

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Diplomová práce

Modelování důvěry v sociální síti

Plzeň, 2012

Vladimír Václavek

Prohlášení

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni 17. 5. 2012

.....

Vladimír Václavek

Poděkování

Tímto bych rád poděkoval paní Ing. Arnoštce Netrvalové, Ph.D za její vstřícný přístup, neuvěřitelnou trpělivost, poskytnutí studijních materiálů a neocenitelné rady a připomínky při realizaci této práce.

Dále bych rád poděkoval všem účastníkům mnou vytvořené skupiny na Facebooku za jejich dobrou vůli a pomoc k získání reálných dat pro provádění experimentů s modelem důvěry. Konkrétně se jedná o Šárku Fleiberkovou, Lukáše Havla, Lenku Havlovou, Oldřicha Honsnejmana, Irenu Lichnovskou, Marii Václavkovou, Marii Vojáčkovou, Sashu Pavlů, Tomáše Ptáčka, Miroslava Tomana, Lucii Vobejdovou a Renátu Žákovou.

Abstract

Trust modelling in a social network

The thesis deals with an approach to a concept of the trust and the trust modelling in a social network. The trust definition, essential terms, trust representation, social networks and social networks representation are introduced. Then the thesis describes the simulations tools based on multi-agent systems and the concept of the trust model in a social network. This model is programmed in Java with JADE framework and is applied to three examples with data from social network Facebook. The examples are included at the end of this thesis.

Obsah

1	Úvod	1
2	Důvěra	2
2.1	Definice důvěry	2
2.2	Důvěra v informačních technologiích	4
2.3	Reprezentace důvěry	5
2.3.1	Reprezentace důvěry mezi dvěma entitami	6
2.3.2	Reprezentace důvěry skupiny entit	8
2.4	Utváření důvěry	9
3	Sociální síť	11
3.1	Definice sociální sítě	11
3.1.1	Uzly sociální sítě	12
3.1.2	Relace sociální sítě	13
3.2	Vlastnosti sociální sítě	15
3.2.1	Vlastnosti uzlů sociální sítě	16
3.2.2	Vlastnosti relací sociální sítě	17
3.2.3	Vlastnosti celé sociální sítě	17
3.3	Reprezentace sociální sítě	18
3.4	Sociální sítě v informačních technologiích	21
3.4.1	Facebook	21
3.4.2	Twitter	22
3.4.3	Google Plus	22
3.4.4	LinkedIn	23
3.4.5	Lidé.cz	23
3.4.6	Spolužáci.cz	23
4	Model důvěry v sociální síti	24
4.1	Přístupy pro tvorbu modelů důvěry	24
4.2	Modelování důvěry	25
4.3	Technologie a nástroje pro tvorbu	25
4.3.1	Multiagentní systémy	26
4.3.1.1	JADE	27
4.3.1.2	RETSINA	28
4.3.1.3	SWARM	29

4.4	Personální model důvěry v sociální síti.....	30
5	Realizace softwarového systému	35
5.1	Analýza aplikace	35
5.2	Architektura	35
5.2.1	UML diagram tříd	36
5.2.2	Agent koordinátor	38
5.2.3	Agent aktér	38
5.2.4	Komunikace agenta koordinátora a agenta aktéra	38
5.3	Implementace.....	40
5.3.1	Třída Duvera	40
5.3.2	Třída Okno.....	42
5.3.3	Třída Lokalizace	44
5.3.4	Třída Nacitani	45
5.3.5	Třída Generovani	46
5.3.6	Třída Subjekt	48
5.3.7	Třída Akter	49
5.3.8	Třída Modelovani.....	51
5.3.9	Třída Vystup	52
5.3.10	Třída Util	54
6	Experimenty s modelem důvěry v sociální síti	55
6.1	Získ reálných dat	55
6.2	První část experimentu – čtyřčlenná skupina	56
6.2.1	Základní hodnoty	56
6.2.2	Výsledky první části experimentu.....	58
6.3	Druhá část experimentu – sedmičlenná skupina	63
6.3.1	Základní hodnoty	63
6.3.2	Výsledky druhé části experimentu	65
6.4	Třetí část experimentu – dvanáctičlenná skupina	71
6.4.1	Základní hodnoty	71
6.4.2	Výsledky třetí části.....	74
6.5	Zhodnocení experimentu	78
7	Závěr.....	79
	Přehled zkratk.....	80
	Literatura.....	81

Online zdroje.....	83
Přílohy	84

Rejstřík obrázků

Obrázek 1 - Reprezentace míry důvěry spojitou veličinou	6
Obrázek 2 - Reprezentace míry důvěry pomocí fuzzy logiky	6
Obrázek 3 - Čtverec důvěry mezi dvěma entitami	7
Obrázek 4 - Varianty čtverce důvěry	7
Obrázek 5 - Čtverec důvěry celé komunity	8
Obrázek 6 – Ukázka sociální sítě, převzato z www.istockphoto.com	12
Obrázek 7 - Malá a velká sociální síť zobrazená grafem, převzato z www.bordalierinstitute.com	18
Obrázek 8 - Matice incidence neorientované a orientované sítě	19
Obrázek 9 - Matice sousednosti neorientované a orientované sítě	20
Obrázek 10 - Listy sousednosti neorientované a orientované sítě	21
Obrázek 11 - Agent a jeho prostředí	26
Obrázek 12 - Hlavní prvky architektury JADE	28
Obrázek 13 - Hlavní prvky architektury RETSINA.....	29
Obrázek 14 - SWARM virtuální stroj	30
Obrázek 15 - Asymetrie míry důvěry.....	31
Obrázek 16 - Jednoduchá síť s pěti aktéry.....	32
Obrázek 17 - Síť tří prvků s reputacemi	33
Obrázek 18 - Zjednodušený UML diagram tříd.....	37
Obrázek 19 - Komunikace mezi agenty	39
Obrázek 20 - Struktura textového souboru ve sloupcích.....	45
Obrázek 21 - Sociální síť aktérů v první části experimentu	56
Obrázek 22 - Sociální síť aktérů ve druhé části experimentu.....	63
Obrázek 23 - Sociální síť aktérů ve třetí části experimentu	72
Obrázek 24 - Ukázka výstupního textové souboru	84
Obrázek 25 - Ukázka výstupního XML souboru	84
Obrázek 26 - Grafické rozhraní aplikace.....	90

