnováhy z teorie nekooperativních her.

3.5.3 Cournotův model duopolu

Tento model je popsán v [1] a jeho praktické využití je rozebráno v programu č. 4, ve kterém je snaha o to, převést tento model duopolu do programu, který by dokázal řešit konkrétní příklady. V tomto modelu se vyskytují dvě firmy, které budou pro lepší přehlednost nazvané jako firma A a firma B. Strategická proměnná je objem produkce a objemy produkce pro firmy, které se konfliktu účastní, jsou:

q_1 pro objem produkce firmy A

q_2 pro objem produkce firmy B

V modelu se tržní cena výrobku odvíjí od celkového objemu produkce obou zúčastněných firem a platí:

$$p = g(q_1 + q_2)$$

Pokud se zvyšuje objem produkce, cenová funkce produktu klesá a platí tedy, že čím větší je objem produkce, tím nižší bude cena jednoho produktu na trhu. Příjmová funkce pro každou firmu je součin tržní ceny produktu a objem produkce firmy a lze ji popsat pomocí následující rovnice kde i značí firmu A nebo firmu B:

$$R_i(q_1, q_2) = pq_i$$

Pro zjištění změny příjmu při změně objemu produkce se využívá pravidlo o derivaci součinu funkcí a lze zapsat jako:

$$\frac{\partial R_i(q_1, q_2)}{\partial q_i} = p + q_i \frac{\partial p}{\partial q_i}$$

Funkce $p = g(q_1, q_2)$ je klesající a platí tedy:

$$\frac{\partial p}{\partial q_i} < 0 \, , p + q_i \frac{\partial p}{\partial q_i} < p$$

Nákladová funkce firem se označuje $C_i(q_i)$ a lze určit, jak se mění celkové náklady firmy při změně objemu produkce:

$$\frac{\partial C_i(q_i)}{\partial q_i}$$

Zisková funkce firmy A je pak závislá nejen na chování firmy A, ale i na chování firmy B, což platí také pro firmu B:

$$z_i(q_1, q_2) = pq_i - C_i(q_i) = g(q_1 + q_2)q_i - C_i(q_i) = R_i(q_1, q_2) - C_i(q_i)$$

Firma A i firma B se logicky snaží maximalizovat svůj zisk s předpokladem, že ví, jakou očekávanou strategii zvolí konkurenční firma. Pokud firma B zvolí předpokládanou strategii objemu výroby $q_2°$, pak firma A maximalizuje svůj zisk, který bude závislý na objemu výroby q_1 a objemu konkurenční firmy $q_2°$:

$$z_1(q_1, q_2°) = R_1(q_1, q_2°) - C_1(q_1)$$

Aby firma A dosáhla maxima zisku, musí platit podmínka 1. řádu derivace pro existenci extrému, tedy výsledek ziskové funkce po derivaci musí být roven 0:

$$\frac{\partial z_1(q_1, q_2°)}{\partial q_1} = \frac{\partial R_1(q_1, q_2°)}{\partial q_1} - \frac{\partial C_1(q_1)}{\partial q_1} = 0$$

Firma B pak při maximalizaci zisku předpokládá, že firma A zvolí strategii objemu výroby $q_1°$:

$$z_2(q_1°, q_2) = R_2(q_1°, q_2) - C_2(q_2)$$

Pro maximalizaci zisku firmy B pak musí opět platit podmínka 1. řádu derivace pro existenci extrému a výsledek ziskové funkce po derivaci musí být roven 0:

$$\frac{\partial z_2(q_1°, q_2)}{\partial q_2} = \frac{\partial R_2(q_1°, q_2)}{\partial q_2} - \frac{\partial C_2(q_2)}{\partial q_2} = 0$$

Obě firmy tedy znají strategie konkurenta a rovnovážný stav je určen ze soustavy rovnic strategií $q_1 a\ q_2$, které splňují maximalizaci zisku:

$$\frac{\partial z_1(q_1, q_2°)}{\partial q_1} = \frac{\partial R_1(q_1, q_2°)}{\partial q_1} - \frac{\partial C_1(q_1)}{\partial q_1} = 0$$

$$\frac{\partial z_2(q_1°, q_2)}{\partial q_2} = \frac{\partial R_2(q_1°, q_2)}{\partial q_2} - \frac{\partial C_2(q_2)}{\partial q_2} = 0$$

Při výpočtu soustavy rovnic je možno určit reakce chování duopolistů, ze kterých lze určit objem produkce podle velikosti produkce konkurence:

$$q_1 = \varphi_1(q_2), \qquad q_2 = \varphi_2(q_1)$$

4 Metody řešení vybraných strategií v programu MATLAB

4.1 Úvod

Pro ověření strategií z teorie her lze vytvořit program simulující dané modely na konkrétních příkladech. Pro tvorbu těchto programů bude použito vývojové prostředí MATLAB. U každého programu bude nejdříve teoretický popis, jak program funguje a bude k němu vytvořen pseudokód, jehož struktura je inspirovaná [13]. Programy budou spustitelné v prostředí MATLAB a jejích výpis zdrojového kódu bude v práci zahrnut.

4.2 Seznámení s programem MATLAB

MATLAB je interaktivní systém, který se používá pro výpočty a vizualizaci dat ve vědecké i inženýrské sféře. Podle [6] je to program, který dokáže řešit numerickou analýzu, práci s maticemi, zpracování signálů a řešení dokáže převést do grafické podoby, která je přehledná a dobře pochopitelná pro uživatele. Velkou výhodou programu MATLAB je jeho snadná rozšiřitelnost. Systém je možné rozšířit o vlastní napsané funkce nebo celé aplikace. K dispozici je také mnoho nadstaveb, kterým se říká toolboxy. Tyto toolboxy jsou kolekce funkcí, které mohou pomoci k hledání řešení v různých třídách problémů. Toolboxy jsou dostupné například pro tyto okruhy problémů:

- Zpracování signálů
- Zpracování obrazu
- Teorie řízení
- Identifikace systémů
- Optimalizace
- Neuronové sítě
- Statistika
- Symbolická matematika

Firma The MathWorks, která je autorem programu MATLAB, se stále aktivně věnuje podpoře svého produktu. Čtvrtletně je vydáván časopis *Newsletter*, ve kterém jsou všechny důležité informace o novinkách a aktualitách jak o samotném programu MATLAB, tak o rozšířeních a nových produktech, které se s programem MATLAB dají využívat.

Pro potřeby výzkumu v této práci je MATLAB využit k ověření funkčnosti několika metod z teorie her, které jsou probrané v teoretické části. Pro vybrané metody je vytvořen program, který ověřuje funkčnost daných metod. Jako vstup pro každý algoritmus je zadávána jedna nebo více matic. Každá matice představuje prostor strategií hráče, který se, účastní dané maticové hry. Po zadání matic algoritmus zhodnotí, jaký bude výstup této maticové hry.

4.2.1 Základní koncepce programu MATLAB

Základem práce s programem MATLAB je zadávání matic jako vstupů. Matice řádu jedna jsou využívány jako skaláry a matice s jedním řádkem nebo sloupcem jsou využívány jako vektory. Popis všech základních koncepcí a možností práce v programu MATLAB by byl opravdu rozsáhlý a pro účely této práce zbytečný. Popsané jsou tedy jen koncepce, které budou využívané v této práci. Pro podrobnější informace o ovládání a možnostech programu MATLAB lze využít [6].

4.2.2 Zadávání maticí

Matice se v programu MATLAB mohou zadávat více způsoby:

- zadání jako seznam prvků,
- vygenerování pomocí příkazu nebo funkce,
- načtení ze souboru vytvořeného lokálním editorem,
- načtení z externího datového souboru nebo aplikace.

V této práci jsou matice zadávány výhradně jako seznam prvků. Při zadávání dat, jakožto seznamu prvků, je vždy nutné dodržet následující pravidla:

- všechny prvky matice v jednom řádku jsou oddělené mezerou nebo čárkou,
- všechny řádky matice jsou oddělené středníkem nebo znakem, který ukončuje řádek,
- celá matice je uzavřená pomocí hranatých závorek.

Příklad zadání matice může tedy vypadat například takto:

$$A = [1\ 2\ 3; 4\ 5\ 6; 7\ 8\ 9]$$

Matice je také uložena do paměti A a je možno ji dále využívat. K načtení matice do mezipaměti je možné využít příkaz *load* a pro zobrazení hodnot z matice příkaz *read*.