Rejstřík tabulek

Tabulka 1 – Příklad uzlů sociální sítě	12
Tabulka 2 – Příklad binárních relací	13
Tabulka 3 – Příklad relací s výběrem z více možností.....	14
Tabulka 4 – Příklad relací skupinově zaměřených	14
Tabulka 5 – Příklad relací s úplným pořadím.....	15
Tabulka 6 – Příklad relací intervalových.....	15
Tabulka 7 - Koeficient vzdálenosti doporučení	32
Tabulka 8 - Struktura balíků aplikace	36
Tabulka 9 - Popis komunikace mezi agenty	39
Tabulka 10 - Metody a atributy třídy Duvera.java.....	41
Tabulka 11 - Metody a atributy třídy Okno.java	43
Tabulka 12 - Metody a atributy třídy Loklizace.java	44
Tabulka 13 - Metody a atributy třídy Nacitani.java	46
Tabulka 14 - Metody a atributy třídy Generovani.java	47
Tabulka 15 - Metody a atributy třídy Subjekt.java	48
Tabulka 16 - Metody a atributy třídy Akter.java	50
Tabulka 17 - Metody a atributy třídy Modelovani.java	52
Tabulka 18 - Metody a atributy třídy Vystup.java	53
Tabulka 19 - Metody a atributy třídy Util.java	54
Tabulka 20 - Prvky sítě a jejich pozice v matici v první části experimentu.....	56
Tabulka 21 - Váha doporučení, reputace a dispozice každého aktéra sítě v první části experimentu	57
Tabulka 22 - Dispozice k důvěře v jednotlivých iteracích v první části experimentu	57
Tabulka 23 - Prvky a jejich pozice v matici ve druhé části experimentu.....	63
Tabulka 24 - Váha doporučení, reputace a dispozice každého aktéra sítě ve druhé části experimentu	64
Tabulka 25 - Dispozice k důvěře v jednotlivých iteracích ve druhé části experimentu..	64
Tabulka 26 - Prvky a jejich pozice v matici ve třetí části experimentu	71
Tabulka 27 - Váha doporučení, reputace a dispozice každého aktéra sítě ve třetí části experimentu	73
Tabulka 28 - Dispozice k důvěře v jednotlivých iteracích ve třetí části experimentu	73
Tabulka 29 - Porovnání výsledků modelování s odpověďmi v dotaznících	78

Rejstřík grafů

Graf 1 - Vývoj dispozice k důvěře v první části experimentu	58
Graf 2 - Reputace udělená prvku C ostatními aktéry v první části experimentu	58
Graf 3 - Vývoj důvěry k prvku C u ostatních aktérů v první části experimentu.....	59
Graf 4 - Vývoj důvěry mezi prvky B a C v první části experimentu	60
Graf 5 - Reputace udělená prvku D ostatními aktéry v první části experimentu.....	60
Graf 6 - Vývoj důvěry k prvku D u ostatních aktérů v první části experimentu	61
Graf 7 - Vývoj důvěry všech aktérů v první části experimentu	62
Graf 8 - Vývoj dispozice k důvěře ve druhé části experimentu.....	65
Graf 9 - Reputace udělená prvku F ostatními aktéry v druhé části experimentu.....	66
Graf 10 - Vývoj důvěry k prvku F u ostatních aktérů v druhé části experimentu	66
Graf 11 - Reputace udělená prvku G ostatními aktéry v druhé části experimentu	67
Graf 12 - Vývoj důvěry k prvku F u ostatních aktérů v druhé části experimentu	67
Graf 13 - Vývoj důvěry k prvku A u aktérů B a G v druhé části experimentu.....	68
Graf 14 - Vývoj důvěry k prvku F u aktérů B a E v druhé části experimentu.....	69
Graf 15 - Vývoj důvěry všech aktérů v druhé části experimentu	70
Graf 16 - Vývoj dispozice k důvěře ve třetí části experimentu	74
Graf 17 - Reputace udělená prvku K ostatními aktéry ve třetí části experimentu	75
Graf 18 - Vývoj důvěry k prvku K u aktérů C, I, E a L ve třetí části experimentu	75
Graf 19 - Vývoj důvěry k prvku L u všech aktérů ve třetí části experimentu	76
Graf 20 - Vývoj Důvěry mezi prvky A a I ve třetí části experimentu.....	76
Graf 21 - Vývoj důvěry mezi prvky A, B, C a D od začátku experimentu	77

1 Úvod

Každý den se setkáváme s mnoha lidmi, které už známe, poznáváme anebo je možná nikdy znát nebudeme. Všechno záleží na tom, jak se nám člověk zalíbí, jak si s ním rozumíme či jak moc mu můžeme věřit. Právě v těchto okamžicích nabývá na významu pojem důvěra. Fenomén, který má velký význam jak v humanitních vědách, jako jsou například sociologie či psychologie, tak i vědách přírodních, poukažme na biologii, a vědách technických, kde máme na mysli hlavně elektrotechniku, strojírenství a v neposlední řadě i informatiku. Konkrétně v informatice vzpomeňme na multi-agentní systémy, sociální sítě, elektronické obchodování, bankovníctví, bezpečnost a samozřejmě mnoho dalších.

Svět je plný pokroku a neustále spěje dopředu. Velkou popularitu získaly v posledních letech hlavně sociální sítě, které zná asi každý a velká část populace je pak sama používá. Do těch opravdu nejznámějších určitě zařadíme Facebook, Twitter, MySpace či nedávno vzniklý Google+. Vždyť právě první jmenovaný měl v únoru tohoto roku 845 miliónů uživatelů aktivních alespoň jednou do měsíce podle [DBOOK]. A jak už bylo naznačeno v odstavci výše, právě důvěra v sociálních sítích patří k velmi důležitým aspektům. Každý den se tam potkají milióny lidí, kteří spolu chatují, řeší pracovní záležitosti, posílají si obrázky, fotky a videa, vyměňují různé dokumenty nebo hrají hry. Dochází mezi nimi k neustále interakci a ta ovlivňuje jejich důvěru a názory.

Tato práce se zabývá především modelováním a utvářením důvěry v sociálních sítích s využitím simulačních prostředků založených na agentních technologiích. V její první části, která je především teoretická, si definujeme důležité pojmy, jako je důvěra, vysvětlíme si, jak ji můžeme reprezentovat a jak se vyvíjí. Poté si něco povíme o sociálních sítích a utváření důvěry v takové síti. Nakonec se nezapomeneme podívat na simulační prostředky založené na agentních technologiích.

Druhá část práce je zaměřena praktičtěji. Ukážeme si konkrétní model důvěry v sociální síti a jeho implementaci v jazyce Java s využitím simulačního prostředku JADE. Právě v této aplikaci si nasimulujeme experiment s daty získanými během pokusu na Facebooku a ukážeme si výsledky navrženého modelu.

V poslední třetí části nalezneme přílohy, dodatky, uživatelský manuál k programu, návod jak aplikaci přeložit a spustit a další obrázky.

2 Důvěra

Důvěra je tak běžné slovo, že ho zná jistě každý a každý by ho vysvětlil jiným způsobem. Všichni se na důvěru dennodenně spoléháme a podle ní se rozhodujeme, ať už vědomě nebo ne, utváříme díky ní společnost. Malá důvěra může vést k vzájemným sporům, nezdravé rivalitě, negativnímu postoji a znemožnění společné spolupráce. Důvěřujeme svým rodičům a přátelům, kteří nás podporují, důvěřujeme motoristům na silnicích, že dodržují silniční pravidla, důvěřujeme slovům, která jsme přijali od jiného člověka, důvěřujeme dennímu tisku, že nám podáváme pravdivé informace, důvěřujeme politikům, že se postarají o naše dobro. A to jsme se do této chvíle bavili o důvěře jen z hlediska naší společnosti. Podívejme se třeba z okna do světa zvířat, samička si vybírá nejsilnějšího samečka, protože věří v jeho schopnost mít potomky a přežít, jedinci projevují důvěru stádu, ve kterém žijí, pro svou ochranu a pomoc s hledání potravy. V moderním světě si neustále zjednodušujeme život využíváním internetu, kde důvěřujeme internetovému bankovníctví, obchodům, ale i správnosti získaných dat z webových stránek, kvalitě služeb atd.

To ale stále není kompletní přehled možností, důvěru najdeme v sociologii, v ekonomii, v marketingu a mnoha dalších oborech a disciplínách. Potom tedy není divu, že existuje tolik teorií a definic, co vlastně je důvěra.

2.1 Definice důvěry

Obecně platnou definici důvěry bychom jen těžko hledali právě proto, že se důvěra vztahuje k tolika různým oblastem. Význam důvěry je natolik rozsáhlý, že i významové slovníky nabízí několik různých vysvětlení [FRDIC-TC]:

- Víra v poctivost, pravdivost, schopnost a charakter jiné osoby nebo věci.
- Péče, opatrovnictví. Působení ve prospěch někoho jiného.
- Naděje v budoucnost, naděje v dobrou věc.
- Víra v čest, právo, naději v jednání.
- Záměr a schopnost platit za budoucí služby.

Ukázku obtížnosti vymezení pojmu důvěry nám poskytuje práce [MARSH], kde S. P. Marsch dochází k závěru, že popsat důvěru v jedné definici je zhora nemožné. Další

přehled pokusů vymezení důvěry různých vědců můžeme najít v [PETPV]. Zde uvedeme jen opravdu ty nejzajímavější z nich, například dle R. Schottlaendera [TH-VKP]: „Důvěra plyne ze zkušenosti a naděje v dobro v lidech.“

Trochu jiným pohledem vidí důvěru Johnson a Matrossa, kteří tvrdí že se důvěra projevuje v připravenosti či snaze mluvit o věcech, které mohou vyvolat znehodnocení a odmítnutí, a znázorňují tak pro zúčastněné určitá rizika.

Jedna z definic, která si určitě zaslouží být zmíněna, je z díla [DEUT] a zní: „K důvěře v chování dochází, když jedinec vnímá neurčitý postup, jehož výsledky mohou být dobré nebo špatné a výskyt dobrého nebo špatného výsledku je závislý na jednání dalších osob; v konečné fázi pak špatný výsledek více ublíží, než dobrý výsledek prospěje. Když si jedinec zvolí takový postup, říká se o něm, že je důvěřivý, jestliže ne, pak je nedůvěřivý.“ Všimněme si faktu, že negativní prožitek ze zklamání od blízkého člověka, je daleko silnější než potěšující prožitek člověka neznámého (toho, kterému nedůvěřujeme).