Pokud je program použitelný pro různé varianty zadání vstupů, může být využit příkaz *input.* Pomocí tohoto příkazu může uživatel, který program spustí, zadávat libovolné hodnoty vstupů. Definování matice A, kterou zadá uživatel pomocí příkazu input, pak může vypadat například takto:

A = input('Zadejte prosím matici A')

Pokud je použit příkaz input a není definováno, že zadaná matice musí mít určitý počet prvků, je velmi užitečné využívat příkaz *size*, díky kterému je možno zjistit počet prvků v matici. Počet prvků matice je důležité znát, například pokud jsou v programu použity cykly, ve kterých je potřeba prohledat všechny prvky dané matice.

4.2.3 Cyklus FOR a podmínky IF

V progamu MATLAB se vyskytuje možnost použít cyklus FOR, který je známý z mnoha jiných programovacích jazyků. Tento cyklus umožňuje vykonat předem daný počet příkazů pro dané prvky z matice. Pro zadání, jaké prvky matice se mají pomocí cyklu *for* prozkoumat, je užitečné využít dvou v sobě vnořených cyklů, kdy jeden kontroluje řádky a druhý sloupce matice. Pokud například chce uživatel zkontrolovat všechny prvky vyskytující se v matici, lze toho jednoduše dosáhnout tak, že si nejdříve definuje pomocí funkce *size* počet řádků a počet sloupců v matici:

$$size(A) = [rowSize, colSize]$$

Poté je v *rowSize* a *colSize* uložen počet řádků a sloupců v matici. Pro zkontrolování všech prvků matice je pak možno vytvořit dva v sobě vnořené cykly.

$$for\ iRow = 1{:}\,rowSize$$

$$for\ iCol = 1{:}\,colSize$$

K určení podmínek, které pomáhají nalézt správné hodnoty z matice je používaná funkce *if,* která definuje omezeníplatná pro zkoumané hodnoty z matice.

4.2.4 Další užitečné funkce

V následujících programech se vyskytuje ještě několik užitečných funkcí z prostředí MATLAB, které pomáhají řešit matematické výpočty.

- *Diff(x)* slouží k provedení derivace prvního stupně pro rovnici x
- *Numel(x)* spočítá kolika ciferní je zadané číslo
- *Solve(x) == y* lze použít k řešení rovnic s neznámou, pomocí více funkcí *solve* lze zadat i soustavy rovnic s více neznámými hodnotami
- *Vpa(x,y)* převádí hodnotu zlomku x na celé číslo, y označuje kolik desetinných míst bude vypsáno
- *Int2str(x)* převádí číslo x na textový výstup, lze využít v odpovědi, která je vypsaná do konzole
- *Ones(x),zeros(x)* vytvoří matici jedniček nebo nul
- *Linprog* slouží k řešení úloh lineárního programování
- Error(´text´) zobrazuje hlášení o chybě ve funkci a ukončí program

4.3 Program č. 1 – Hledání sedlového bodu, rozhodnutí o typu hry

Program pro jednomaticovou hru se zabývá hledáním sedlového bodu a určením, zda se jedná o hru v ryzích nebo smíšených strategiích. Uživatel je po spuštění programu vyzván, aby zadal matici hry. K zadání matice je využita funkce INPUT.

Po zadání matice je vypočítaná celková délka a šířka matice pomocí funkce SIZE. Následně se definují dvě pole pro ukládání indexů prvků z matice a provede se cyklus, který zkontroluje všechny prvky. Ze všech prvků v matici se hledají právě ty, které jsou nejmenší ve svém řádku a zároveň největší ve svém sloupci. Každý takový prvek, který splňuje podmínky, představuje sedlový bod. Poté je spočítán počet sedlových bodů v matici.

Pokud byl nalezen jeden nebo více sedlových bodů, značí to, že matice má řešení v ryzích strategiích a jako výstup programu se vypíší všechny nalezené sedlové prvky a jejich pozice (označení řádku a sloupce) v matici. Pokud nebyl nalezen žádný sedlový prvek, matice má řešení ve smíšených strategiích a následuje výpočet hodnoty hry a pravděpodobnosti výhry při volbě určitých strategií. Pro tento postup jsou definovány podmínky hledání smíšených strategií a pomocí funkce LINPROG je nalezena pravděpodobnost výhry pro všechny strategie. Dále je také vypočítaná hodnota hry

4.3.1 Pseudokód pro program č. 1

Následující pseudokód popisuje algoritmus programu č. 1 pro hledání sedlového prvku a rozhodnutí o typu hry.

Výsledek 1: Výpis hodnoty sedlového prvku a jeho pozice **Výsledek 2:** Výpis hodnot všech sedlových prvků a jejich pozicí **Výsledek 3:** Výpis hodnoty hry a pravděpodobnosti **Vstup**: matice A s libovolným počtem prvků **Výstup:** pozice a hodnota všech nalezených prvků, hodnota a pravděpodobnost hry.

input matice A
int rowSize a colSize pomocí Size(A) uloží do proměnných šířku a výšku matice
int row[] a col[] pro ukládání sedlových prvků.
for iRow = 1:rowSize **do**
for jCol = 1:colSize **do**
if A(iRow,iCol) je nejmenší hodnota v řádku A(iRow,iCol) je největší hodnota ve sloupci
do Ulož index této hodnoty do pole row[], col[]
end
if size row[] = 1 && size col[] = 1
print Výsledek 1
else if row[] > 1 && size col[] > 1
print Výsledek 2
else if row[] > 1 && size col[] > 1
int AA [ones(colSize,1)-A'], Aeq[0 ones(1,rowSize)]
b = zeros(colSize,1), beq = [1];lb=[-inf;zeros(rowSize,1)];f = -[1;zeros(rowSize,1)];
p = **linprog**(f,AA,b,Aeq,beq,lb); v = P(1,1);P(1,:)=[];
print Výsledek 3

4.3.2 Zdrojový kód pro program č. 1

Následující zdrojový kód je exportovaná verze prvního programu přímo z MATLABu do .doc verze:

```
A=input('Zadejte matici:')
[rowSize,colSize] = size(A);
row=[];
col=[];
for iRow = 1:rowSize
for jCol = 1:colSize
if A(iRow,jCol) == min(A(iRow,:)) && A(iRow,jCol)== max(A(:,jCol));
row(length(row)+1)=iRow;
col(length(col)+1)=jCol;
end
end
end
if (length(row)==1 && length(col)==1)
Reseni=['Sedlovy prvek je na pozici: (' int2str(row) ',' int2str(col) ') a jeho hodnota
je: ' num2str(A(row,col))]
else
multipleSaddles=[];
for iRow=1:length(row)
 multipleSaddles=[multipleSaddles '(' int2str(row(iRow)) ',' int2str(col(iRow)) ')'];
 Reseni=['Sedlove prvky jsou na pozicich: (' multipleSaddles ') a jejich hodnota je: '
num2str(A(row(1),col(1)))]
end
end
if (length(row)==0 && length(col)==0)
AA = [ones(colSize,1) -A'];
Aeq = [0 ones(1,rowSize)];
b = zeros(colSize,1);
beq = [1];
```

```
lb = [-inf;zeros(rowSize,1)];
f = -[1;zeros(rowSize,1)];
P = linprog(f,AA,b,Aeq,beq,lb);
v=P(1,1);P(1,:)=[];
Reseni=['Neexistuje žádný sedlový bod a hodnota hry je: ' num2str(v) ' a smíšené strategie jsou dány maticí:']
Pravdepodobnost=P
end
```

4.4 Program č. 2 – Dvoumaticová hra, rozhodnutí o spolupráci hráčů

Tento program pracuje se dvěma maticemi, kde matice A značí prostor strategií prvního hráče a matice B značí prostor strategií pro druhého hráče. Pomocí programu lze rozhodnout, zda se hráčům vyplatí vzájemně spolupracovat nebo je pro ně výhodnější se spolupráci vyhnout, jelikož jim spolupráce nepřinese užitek.