Mezi nejdůležitější definice týkající se důvěry a našeho tématu, ale zajisté patří ta, kterou popsal na sympoziu o důvěře v Cambridge v Anglii roku 1988 Diego Gambetta: „Důvěra (analogicky pak nedůvěra) je konkrétní úroveň subjektivní pravděpodobnosti, se kterou agent odhadne vykonání určité akce jiného agenta či skupiny agentů ještě předtím, než takovou akci zpozoruje (nebo nezávisle na jeho schopnosti to kdykoliv zpozorovat) a v kontextu s tím provede svou vlastní akci. Agent je obecně jedinec či věc (entita), která působí na prostředí či další agenty, a má vlastnosti a své cíle, kterých se snaží dosáhnout.“

Všechny definice mají nakonec něco společného a sami můžeme určit několik základních znaků, které jsou pro důvěru typické:

- subjektivita;
- nejistota;
- specifická v různých situacích;
- není absolutní;
- existence rizika a nejistoty;
- nedá se předpovídat;
- závisí na čase;
- není tranzitivní.

Tyto znaky jsou blíže popsány v práci [ABRAH] a rozšířeny o další důležité aspekty.

2.2 Důvěra v informačních technologiích

V moderním světě se již využívání počítačů nevyhneme, stále více se objevují systémy s umělou inteligencí, společnosti s virtuální inteligencí, peer-to-peer sítě, ad hoc sítě, sociální sítě, internetové obchody, internetové bankovníctví, multiagentní systémy a podobně.

Význam důvěry v informačních technologiích se objevuje zejména při snaze nabídnout bezpečné, optimalizované a výkonné služby uživatelům internetu. Důvěra je pak odvozena od interakce mezi dvojicí uzlů a jejich zkušeností založené na míře dobrého a špatného hodnocení služby. Nalezneme několik specifických algoritmů jako je třeba PageRank pro hodnocení důležitosti webových stránek pomocí počtu odkazů a hodnocení odkazujících stránek či od něj odvozený EigenTrust algoritmus [EITRU] určující globální míru důvěry v peer-to-peer sítích, kdy každý uzel počítá svou vlastní reputaci hlasováním ostatních uzlů v závislosti na jejich reputaci. Právě díky výsledkům algoritmů jsme potom schopni určit kvalitní služby a zlepšovat jejich provozuschopnost do budoucnosti.

Představme si počítačovou síť složenou z velkého počtu uzlů, kde každý uzel určuje spojení s dalším uzlem a způsob přenosu informace mezi nimi. Každý uzel tedy na základě svých předchozích zkušeností spolupracuje s uzlem dalším za účelem nejlepšího výsledku. Pak logicky můžeme hovořit o důvěře mezi těmito uzly. Právě na takovémto principu fungují P2P sítě.

Dále se podívejme na internetové obchody. Prodejnost uvedeného zboží je založena na reputaci jednotlivých obchodů. My potom projevujeme důvěru takovému obchodu na základě vzájemné důvěry mezi obchodníkem a zákazníkem podle počtu spokojených zákazníků, zabezpečení a ochrany plateb, dodávce zboží atd. Elektronické obchody využívají dobré reputace k větší důvěryhodnosti. Algoritmy určující reputaci jsou založené na hlasování a získávání kladných a záporných bodů od různě postavených (každý uživatel má svou úroveň či reputaci, čím je potom jeho úroveň vyšší, tím větší váhu má jeho hodnocení) hodnotících uživatelů. Takové algoritmy můžeme vidět například na portálech Heuréka, Aukro či Ebay.

Další příklad bude z oblasti multiagentních systémů, kde důvěra znamená subjektivní očekávání chování agenta v budoucnosti. Úkolem je zvýšení bezpečnosti, spolehlivosti, optimalizaci a zdokonalování služeb takového systému.

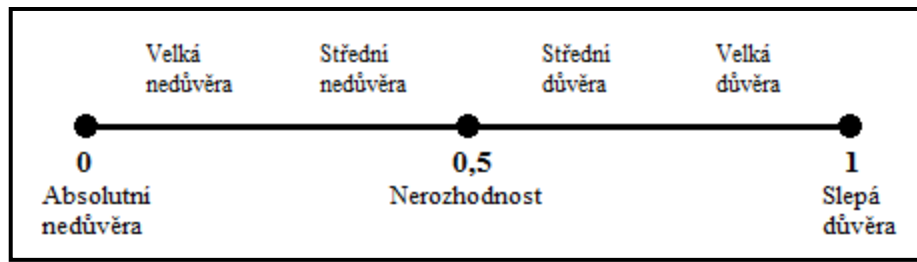
Ve velmi rychle se rozšiřujících sociálních sítích, které v poslední době velmi získaly na populárnosti, se musíme zaměřit na důvěru méně globálně, než tomu bylo v e-obchodování či P2P sítích. Uvědomme si potenciál, který nabízí od možnosti spojovat lidi z celého světa, přes sdílení informací a znalostí až po reklamu a nabídku služeb všeho druhu. Už tímto krátkým přehledem jsme si ukázali, jak sociální média obecně modelují moderní společnost a život v ní viz [SOCMO]. [GOLD] nabízí algoritmy pro počítání důvěry mezi dvěma osobami, které jsou sociální sítí propojeny. Základem výpočtu pak je dotaz na důvěru mezi osobami napříč celou zúčastněnou sociální sítí.

Celou situaci nám ztěžují různé pirátské útoky, které se snaží obejít důvěryhodnost a nabídnout tak falešnou představu a informace o poskytované službě. Zajímavou myšlenku řešení elektronické bezpečnosti a důvěry můžeme najít v [TRCO], kde se snažíme připodobnit obranu a bezpečnost systému vnímání bezpečnosti lidského mozku a chování v kritických situacích.