Po spuštění programu uživatel zadá obě matice pomocí funkce INPUT. Po zadání obou matic je nejdříve zkontrolováno, zda mají matice stejné dimenze. Pokud nejsou dimenze stejné, program se přeruší a oznámí uživateli chybu, jelikož v takové hře musí mít matice stejné dimenze. Pokud jsou matice správně zadané, program sečte obě matice a uloží je do matice C, která značí prostor strategií v případě spolupráce. Matice A, B a C mají stejné dimenze a stačí tedy vypočítat počet sloupců a řádků pouze u jedné z těchto matic. To je provedeno příkazem SIZE. Následně se definují dvě pole pro ukládání indexů prvků z matice a je proveden cyklus, který postupně kontroluje všechny prvky z matice A a matice B. U každého prvku zkontroluje, zda je daný prvek největší ve sloupci u matice A a největší v řádku u matice B. Všechny prvky, které jsou na stejném indexu v matici a zároveň splňují obě podmínky, jsou uloženy jako sedlový bod hry.

Pokud nebyl nalezen žádný sedlový bod pro obě matice na stejné pozici, hra nemá řešení v ryzích strategiích a musí se hledat řešení ve smíšených strategiích.

Program č. 1 se již zaměřuje na hledání řešení ve smíšených strategiích a pro účely hledání rozhodnutí o kooperaci hráčů je implementace smíšených strategií značně komplikovaná. Uživatel je tedy pouze informován o tom, že nebyl nalezen sedlový bod a že tato hra má řešení ve smíšených strategiích, které v tomto programu nemají řešení.

Pokud byl nalezen jeden nebo více sedlový bodů na stejných pozicích jak pro matici A tak pro matici B, značí tyto body možné řešení v ryzích strategiích. V situaci, kdy je nalezeno více takových bodů, jsou hodnoty porovnány a je vybrána ta varianta ve které, je hodnota zaručené výhry pro oba hráče nejvyšší. Následně se tyto zaručené výhry pro oba hráče sečtou a jsou porovnány s maximální hodnotou z matice C. V případě, že je hodnota z matice C vyšší než součet zaručených výher obou hráčů, značí to, že se spolupráce vyplatí, pokud je hodnota z matice C nižší než součet zaručených výher pro oba hráče, spolupráce se jim nevyplatí.

4.4.1 Pseudokód pro program č. 2

Výsledek 1: Řešení ve smíšených strategiích **Výsledek 2:** Spolupráce hráčů se vyplatí **Výsledek 3**: Spolupráce hráčů se nevyplatí **Vstup**: matice A a matice B **Výstup:** Rozhodnutí o výhodnosti spolupráce hráčů
input matice A **input** matici B **if** size(A) /== size(B) **error** **else count** A + B = matice **C**
Pozn.: *pokračování pseudokódu na další straně*

```
int rowSize a colSize pomocí size(C) uloží do proměnných šířku a výšku matice
for iRow = 1:rowSize do
for jCol = 1:colSize do
if  A(iRow,iCol) je největší hodnota ve sloupci && B(iRow,iCol) je největší hodnota v řádku
do Ulož index této hodnoty do pole row[], col[]
end
if nebyl nalezen žádný společný sedlový bod
print Výsledek 1
else if byl nalezen společný sedlový bod
count maximální hodnotu v matici C (maxC)
count maximální součet sedlových bodů v matici A a matici B (maxAB)
if  maxC > maxAB
print Vysledek 2
else
print Vysledek 3
end
```

4.4.2 Zdrojový kód pro program č. 2

Následující zdrojový kód je exportovaná verze prvního programu přímo z MATLABu do .doc verze:

```
A=input('Zadejte matici A:')
B=input('Zadejte matici B:')
if any(size(A) ~= size(B))
error('Matice musí mít stejné dimenze');
end
C=A+B
[rowSize,colSize] = size(A);
row=[];
col=[];
for iRow = 1:rowSize
```

```
for jCol = 1:colSize
if A(iRow,jCol) == max(A(:,jCol)) && B(iRow,jCol)== max(B(iRow,:));
row(length(row)+1)=iRow;
col(length(col)+1)=jCol;
end
end
end

maximumC=max(C(:))
maximumAB=max(max(A(row,col))+ max(B(row,col)))
 if (length(row)<1 | length(col) < 1)
  Reseni=['Nebyl nalezen žádný společný sedlový prvek a není možné určit zda se kooperace hráčů vyplatí, hra má řešení ve smíšených strategiích, které je probrané v příkladu č.1']
 end
if (length(row)>=1 | length(col)>= 1)
if maximumC>maximumAB
   Reseni=['Hodnota výhry při spolupráci je ' int2str(maximumC) ' A hodnota výhry bez spolupráce je ' int2str(maximumAB) ' spolupráce se tedy vyplatí']
end
if maximumC==maximumAB
    Reseni=['Hodnota výhry při spolupráci je ' int2str(maximumC) ' A hodnota výhry bez spolupráce je ' int2str(maximumAB) ' spolupráce tedy nemá žádný význam']
end
if maximumC<maximumAB
    Reseni=['Hodnota výhry při spolupráci je ' int2str(maximumC) ' A hodnota výhry bez spolupráce je ' int2str(maximumAB) ' spolupráce se tedy nevyplatí']
end
end
```

4.5 Program č.3 – Konflikt dvou firem, hry v rozvinutém tvaru

Tento program simuluje příklad her v rozvinutém tvaru jinak nazývaných také jako „tahové hry“. V příkladu vystupuje firma A, která již působí na trhu a má svoji pozici s určitým ročním ziskem. Firma B se rozhoduje, zda vstoupit na trh jako konkurent firmy A.

Firma A má možnost bojovat proti vstupu firmy B na trh, nebo může zůstat nečinná a nebránit se vstupu konkurence na trh. Tento konflikt se ukazuje ve formě hry v rozvinutém tvaru a každá firma volí svoji strategii na základě předchozího tahu konkurence. Optimální strategie pro každou firmu se tedy v průběhu hry mění. Tento konflikt a všechny možnosti strategií s hodnotou výhry lze znázornit v následující tabulce:

Příklad 3	Firma B – žádná akce	Firma B – vstup na trh
Firma A – boj	a (1); b (1)	a (3); b (3)
Firma A – žádná akce	a (2); b (2)	a (4); b (4)

Do programu se nejdříve zadá vstup s hodnotami výher pro firmu A a firmu B kde platí:

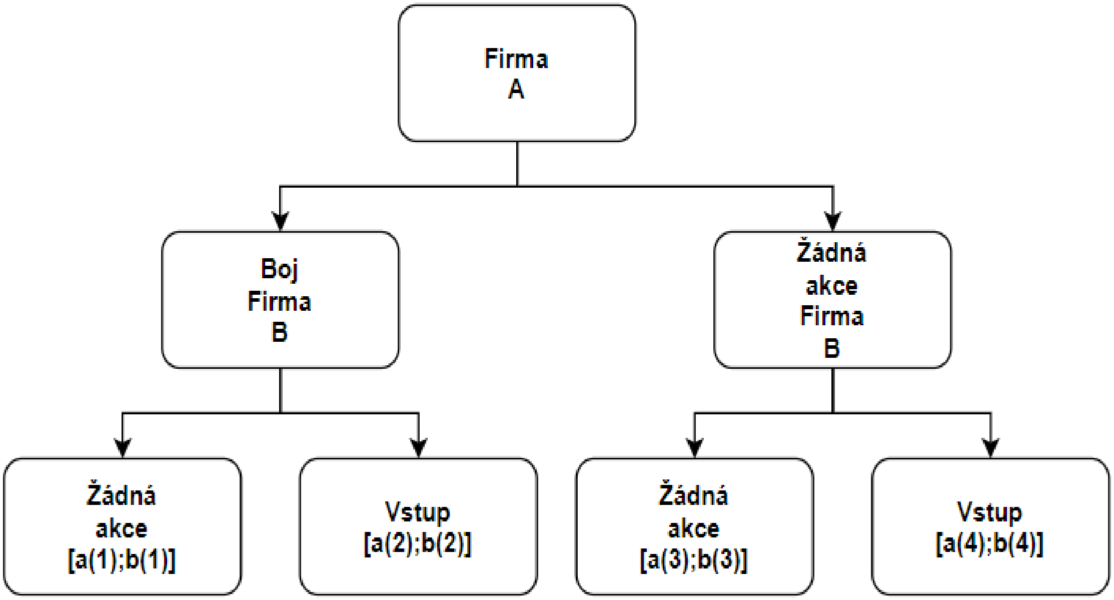
- a (1), b (1) – firma A se brání vstupu konkurence a firma B nevstoupí na trh
- a (2), b (2) – firma A se nebrání vstupu konkurence a firma B nevstoupí na trh
- a (3), b (3) – firma A se brání vstupu konkurence a firma B vstoupí na trh
- a (4), b (4) – firma A se nebrání vstupu konkurence a firma B vstoupí na trh