2.3 Reprezentace důvěry

Definice důvěry byla sice velmi obtížná, naopak znázornění již tolik není. První se podíváme, jak důvěru vyjádřit obecně a potom konkrétně mezi jednotlivci. Velikost (míra) důvěry se nejčastěji vyjadřuje jako spojitá veličina s pevně stanoveným intervalem velikosti. Jsou známy i výjimečné případy, kdy se používají veličiny diskrétní, ale pro méně přesné zobrazení se nedoporučuje.

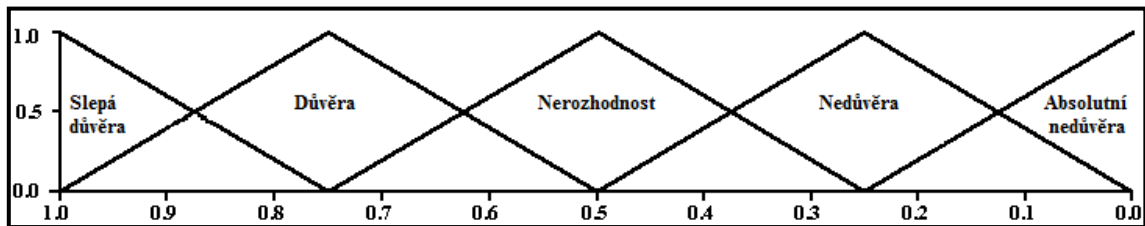
Konkrétní hodnoty míry důvěry se nejčastěji uvádí v intervalu od 0 do 1, kdy 0 značí naprostou nedůvěru a 1 slepou důvěru mezi dvěma nebo více entitami v komunitě. Střední hodnota intervalu je pak definována jako nerozhodný stav. Takováto reprezentace důvěry je modelována na základě pravděpodobnosti a lépe ji vidíme na obrázku 1. Obrázek nám pro detailnější popis udává další stupně důvěry.



Obrázek 1 - Reprezentace míry důvěry spojitou veličinou

Pro konkrétní případy může být toto zobrazení ještě více rozšířeno a to buď zvětšením intervalu spojitě veličiny, nebo navýšením počtu stupňů. Většinou pak dochází k oběma těmito modifikacím, princip však zůstává stále stejný.

Chceme-li dosáhnout subjektivnějšího ztvárnění důvěry na úrovni vnímání jedinců, je možné využít zobrazení pomocí fuzzy logiky [DEMOT]. Na obrázku 2 vidíme, že přechod mezi jednotlivými stupni není ostrý, tzn. existují zde oblasti překryvu, a to co může jedna entita považovat za nerozhodný stav už jiná bere jako důvěru a obráceně.



Obrázek 2 - Reprezentace míry důvěry pomocí fuzzy logiky

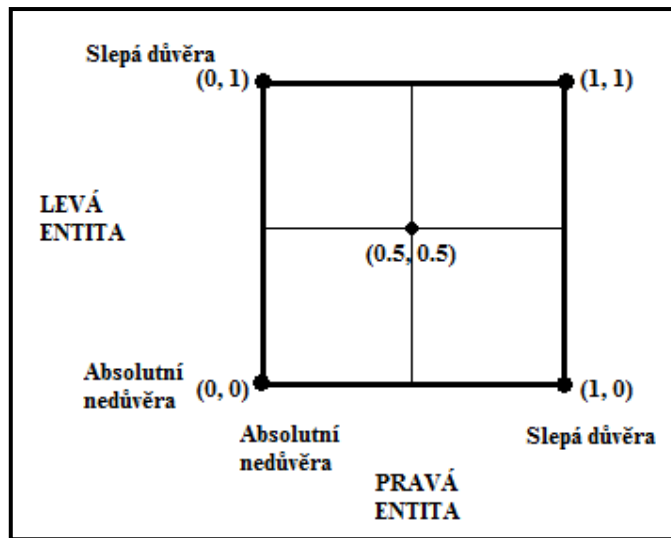
I

v tomto případě samozřejmě platí možnosti rozšíření zvětšením intervalu, přidání dalších stupňů nebo obojího jako tomu je v [MARSH]. Princip zobrazení velikosti důvěry zůstává i nadále stejný.

2.3.1 Reprezentace důvěry mezi dvěma entitami

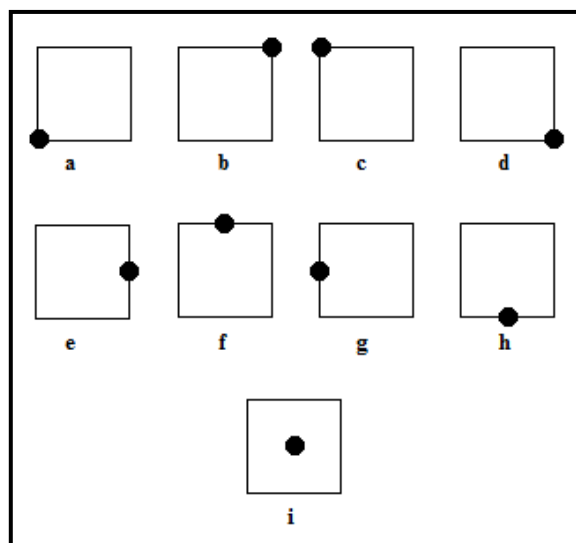
Jak se zobrazuje důvěra, jsme si již ukázali v minulé kapitole, nyní se zaměříme na to, jak ji zakreslit mezi dvěma entitami. Podle definice Diega Gambetty víme, že důvěra se váže vždy mezi dva objekty (poskytovatelem z anglického trustor a příjemcem z anglického trustee). Většinou máme celou skupinu entit, důvěru však chceme zpravidla určovat jen mezi jejími dvojicemi. Proto je také možné i ve skupině pohlížet na důvěru jako na vztah vybraných dyád. Vztah mezi takovou dvojicí prvků popsal

[URB] ve dvojrozměrném souřadném systému pomocí čtverců důvěry. Více vidíme na obrázku 3, kde se velikost důvěry nachází v intervalu $(0, 1)$. Hodnoty důvěry T pak označíme jako T_L a T_P , kde T_L vyjadřuje důvěru zleva doprava ve čtverci důvěry, tedy subjektu levého k subjektu pravému. To samozřejmě platí i obráceně.