Vstup je zadán v následujícím formátu do programu MATLAB:

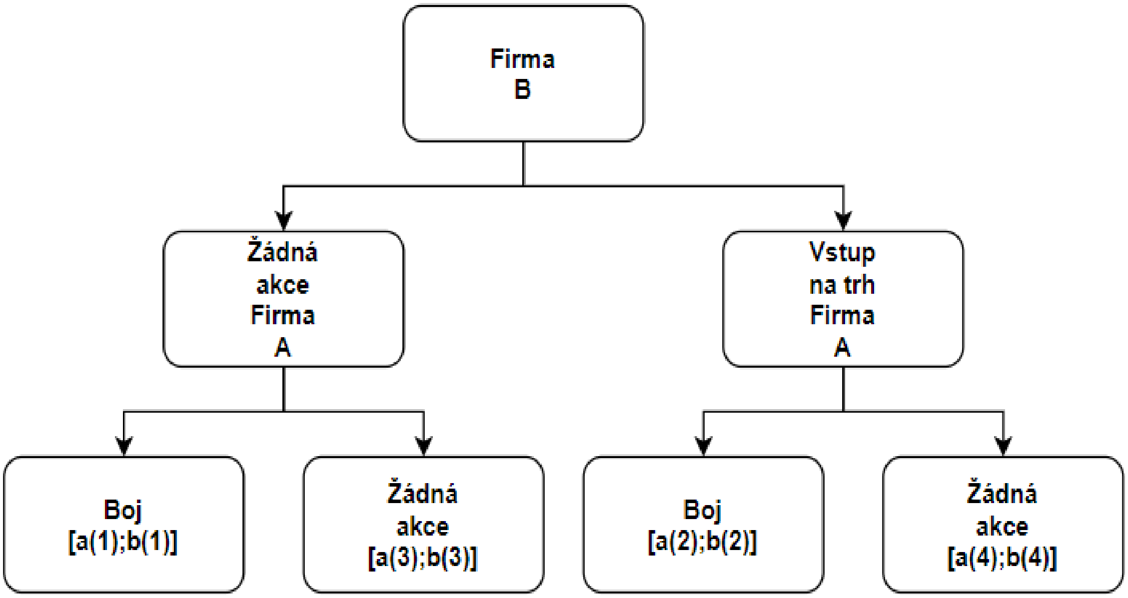
$$A = [a(1), a(3);\ a(2), a(4)]$$

$$B = [b(1), b(3); b(2), b(4)]$$

Pro lepší představu je možno obě situace ukázat na následujících obrázcích stromu hry:



Obrázek 3 Strom hry – firma A je první na tahu, Vlastní obrázek



Obrázek 4 Strom hry – firma B je první na tahu, Vlastní obrázek

Program je upraven tak, aby uživatel nezadával pouze kompletní matice strategií. Je to hlavně z toho důvodu, že v reálném konfliktu by neměl program takový užitek, jelikož zástupci firem obvykle neví, jaké jsou všechny hodnoty výher z možných strategií. Matice se tedy sama vytvoří za pomoci vzorců, které vypočítávají hodnoty v matici. Tyto vzorce jsou vytvořeny podle logického uvážení a není ověřeno, zda by fungovali v reálném prostředí. Při implementaci do reálného prostředí by bylo možné vzorce podle potřeby upravit, aby výstup z programu byl přesnější.

V programu se předpokládá, že budou zadány tyto informace:

- Aktuální zisk firmy A ($z1$)
- Odhadovaný zisk firmy B při vstupu na trh, pokud se firma A vstupu nebude bránit ($z2$)
- Náklady na reklamní kampaň pro firmu A, která chce oslabit konkurenční firmu B při jejím vstupu na trh (c)
- Odhadované procentuální oslabení zisku firmy B v případě, že firma A investuje do reklamní kampaně (l)

Matice jsou pak vytvořeny pomocí vzorců:

4.5.1 Firma A investuje a firma B je nečinná

a (1), b (1) – Firma A investuje do reklamní kampaně, ale firma B se kvůli tomu rozhodne na trh nevstoupit. Firma A tedy ztratí náklady na kampaň bez oslabení konkurence, ale část nákladů se jí vrátí, jelikož kampaň neproběhne v plné míře podle původního plánu. Pro firmu A platí vzorec:

$$a(1) = z_1 - \frac{c}{2}$$

Zisk firmy B bude v tomto případě 0, jelikož na trh vůbec nevstoupí:

$$b(2) = 0$$

4.5.2 Firma A neinvestuje a firma B je nečinná

$\boldsymbol{a}\ (\mathbf{2}), \boldsymbol{b}\ (\mathbf{2})$ – Firma A neinvestuje do reklamní kampaně a firma B nevstupuje na trh. Pro firmu A platí vzorec:

$$\boldsymbol{a}(\mathbf{2}) = \boldsymbol{z_1}$$

Pro firmu B platí stejný vzorec jako v první variantě:

$$b(2) = 0$$

4.5.3 Firma A investuje a firma B vstoupí na trh

$\boldsymbol{a}\ (\mathbf{3}), \boldsymbol{b}\ (\mathbf{3})$ – Firma A investuje do reklamní kampaně a firma B přesto vstoupí na trh. V tomto případě firma A získá svůj odhadovaný zisk a od něj odečítá náklady za kampaň a zisk firmy B, který jim z trhu konkurenční firma vezme. Pro firmu B je však jejich odhadovaný zisk částečně snížen na základě reklamní kampaně konkurence. Pro firmu A bude platit vzorec:

$$a(3) = \ z_1 - c - (z2 - \left(\frac{1}{100}\right) * z_2)$$

Pro firmu B pak platí vzorec:

$$b(3) \ = \ z2 - \left(\frac{1}{100}\right) * z_2$$

4.5.4 Firma A neinvestuje a firma B vstoupí na trh

$\boldsymbol{a}\ (\mathbf{4}), \boldsymbol{b}\ (\mathbf{4})$ – V posledním případě se firma A rozhodne neinvestovat do kampaně proti konkurenci a firma B vstoupí na trh. Firma A tedy získá svůj odhadovaný zisk se ztrátou zisku, který ji sebere firma B. Pro firmu A bude platit vzorec:

$$a(4) = \ z_1 - z_2$$

Pro firmu B pak platí vzorec:

$$b(4) = \ z_2$$

Při spuštění programu je uživatel vyzván, aby zadal pomocí funkce INPUT potřebné informace pro určení správné strategie. Nejdříve je zadán roční zisk firmy A a poté odhadovaný roční zisk firmy B při vstupu na trh. Dále se zadá, jak velká by byla investice do reklamní kampaně v případě boje s konkurenční firmou na trhu a také odhad, jak moc by tato kampaň mohla oslabit konkurenční firmu.

Po zadání všech hodnot jsou vytvořeny matice označující prostory strategií pro firmu A a firmu B. Následně je možnost zvolit, která z firem bude první na tahu, tedy udělá ve sporu první krok a také jakou strategii v tomto tahu zvolí. Po zadání těchto hodnot jsou porovnány hodnoty výhry a firma, která je druhá na tahu dostane informace o tom, jaká volba strategie je pro ně v této situaci výhodnější. Jsou vypsány hodnoty výher při obou možných strategiích a je porovnáno, která z nich je výhodnější a přinese větší zisk. Program je tímto optimalizován pro všechny varianty vstupů od obou firem a dokáže řešit případy kdy je na tahu firma A i případy kdy je první na tahu firma B.

V praktické situaci může uživatel zadat, jak se zachovala první firma na tahu a program mu zhodnotí, jak se má rozhodnout v druhém tahu, aby jeho firma dosáhla vyššího zisku.

4.5.5 Pseudokód pro program č. 3

Následující pseudokód popisuje algoritmus programu č. 3 pro konflikt dvou firem v rozvinutém tvaru.