Obrázek 3 - Čtverec důvěry mezi dvěma entitami

Zobrazením důvěry ve čtverci důvěry je potom průsečík přímek vynesných z obou entit – pravé i levé – v bodě určujícím velikost důvěry jedné entity k entitě druhé. Tento průsečík má tak devět různých variant popisujících rozdílnou situaci ve čtverci důvěry, viz obrázek 4.



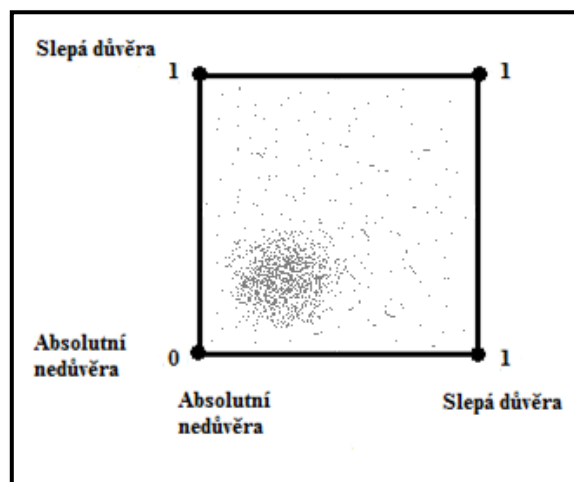
Obrázek 4 - Varianty čtverce důvěry

Varianty čtverce důvěry:

- a) Pár se vzájemnou absolutní nedůvěrou
- b) Pár se vzájemnou slepou důvěrou
- c) Pravá entita cítí k levé slepou důvěru a levá entita k pravé absolutní nedůvěru
- d) Levá entita cítí k pravé slepou důvěru a pravá entita k levé absolutní nedůvěru
- e) Levá entita je nerozhodná vůči entitě pravé a pravá slepě důvěřuje levé
- f) Pravá entita je nerozhodná vůči entitě levé a levá slepě důvěřuje pravé
- g) Levá entita je nerozhodná vůči entitě pravé a pravá naprosto nedůvěřuje levé
- h) Pravá entita je nerozhodná vůči entitě levé a levá naprosto nedůvěřuje pravé
- i) Oba z páru jsou nerozhodní vůči tomu druhému či mezi nimi neexistuje žádný vztah

2.3.2 Reprezentace důvěry skupiny entit

V minulé kapitole jsme si popsali jednoduchý způsob zobrazení ve čtvercích důvěry mezi dvěma prvky. Je to nejběžnější způsob zobrazování, ale pro úplnost uvedeme, jak zobrazit důvěru mezi všemi prvky ve skupině do dvourozměrného souřadného systému. Důvěru zaznamenáváme do jednoho čtverce důvěry pro všechny páry až do konečného počtu párů. Ve čtverci důvěry pak vznikají různé oblasti, kde oblast s nejkoncentrovanějším počtem průsečíků udává přibližnou důvěru v rámci této komunity. Jeden z příkladů je zaznamenán na obrázku 5. V tomto příkladu si můžeme povšimnout, že větší část komunity si mezi sebou moc nedůvěřuje.



Obrázek 5 - Čtverec důvěry celé komunity

2.4 Utváření důvěry

Utváření důvěry je velmi složitý proces, kdy záleží na každém jedinci, jak moc ho daná skutečnost ovlivní. Někteří lidé začnou důvěřovat až po velmi dlouhém čase, během kterého nesmí dojít k žádné nepříjemné události. Jiní zase po prvních několika hodinách vědí, že druhému člověku mohou důvěřovat prakticky v čemkoli. Opět se dostáváme k tomu, že vývoj důvěry je velmi subjektivní. Akce, která pro jednoho člověka nic neznamená, u druhého vyvolává pocity absolutní nedůvěry.

Úbytek i přírůstek důvěry je přímo závislý na okolnostech a zásadách ostatních a záleží na momentálním stavu a rozpoložení ovlivňovaného jedince. Můžeme vyjmenovat základní faktory ovlivňující utváření:

- zkušenost
- důvěryhodnost
- reputace
- doporučení
- kontext
- riziko
- evoluce

Zkušenost má při utváření velký vliv. Každý jedinec vnímá, jak mu subjekt ke kterému se důvěra utváří, v minulosti pomohl, co pro něj udělal, jaké bylo jeho chování a jak byly jeho činy nápomocné při řešení úkolu, problému. Do faktoru zkušenosti tedy řadíme předchozí kontakty a konfrontace.

Důvěryhodnost závisí na počtu vzájemných kontaktů. Čím více se se subjektem stýkáme, tím lépe si na něj můžeme vytvořit názor a tím spíše poznáme, jak moc se mu dá důvěřovat. Vidíme, že i důvěryhodnost je silným faktorem ovlivňujícím vývoj důvěry, protože si obrázek na subjekt utváříme my sami.