Výsledek: Určení správného rozhodnutí pro hodnotu nejvyšší výhry na základě předchozího tahu soupeře

Vstup: zisk firmy A

Vstup: zisk firmy B

Vstup: Náklady na reklamní kampaň

Vstup: Míra oslabení konkurence

Vstup: Která firma bude první na tahu

Vstup: jakou strategii první firma na tahu zvolí

Výstup: porovnané hodnoty výher všech možných voleb strategie a označení ideální strategie

input z1

input z2

input c

Pozn.: *pokračování pseudokódu na další straně*

```
input l
int reklamniKampan = z1 – c – (z2 – ((1/100)*z2))
int matice A = (z1 – (c/2),reklamniKampan;z1,z1-z2)
int matice B = (0,z2-((1/100)*z2);0,z2)
input prvni
input strategie
if prvni == A &&  strategie = obrana
Porovnej hodnoty B(3) a B(1)
print Vysledek
else if prvni == A &&  strategie = bezAkce
Porovnej hodnoty B(4) a B(2)
print Vysledek
else if prvni == B && strategie = vstupNaTrh
Porovnej hodnoty A(3) a A(4)
print Vysledek
else if prvni == B && strategie = bezAkce
Porovnej hodnoty A(4) a A(2)
print Vysledek
```

4.5.6 Zdrojový kód pro program č .3

Následující zdrojový kód je exportovaná verze prvního programu přímo z MATLABU do .doc verze:

```
z1 = input('Zadejte aktuální roční příjem firmy A(Kč): ')
z2 = input('Zadejte odhadovaný roční příjem firmy B při vstupu na trh(Kč): ')
c = input('Zadejte náklady na kampaň oslabující konkurenční firmu(Kč): ')
l = input('Zadejte o kolik procent může kampaň oslabit zisk firmy B(%): ')
reklamniKampan = z1-c-(z2-((1/100)*z2))
A=[z1-(c/2),reklamniKampan;z1,z1-z2]
B=[0,z2-((l/100)*ziskB);0,z2]
if any(size(A) ~= size(B))
error('Matice musí mít stejné dimenze');
end
Prvni=input('Zvolte ktera firma bude první na tahu. 1 - Firma A 2 - Firma B ')
if (Prvni == 1)
boj=input('Bude firma A bránit firmě B ve vstupu na trh ? 1 - Ano 2 - Ne ')
if(boj == 1)
if (B(3)>B(1))
vysledek = ['Firmě B se vyplatí vstoupit na trh. Hodnota výhry při vstupu je: '
int2str(B(3)) ' Kč a bez akce je: ' int2str(B(1)) ' Kč']
else
vysledek = ['Firmě B se nevyplatí vstoupit na trh. Hodnota výhry při vstupu je: '
int2str(B(3)) ' Kč a bez akce je: ' int2str(B(1)) ' Kč']
 end
 else if (boj == 2)
 if (B(4)>B(2))
vysledek = ['Firmě B se vyplatí vstoupit na trh. Hodnota výhry při vstupu je: '
int2str(B(4)) '  Kč a bez akce je: ' int2str(B(2)) ' Kč']
else
vysledek = ['Firmě B se nevyplatí vstoupit na trh. Hodnota výhry při vstupu je: '
int2str(B(4)) ' Kč a bez akce je: ' int2str(B(2)) ' Kč']
```

```
end
end
end
else if (Prvni == 2)
vstupNaTrh = input('Rozdhodne se firma B vstoupit na trh ? 1 - Ano 2 - Ne')
 if(vstupNaTrh == 1)
 if (A(3)>A(4))
vysledek = ['Firmě A se vyplatí bojovat s firmou B. Hodnota výhry firmy A při boji je: ' int2str(A(3)) ' Kč a bez akce je: ' int2str(A(4)) ' Kč']
else
vysledek = ['Firmě A se nevyplatí bojovat s firmou B. Hodnota výhry firmy A při boji je: ' int2str(A(3)) ' Kč a bez akce je: ' int2str(A(4)) ' Kč']
 end
else if (vstupNaTrh == 2)
if (A(2)>A(1))
vysledek = ['Firmě A  se nevyplatí bojovat s firmou B. Hodnota výhry při boji je: ' int2str(A(1)) ' Kč a bez akce je: ' int2str(A(2)) ' Kč']
else
vysledek = ['Firmě A se vyplatí bojovat s firmou B. Hodnota vyhrý při boji je: ' int2str(A(1)) ' Kč a bez akce je: ' int2str(A(2)) ' Kč']
end
end
end
end
end
```

4.6 Program č. 4 –nalezení úrovně produkce, hodnoty zisku a rovnovážné ceny

Tento program se zakládá na Cournotovu modelu duopolu. Platí zde pravidlo, že tržní cena daného produktu se odvíjí od celkové produkce obou duoopolistů a je označena jako:

$$p = 10^x - (q_1 + q_2)$$

kde $q_1 a\ q_2$ značí produkci obou hráčů a x slouží k zvyšování ciferní ceny produktu úměrně k nákladům. Náklady na produkt jsou voleny při spuštění programu zadáním ceny za jeden produkt. Hodnota zisku je vypočítána pomocí rovnice:

$$z_i = p * q_i - C_i$$

Při spuštění programu uživatel zadá, jaké jsou náklady na jeden kus produktu pro firmu A a jaké jsou náklady na jeden kus produktu pro firmu B. Na základě těchto nákladů je definována rovnice p a definuje se rovnice zisku. Pro výpočet hodnoty produkce hráčů je rovnice zisku derivována pomocí příkazu *diff* a obě derivované funkce jsou vloženy do soustavy rovnic a následně vyřešeny pomocí příkazu *solve.* Z příkazu *solve* jsou nalezeny hodnoty produkce $q_1 a\ q_2$ a z nich je pak vypočítán celkový zisk při této produkci a zisk za jeden kus produktu. Dále se vypočítá tržní cena výrobku.

4.6.1 Pseudokód pro program č .4

Následující pseudokód popisuje algoritmus programu č. 4 pro konflikt dvou firem v rozvinutém tvaru.

Výsledek: Zisk pro firmu A a pro firmu B z jednoho kusu produktu **Výsledek 2:** Tržní hodnota produktu **Vstup**: Náklady za jeden kus produktu pro firmu A a firmu B **Výstup:** Zisk firmy A a firmy B včetně průměrné tržní ceny produktu
Syms q1 q2 pro definici dvou neznámých **input** náklady firmy A **(c1)** **input** náklady firmy B **(c2)**
Pozn.: *pokračování pseudokódu na další straně*

```
int a = 10^(numel(c1)
int b = 10^(numel(c2)
Int trzni cena vyrobku (p)
Int zisk firmy A (z1)
Int zisk firmy B (z2)
Proveď derivaci zisku
firmaA = Diff(z1)
firma B = Diff(z2)
Soustava rovnic pro nalezení velikosti produkce q1,q2 pro A a B
eqns = [solve(firmaA,q1),solve(firmaB,q2)]
solve(eqns,q1,q2)
solve(p,trzniCena)
solve(z1,ziskaA)
solve(z2,ziskB)
print Vysledek 1
```

4.6.2 *Zdrojový kód pro program č. 4*

Následující zdrojový kód je exportovaná verze prvního programu přímo z MATLABu do .doc verze:

```
syms q1 q2
c1 = input('Zadejte náklady firmy A na jeden kus produktu (Kč):')
c2 = input('Zadejte náklady firmy B na jeden kus produktu (Kč):')
a = 10^(numel(num2str(c1)))
b = 10^(numel(num2str(c2)))
if a>=b
p = 10^(numel(num2str(c1)))-(q1+q2)
else
p = 10^(numel(num2str(c2)))-(q1+q2)
end
z1 = p*q1-c1*q1
z2 = p*q2-c2*q2
```

```
firmaA = diff(z1,q1)
firmaB = diff(z2,q2)
eqns = [solve(firmaA,q1) == q1, solve(firmaB,q2) == q2];
soustavaRovnic = solve(eqns,[q1,q2])
produkceA = vpa(soustavaRovnic.q1,4)
produkceB = vpa(soustavaRovnic.q2,4)
if a>=b
trzniCena = vpa(10^(numel(num2str(c1)))-(produkceA+produkceB),4)
else
trzniCena = vpa(10^(numel(num2str(c2)))-(produkceA+produkceB),4)
end
ziskA = vpa(trzniCena*produkceA-(c1*produkceA),6)
ziskB = vpa(trzniCena*produkceB-(c2*produkceB),6)
ziskZaJedenKusA = vpa(ziskA/produkceA,4)
ziskZaJedenKusB = vpa(ziskB/produkceB,4)
TrzniCena = ['Rovnovazna cena produktu je:' int2str(trzniCena) ' Kč']
ZiskFirem = ['Zisk firmy A z jednoho kusu produktu bude:'
int2str(ziskZaJedenKusA) ' Kč' ' a zisk firmy B z jednoho kusu produktu bude:'
int2str(ziskZaJedenKusB) ' Kč']
```

5 Aplikace teorie her v managementu

5.1 Popis zpracovaných programů

V předchozí kapitole jsou zpracovány čtyři programy, které rozebírají různé modely strategií řešící konfliktní situace z teorie her. Jak již bylo zmíněno, mnoho strategií z teorie her lze využívat v oblasti obchodu a ekonomie pro pomoc při rozhodování nebo dosažení lepších zisků a efektivity firem.

5.1.1 Hledání sedlového bodu v jednomaticové hře a určení typu hry

Pro program č. 1 jsou uvedeny tři praktické příklady, které prezentují konkrétní výsledky při hledání sedlových bodů. Pokud je nalezen jeden nebo více sedlových bodů, program vypíše jejich hodnoty a pozice v matici. Pokud není nalezen žádný sedlový bod, program vypíše pravděpodobnosti hry a hodnotu hry. Tento program spíše ověřuje základní metodu hledání Nashovy rovnováhy a nemá příliš efektivní využití v konfliktech z reálného světa.

5.1.2 Rozhodnutí o výhodnosti spolupráce dvou konkurenčních firem na trhu

Program č. 2 je předveden na příkladu, ve kterém se rozhoduje, zda se dvěma hráčům vyplatí vzájemná spolupráce ve dvoumaticové hře s nekonstantním součtem. Prostory strategií obou hráčů jsou zadané pomocí matice a výsledek programu určuje zaručenou hodnotu výhry pro každého hráče v případě, kdy spolupracovat nebudou a hodnotu společné výhry v případě, kdy se rozhodnou spolupracovat. Tento program již lze využít v konkrétních příkladech, kdy dvě konkurenční firmy na trhu můžou v určitých odvětvích obchodu spolupracovat za účelem dosažení vyššího zisku.

5.1.3 Rozhodnutí firmy zda vstoupit na konkurenční trh

Program č. 3 je předveden na dvou příkladech, ve kterých vystupuje firma A, působící na trhu s určitým zbožím a firma B, která má zájem o to na daný trh vstoupit. V příkladech jsou zadané hodnoty zisku firmy působící na trhu a také odhadované hodnoty zisku firmy, která chce vstoupit na trh. Dále je potřeba znát náklady pro firmu A, které by investovala do reklamní kampaně a odhadovanou ztrátu, kterou by tato reklamní kampaň konkurenční firmě způsobila. Při zadání všech těchto hodnot jsou vytvořené matice pro obě firmy, které určují jejich prostory strategií. Tyto příklady lze použít k nalezení doporučení jak se chovat jako firma, která reaguje na tah konkurence a má dokonalý přehled o informacích z tahu jejich konkurence.

5.1.4 Výpočet zisku z produktu při určitých nákladech a objemu produkce

Program č. 4 se snaží o aplikaci Cournotova modelu duopolu na praktické příklady. Jsou vytvořeny dva příklady, ve kterých vystupují firmy A a B. Obě firmy se věnují výrobě podobných produktů a rozdělují, si mezi sebou zákazníky v tomto odvětí. Podle Cournotova modelu duopolu je rovnovážná cena na trhu pro daný produkt odvozena od produkce celkové produkce obou duopolistů. Firmy mohou pomocí tohoto programu zjistit, jaký je jejich zisk z produkce tohoto zboží.

5.2 Hledání sedlového bodu v jednomaticové hře a určení typu hry

Pro ověření prvního příkladu lze uvést všechny situace, které mohou v takovém konfliktu nastat. Možnosti, jak tento příklad může skončit, jsou následující:

- bude nalezen jeden sedlový prvek, který určuje rovnovážnou strategii,
- bude nalezeno více sedlových prvků, které jsou všechny možností rovnovážné strategie,
- nebude nalezen žádný sedlový prvek a příklad má řešení ve smíšených strategiích.

Pro každou možnost lze uvést konkrétní příklad matice, která povede k jednomu z možných řešení:

Pro jeden sedlový prvek: lze zadat matici A:

$$A = \begin{bmatrix} 6 & 6 & 2 \\ -7 & -6 & -6 \\ 7 & 7 & -6 \end{bmatrix}$$

Pro více sedlových prvků lze zadat matici B:

$$B = \begin{bmatrix} 0 & 2 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 0 \end{bmatrix}$$

Pro žádný sedlový prvek lze zadat matici C:

$$C = \begin{bmatrix} 0 & -1 & -2 \\ -2 & 0 & 1 \end{bmatrix}$$

5.2.1 Výpis řešení z programu č. 1 pro matici A

```
>> Program1
Zadejte matici:[6,6,2;-7,-6,-6;7,7,-6]
A =
    6    6    2
   -7   -6   -6
    7    7   -6
Reseni = Sedlový prvek je na pozici: (1,3) a jeho hodnota je: 2
```

5.2.2 Analýza řešení z programu č. 1 pro matici A

Při zadání matice A je nalezen jeden sedlový prvek na pozici v matici (1,3) kde první hodnota značí sloupec matice a druhá hodnota značí řádek matice. Hodnota sedlového bodu je 2 a značí Nashovu rovnováhu v ryzích strategiích.

5.2.3 Výpis řešení z programu č. 1 pro matici B

```
>> Program1
Zadejte matici:[0,2,0;0,1,0;0,3,0]
A =
    0    2    0
    0    1    0
    0    3    0
Reseni = Sedlové prvky jsou na pozicích: ((1,1)(1,3)(2,1)(2,3)(3,1)(3,3)) a jejich hodnota je: 0
```

5.2.4 Analýza řešení z programu č. 1 pro matici B

Při zadání matice B je nalezeno více sedlových prvků v matici na pozicích ((1,1)(1,3)(2,1)(2,3)(3,1)(3,3)), u kterých první hodnota značí sloupec matice a druhá hodnota značí řádek matice. Hodnota sedlového bodu je 0 a značí Nashovu rovnováhu v ryzích strategiích.

5.2.5 Výpis řešení z programu č. 1 pro matici C

```
>> Program1
Zadejte matici:[0,-1,-2;-2,0,1]
A =
    0   -1   -2
   -2    0    1
Reseni = Neexistuje žádný sedlový bod a hodnota hry je: -0.8 a smíšené strategie jsou dány maticí:
Pravdepodobnost =
   0.6000
   0.4000
```

5.2.6 Analýza řešení z programu č. 1 pro matici C

Při zadání matice C není nalezen žádný sedlový bod a hra má tedy řešení ve smíšených strategiích. Hodnota hry je stanovena na -0.8 a pravděpodobnost výhry je 60% při volbě strategie a 40% při volbě druhé strategie.

5.3 Rozhodnutí o výhodnosti spolupráce dvou konkurenčních firem na trhu

Na trhu působí dvě firmy, které se rozhodují, zda se jim vyplatí v jejich sporu spolupracovat nebo zda je pro ně lepší, když se spolupráci vyhnou. Spolupráce se firmám vyplatí, pouze pokud díky tomu získají obě firmy větší zaručený zisk, než když se spolupráci vyhnou. Prostory strategií pro obě firmy jsou označené maticemi. Matice pro firmy A a B jsou následující:

$$A = \begin{bmatrix} 3 & -3 \\ 4 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 5 & -1 \\ 1 & 4 \end{bmatrix}$$

5.3.1 Výpis řešení z programu č. 2

```
>> Program2
Zadejte matici A:[3,-3;4,1]
A =
    3   -3
    4    1
Zadejte matici B:[5,-1;1,4]
B =
    5   -1
    1    4
C =
    8   -4
    5    5
maximumC = 8
maximumAB = 5
Reseni =Hodnota výhry při spolupráci je 8 A hodnota výhry bez spolupráce je 5 spolupráce se tedy vyplatí
```

5.3.2 Analýza řešení z programu č. 2

Při zadání matice A a matice B jako vstup do programu č. 2 vyjde výsledek, že zaručená výhra bez spolupráce je pro firmu A hodnota 1 a pro firmu B hodnota 4. Celková možná výhra je tedy bez spolupráce rovna 5. V případě spolupráce je celková možná výhra rovna 8 a firmám se tedy vyplatí v tomto konfliktu spolupracovat. Přebytečnou výhru nad jejich zaručené výhry bez spolupráce si mohou rozdělit podle vlastního uvážení.