V pořadí třetím faktorem je reputace neboli pověst subjektu. Reputací myslíme informaci o oblíbenosti zkoumaného jedince v rámci skupiny, ve které se nachází. Když si uvedeme příklad z informačních technologií konkrétně z elektronického obchodování, tak hovoříme například o počtu hvězdiček u každého obchodu nebo jeho výrobku (čím více hvězdiček, tím lepší reputace), viz Heuréka [HEUR]. Pověst může být silně ovlivněna reklamou a masmédií, kdy může docházet k desinformaci klamnými zprávami.

Doporučení je faktor působící jen ve skupině několika zkoumaných prvků. V případě zkoumání důvěry dvou izolovaných prvků se faktor doporučení neobjevuje. Doporučením myslíme zprávu, kterou nám sdělí jiný jedinec (může jich být i více) na jedince právě zkoumaného. Takto získaná vědomost má pro každého jinou váhu, záleží například na předávajícím, na jeho reputaci či důvěryhodnosti, atd. Nutno podotknout, že jsou známy i příklady, kdy faktor doporučení není brán v potaz s ohledem na podstrčení falešných informací.

Kontextem rozumíme okolí, ve kterém se jedinec za daných podmínek nachází v určitém čase. Prakticky se tedy jedná o místo, čas, děj, náhodu atd. Kontext patří k velmi výrazným činitelům utváření důvěry, kdy v případě úspěchů můžeme leckdy hovořit o štěstí.

Utváření důvěry je vždy ovlivněno rizikem zisku nebo naopak ztráty důvěry. Určuje ochotu jedince riskovat v zájmu úspěchu. Často se tak shledáváme s procesy určujícími míru rizika za daných okolností, které tak ovlivňují konání jedince.

Evoluce nebo vývoj důvěry je výsledkem existence vypsanych faktorů, díky kterým se důvěra mezi entitami v čase neustále mění (vyvíjí). Tím jsme ale jen znovu opsali vlastnosti důvěry definované v kapitole 2.1.

Na základě těchto obecných faktorů je v kapitole 4 představen model pro vývoj důvěry v sociální síti.

3 Sociální síť

Sociální sítě jsou fenoménem dnešní doby a můžeme je najít v mnoha odvětvích a oborech. Asi nejznámější spojení se sociální sítí nalezneme v oblastech informačních technologií, kdy pojmy jako Facebook, Twitter a další nezná opravdu jen málokdo. Právě pro velký počet aktivních uživatelů sociálních sítí na nich najdeme vskutku ledacos od reklamy výrobků až po ohromné kampaně různých institucí. A to stále nemluvíme o nepřeborném množství informací a znalostí, které jsou v nich uloženy. Možná právě proto mají sociální sítě takový vliv na vývoj společnosti.

Pojem sociálních sítí však nemusíme nutně vztahovat pouze k informačním technologiím, nacházíme je totiž ve vědách jako je biologie, sociologie, psychologie, ekonomie, marketing, geografie nebo elektrotechnika.

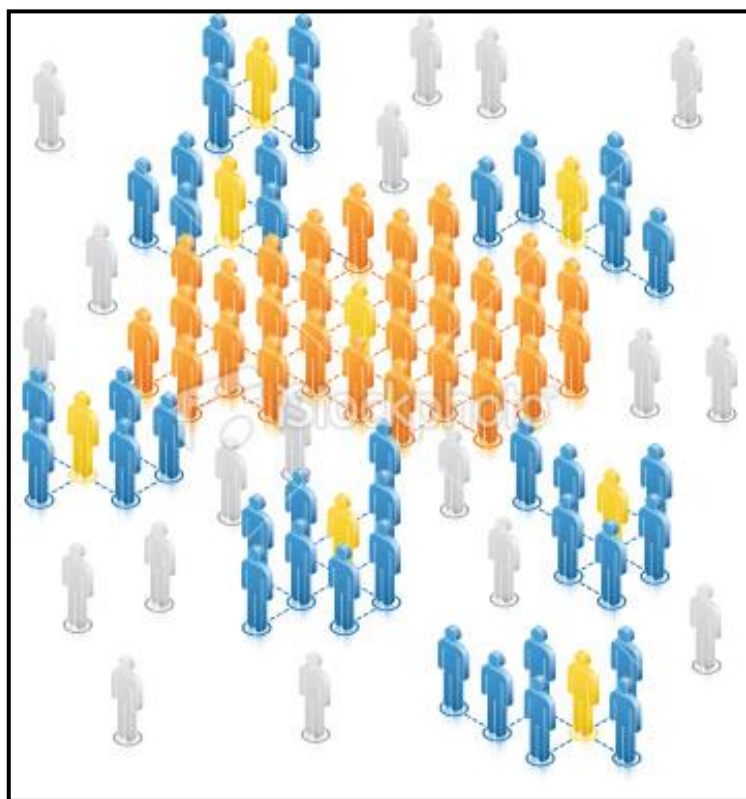
3.1 Definice sociální sítě

Sociální síť pochází z anglického spojení social network a někdy o ní můžeme slyšet jako o síti společenské nebo komunitní, zkráceně jen o komunitě. Sociálními sítěmi se zabývají věda o sítích (z anglického network science). Věda o sítích kombinuje informační technologie s teorií sítí a teorií grafů a zabývá se principy a algoritmy v síti, které určují její chování.