5.4 Rozhodnutí firmy zda vstoupit na konkurenční trh

5.4.1 Příklad 1 - Dřevěné hlavolamy

Firma A se zabývá výrobou originálních dřevěných hlavolamů a v její zemi působení nemá žádnou konkurenční firmu, která by se zabývala výrobou podobných produktů. Roční tržby firmy A činní 8 milionů Kč.

Na trh vstoupí firma B, která odhaduje svoje tržby za první rok na 5 milionů Kč. Firma A se rozhoduje, zda investovat do reklamní kampaně a do vývoje nových hlavolamů, aby oslabila firmu B vstupující na trh nebo zda nechat firmu B vstoupit na trh a riskovat ztrátu části tržby. Reklamní kampaň a vývoj hlavolamů by firmu A vyšel na 2 miliony Kč, ale firmu B by oslabil o 25% jejich odhadovaných tržeb. Pokud nebude firma A reagovat na vstup konkurenční firmy na trh, přijde o část své tržby, kterou získá firma B. Vyplatí se firmě A investovat do oslabení konkurence nebo je pro firmu A výhodnější nechat vstoupit konkurenci na trh?

5.4.2 Výpis řešení z programu č. 3 – Dřevěné hlavolamy

```
>> Program3
Zadejte aktuální roční příjem firmy A(Kč): 8000000
z1 = 8000000
Zadejte odhadovaný roční příjem firmy B při vstupu na trh(Kč): 5000000
z2 = 5000000
Zadejte náklady na kampaň oslabující konkurenční firmu(Kč): 2000000
c = 2000000
Zadejte o kolik procent může kampaň oslabit zisk firmy B(%): 25
l = 25
reklamniKampan =  2250000
A =
         7000000    2250000
         8000000    3000000
B =
               0    3750000
               0    5000000
Zvolte ktera firma bude první na tahu. 1 - Firma A 2 - Firma B 2
Prvni = 2
Rozdhodne se firma B vstoupit na trh ? 1 - Ano 2 - Ne1
vstupNaTrh = 1
vysledek = Firmě A se nevyplatí bojovat s firmou B. Hodnota výhry firmy A při boji je: 2250000 Kč a bez akce je: 3000000 Kč
```

5.4.3 Analýza řešení z programu č .3 – Dřevěné hlavolamy

Při zadání všech hodnot zmíněných v úloze o vstupu konkurenční firmy na trh, lze dojít k řešení, že pro firmu A není výhodné investovat do reklamní kampaně, jelikož dosáhnou větší tržby, pokud firmu B nechají vstoupit na trh. Výdělek firmy A, pokud investuje do reklamní kampaně je 2 250 000 Kč a pokud do kampaně neinvestuje, dosáhne výdělku 3 000 000 Kč.

5.4.4 Příklad 2 – Mobilní operátoři

V zemi působí na trhu mobilních operátorů pouze jedna firma A, která poskytuje mobilní připojení všem klientům. Její roční tržba je odhadována na 2 miliardy Kč. O vstupu na trh uvažuje firma B, která má prostředky k tomu konkurovat firmě A v této oblasti. Podle odhadů by mohlo od firmy A odejít ke konkurenční firmě B 25% současných zákazníků a firma A by tak mohla přijít o 500 000 Kč. Firma A přemýšlí o investici do kampaně, která jim pomůže udržet si část jejich zákazníků a oslabit firmu B. Podle odhadů by kampaň vyšla firmu A na 250 000 Kč a oslabila by firmu B o 75% původní odhadované tržby. Vyplatí se firmě A investovat do kampaně?

5.4.5 Výpis řešení z programu č. 3 – Dřevěné hlavolamy

```
>> Program3
Zadejte aktuální roční příjem firmy A(Kč): 2000000000
z1 = 2.0000e+09
Zadejte odhadovaný roční příjem firmy B při vstupu na trh(Kč): 500000000
z2 = 500000000
Zadejte náklady na kampaň oslabující konkurenční firmu(Kč): 250000000
c = 250000000
Zadejte o kolik procent může kampaň oslabit zisk firmy B(%): 75
l = 75
reklamniKampan = 1.6250e+09
```

Pozn.: *pokračování výpisu na další straně*

```
A =
1.0e+09 *
1.8750    1.6250
 2.0000    1.5000
B =
0   125000000
0   500000000
Zvolte ktera firma bude první na tahu. 1 - Firma A 2 - Firma B 2
Prvni = 2
Rozdhodne se firma B vstoupit na trh ? 1 - Ano 2 - Ne1
vstupNaTrh = 1
vysledek =Firmě A se vyplatí bojovat s firmou B. Hodnota výhry firmy A při boji je: 1625000000 Kč a bez akce je: 1500000000 Kč
```

5.4.6 Analýza řešení z programu č. 3 – Mobilní operátoři

Firmě A se v tomto případě vyplatí investovat do kampaně, jelikož jim to zajistí celkové tržby 1,625 miliardy Kč. Pokud by nechali firmu B vstoupit na trh bez boje, jejich tržba by byla 1,5 miliardy Kč.

5.5 Výpočet zisku z produktu při určitých nákladech a objemu produkce

5.5.1 Příklad 1 – Kancelářské potřeby

Na trhu působí dvě firmy, které vyrábí kancelářské potřeby. Nově vznikla poptávka po poznámkovém bloku s rozměry, které zatím nejsou zavedené v systému. Obě firmy se rozhodnou tento druh sešitu zavést do výroby. Tržní cena tohoto výrobku se odvíjí od celkového množství jeho produkce a každá firma má na výrobu produktu jiné náklady, které se odvíjí od zázemí a možností výroby. Pro firmu A jsou náklady na jeden sešit 11 Kč a pro firmu B jsou náklady na jeden sešit 19 Kč. Jaká bude tržní cena produktu při produkci těchto dvou firem? Jaký bude zisk každé firmy z jednoho kusu sešitu?

5.5.2 Výpis řešení z programu č. 4 – Kancelářské potřeby

```
>> program4
Zadejte náklady firmy A na jeden kus produktu (Kč):11
c1 = 11
Zadejte náklady firmy B na jeden kus produktu (Kč):19
c2 = 19
a = 100
b = 100
p =  100 - q2 - q1
z1 = - 11*q1 - q1*(q1 + q2 - 100)
z2 = - 19*q2 - q2*(q1 + q2 - 100)
firmaA = 89 - q2 - 2*q1
firmaB =  81 - 2*q2 - q1
soustavaRovnic =
   q1: [1x1 sym]
   q2: [1x1 sym]
produkceA = 32.33
produkceB = 24.33
trzniCena = 43.33
ziskA = 1045.44
ziskB = 592.111
ziskZaJedenKusA = 32.33
ziskZaJedenKusB = 24.33
TrzniCena = Rovnovážná cena produktu je:43 Kč
ZiskFirem = Zisk firmy A z jednoho kusu produktu bude:32 Kč a zisk firmy B z jednoho kusu produktu bude:24 Kč
```

5.5.3 Analýza řešení z programu č. 4 – Kancelářské potřeby

Při zadání stanovených nákladů za jeden produkt z příkladu konfliktu, bude rovnovážná cena sešitu na trhu 43 Kč. Firma A bude mít zisk za jeden vyrobený sešit 32 Kč a firma B bude mít zisk za jeden vyrobený sešit 24 Kč.

5.5.4 Příklad 2 – Solární panely

Na trhu působí firma A a firma B, které obě vyrábí solární panely. V současnosti jsou náklady na výrobu jednoho solárního panelu velkého 10x10 metrů pro firmu A 11 300 Kč a pro firmu B 13 800 Kč. Jaká bude průměrná tržní hodnota takového panelu na trhu a jaký zisk z jednoho prodaného panelu získá firma A a firma B?

5.5.5 Výpis řešení z programu č. 4 – Solární panely

```
>> Program4
Zadejte náklady firmy A na jeden kus produktu (Kč):11300
c1 = 11300
Zadejte náklady firmy B na jeden kus produktu (Kč):13800
c2 = 13800
a = 100000
b = 100000
p = 100000 - q2 - q1
z1 = - 11300*q1 - q1*(q1 + q2 - 100000)
z2 = - 13800*q2 - q2*(q1 + q2 - 100000)
firmaA = 88700 - q2 - 2*q1
firmaB = 86200 - 2*q2 - q1
soustavaRovnic =
   q1: [1x1 sym]
   q2: [1x1 sym]
produkceA = 30400.0
produkceB = 27900.0
trzniCena = 41700.0
ziskA = 9.2416e8
ziskB = 7.7841e8
```

Pozn.: *pokračování výpisu na další straně*

```
ziskZaJedenKusA = 30400.0
ziskZaJedenKusB = 27900.0
TrzniCena = Rovnovazna cena produktu je:41700 Kč
ZiskFirem = Zisk firmy A z jednoho kusu produktu bude:30400 Kč a zisk firmy B z jednoho kusu produktu bude:27900 Kč
```

5.5.6 Analýza řešení z programu č. 4 – Solární panely

Při zadání nákladů pro firmu A a firmu B, lze dojít k řešení, že rovnovážná cena produktu je 41 700 Kč. Zisk pro firmu A z jednoho produktu je roven 30 400 Kč a zisk pro firmu B z jednoho produktu je roven 27 900 Kč.

6 Závěry a doporučení

V této práci bylo vytvořeno několik programů, které se snaží ověřit modely řešení konfliktních a rozhodovacích situací z oblasti teorie her. Všechny programy jsou vytvořeny na základě poznatků z teoretické části práce a byla kladena snaha o to, aby každý program co nejlépe ověřil daný model z teorie her na praktických příkladech.

K tvorbě programů bylo použito vývojové prostředí MATLAB. Základní funkce z prostředí MATLAB, využívané ve vytvořených programech, jsou čtenáři popsány proto, aby porozuměl procesům v programech. Jsou také vytvořeny popisy procesů probíhajících v programech a pseudokódy které čtenáři vysvětlují, jak probíhají výpočty a hledání řešení v programech. Každý program se zaměřuje na jiný model z teorie her a ověřuje jeho pravdivost. Jako výstup programu je vždy vypsána odpověď, která může být informativní a může poskytnout uživateli programu ideální přehled o možnostech volby strategie v daném konfliktu, nebo uživateli doporučí, která strategie je pro něj v současné situaci nejvýhodnější volbou. Programy jsou optimalizované tak, aby byly použitelné na libovolné zadání vstupů, které jsou v dané situaci validní.

U některých programů může při špatném zadání, které není očekávané a pravděpodobné v daných sporech nebo není v souladu s modelem, kterému se program věnuje, nastat problém s výpočty a uživateli vypsat nelogické výstupy. Tyto problémy by bylo při širším zkoumání problematiky potřeba vyřešit a více se zaměřit na omezující podmínky vstupů. Uživatel programu by v takových situacích měl být informován o tom, kde nastal problém a proč mu program neposkytl řešení.

Při dalším zkoumání by bylo možné zjistit, jaké konkrétní problémy se v oblasti financí a managementu firem na trhu řeší a pokusit se vytvořit aplikaci, která by umožňovala za využití programů vytvořených v této práci pomoci firmě v rozhodovacích a konfliktních situacích s nalezením nejlepšího řešení sporů.

7 Seznam použité literatury

[1] DLOUHÝ, Martin; FIALA, Petr. Úvod do teorie her. 2.vyd. Praha: Oeconomica, 2009. ISBN 978-80-245-1609-7.

[2] SAWA, Zdeněk. Teorie her, studijní opora. Ostrava: VŠB – Technická univerzita Ostrava, 2015.

[3] MAŇAS, Miroslav. Teorie her a konflikty zájmů. Praha: Vysoká škola ekonomická, 2002. ISBN 8024504502.

[4] NEUMANN, J., MORGENSTERN. Theory of Games and Economic Behavior. Princeton University Press, Princeton, 1944

[5] HYKŠOVÁ, Magdalena. Teorie her, učební text ČVUT[online]. 2019 [cit. 8.9.2019] Dostupné z: http://euler.fd.cvut.cz/predmety/teorie_her/hry.pdf

[6] HERINGOVÁ, Blanka; HORA, Petr. MATLAB – Práce s programem, H-S. 1995 [cit.3.7.2020] Dostupné z: http://www.cdm.cas.cz/czech/hora/vyuka/mvs/tutorial.pdf

[7] COURNOT, A. A.Recherches sur les Principes Mathématiques de la Théorie des Richesses, Paris, France: L. Hachette, 1838;

[8] ZERMELO, E.: Über eine Anwendung der Mengenlehre auf die Teorie des Schashspiels. Proceedings Fifth International Congress of Mathematicians, 1913, 2: 501-504.

[9] BOREL, E.: La Théorie du Jeu et les Équations Intégrales à Noyau Symétrique. Comptes Rendus de l´Académie des Sciences, 1921, 173 1304-1308.

[10] NEUMANN, J.: Zur Teorie der Gesellschaftsspiele. Mathematische Annalen, 1928,100:295-320.

[11] OWEN, Guillermo. Game theory. Fourth edition. Bingley: Emerald, 2013. ISBN 978-1-78190-507-4.

[12] CHVOJ, Martin. Pokročilá teorie her ve světě kolem nás. Praha: Grada, 2013. ISBN 978-80-247-4620-3.

[13] PRAŽÁK, Pavel. Diferenční rovnice s aplikacemi v ekonomii. Hradec Králové: GAUDEAMUS, Univerzita Hradec Králové, 2013. ISBN 978-80-7435-268-3

8 Přílohy

Součástí bakalářské práce je složka souborů, která obsahuje:

- Programy 1-4, které jsou používané v kapitolách 5 a 6 včetně vzorových zadání.

(Programy se spouští pomocí vývojového prostředí MATLAB)

UNIVERZITA HRADEC KRÁLOVÉ
Fakulta informatiky a managementu
Akademický rok: 2018/2019

Studijní program: Aplikovaná informatika
Forma studia: Prezenční
Obor/kombinace: Aplikovaná informatika (ai3-p)

Podklad pro zadání BAKALÁŘSKÉ práce studenta

Jméno a příjmení:	**Robert Míča**
Osobní číslo:	**I1700111**
Adresa:	Na Viničkách 646, Šestajovice, 25092 Šestajovice, Česká republika
Téma práce:	Teorie her v managementu
Téma práce anglicky:	Game theory in management
Vedoucí práce:	doc. RNDr. Pavel Pražák, Ph.D. Katedra informatiky a kvantitativních metod

Zásady pro vypracování:

Cíl práce: Práce se bude zabývat strategiemi z teorie her, napomáhajícími ke zvolení správného rozhodnutí v oblasti managementu. V programu MATLAB bude řešen konkrétní příklad a uvedena jeho analýza. Budou uvedeny příklady firemních sporů a návrh na jejich řešení.
Osnova práce:

1. Úvod
2. Teoretická část – základní poznatky z teorie her
3. Metody řešení vybraných strategií v programu MATLAB
4. Aplikace teorie her v managementu
5. Literární zdroje

Seznam doporučené literatury:

DLOUHÝ, Martin a Petr FIALA. *Úvod do teorie her*. 2., přeprac. vyd. Praha: Oeconomica, 2009. ISBN 978-80-245-1609-7..
CHVOJ, Martin. *Pokročilá teorie her ve světě kolem nás*. Praha: Grada, 2013. ISBN 978-80-247-4620-3.
OWEN, Guillermo. *Game theory*. Fourth edition. Bingley: Emerald, 2013. ISBN 978-1-78190-507-4.

Podpis studenta: Míča

Datum: 10.9.2019

Podpis vedoucího práce:

Datum:

© IS/STAG, Portál – Podklad kvalifikační práce , micaro1, 11. srpna 2020 11:22