Sociální síť je sociální struktura složená z uzlů, obecně individuí, organizací, počítačů nebo dalších entit, která reprezentuje vztahy a spojení mezi těmito entitami. Je dobré si uvědomit, že sociální síť je každá skupina jedinců, která spolu s využitím jakýchkoli prostředků interaguje a komunikuje. Pro takovou komunitu lidí je specifický nějaký společný činitel, kterým může být například společný koníček, příbuzenské vztahy, finance, profese, obor studia atd. Příklad sociální sítě je zobrazen na obrázku 6 – barva postavičky (= uzlu, bude vysvětleno dále) potom určuje důležitost prvku v této síti, kdy šedá je prakticky nezajímavá a žlutá určuje nejpoblárnější jedince z hlediska relací (= spojení dvou uzlů v síti, také bude vysvětleno níže).

Jak již bylo naznačeno, sítě se skládají z uzlů a relací, kdy platí, že existuje konečná množina uzlů (= vrcholů) a síť je pak tvořena všemi dvouprvkovými množinami (= relacemi, vazbami, spojeními) těchto uzlů. Při definici takovéto sítě nehovoříme o

ničem jiném než grafu bez uvažování vícenásobné vazby mezi vrcholy a relace ve smyčce (vazby vycházející a končící v tom samém vrcholu).



Obrázek 6 – Ukázka sociální sítě, převzato z www.istockphoto.com

3.1.1 Uzly sociální sítě

Uzly neboli vrcholy sítě jsou aktéři, kteří se vyskytují v dané síti a jsou spojeni relacemi. Pod pojmem uzlu se nemusí vždy skrývat jen jedna entita, ale například celá skupina entit. Skupina je v rámci zjednodušení reprezentována jako jeden prvek, například firma na trhu s hračkami, sportovní tým v lize atd. Příkladem uzlů v sociální síti je například několik vybraných zaměstnanců ve firmě, které popisuje tabulka 1.

Tabulka 1 – Příklad uzlů sociální sítě

Jméno	Pozice	Věk
Tomáš	ředitel	48
Klára	sekretářka	25
Jiří	programátor	39
Karel	tester	22

Všimněme si, že při tvorbě sociální sítě je mezi uzly vždy nějaké společné pojítko. Pro nalezení vhodných aktérů do sociální sítě můžeme využít několika metod a to buď vymezení oblasti samotnými aktéry (např. zaměstnanci firmy, žáci 1. ročníku vysoké školy oboru informatiky, fanoušci fotbalového týmu, atd.), nebo vymezení oblasti demograficky popř. ekonomicky (např. lidé vydělávající ročně přes 300 tisíc Kč, rodiny s více jak dvěma dětmi, atd.).

3.1.2 Relace sociální sítě

Relace také někdy označovány jako vazby, spojení či hrany sociální sítě jsou spojnicemi dvou uzlů, které mají něco společného. V rámci sociální sítě určují nějaký vztah mezi těmito dvěma vrcholy. Takovýto vztah je měřitelný podle možných odpovědí a jejich počtu, viz [HARI-IST]:

- Binární relace
- Relace s výběrem z více možností
- Relace skupinově zaměřené
- Relace s úplným pořadím
- Relace intervalové

Binární relace patří k nejběžnějším typům vazeb. Tyto relace mají jediný úkol a to určit, zda mezi dvěma entitami nějaký vztah existuje nebo ne. Příklad vidíme v tabulce 2, kde lidé odpovídali na zjišťovací otázku: „Je tato osoba Vaším přítelem? “. V první sloupečku vidíme jména odpovídajících a v prvním řádku osoby, na které se ptáme. Odpovědi potom může být pouze ano nebo ne, což nám přesně postačuje pro vytvoření sítě pomocí binární relace. Výsledky odpovědí potom kódujeme 0 při záporné odpovědi a 1 při odpovědi kladné.

Tabulka 2 – Příklad binárních relací

Možnosti / odpovídající	Tomáš	Klára	Jiří	Karel
Tomáš	---	1	1	0
Klára	1	---	1	1
Jiří	0	0	---	1
Karel	0	1	1	---

Na relace s výběrem více možností již na položenou otázku pro zjištění vztahu mezi dvojicí uzlů neexistuje jednoduchá odpověď typu ano / ne. Dobrým příkladem je otázka: „Jaký vztah máte s touto osobou?“ a jako odpověď je výběr jedné z možností – žádný vztah, pracovní vztah, přátelé, milenci nebo příbuzní, kde jednotlivá spojení kódujeme 0 pro žádný vztah, 1 pro pracovní, 2 pro přátele, 3 pro milence a konečně 4 pro příbuzenstvo. Použijeme opět osoby z minulé tabulky tentokrát s více možnostmi odpovědi – tabulka 3.

Tabulka 3 – Příklad relací s výběrem z více možností

Možnosti / odpovídající	Tomáš	Klára	Jiří	Karel
Tomáš	---	3	2	0
Klára	3	---	1	2
Jiří	1	1	---	4
Karel	0	2	4	---

Skupinově zaměřené relace patří k často používaným, kdy vztah určuje na základě pozitivní (= 1), neutrální (= 0) nebo negativní zkušenosti (- 1). Často pak bývají doplněny o možnost žádné zkušenosti, kterou značí neexistenci této vazby. Skupinově zaměřené se často využívají pro svou jednoduchost, ale i větší obsah informace než třeba relace binární. Typickou otázkou může být: „Jaké zkušenosti máte s touto osobou?“, příkladem výsledků je tabulka 4.

Tabulka 4 – Příklad relací skupinově zaměřených

Možnosti / odpovídající	Tomáš	Klára	Jiří	Karel
Tomáš	---	1	1	-1
Klára	1	---	0	1
Jiří	0	0	---	1
Karel	-1	1	1	---

Podrobnější informace poskytují relace s úplným pořadím, kdy každý z aktérů seřadí všechny hodnocené subjekty podle oblíbenosti od 1 do N. V rámci vazby je pak každé pozici definována určitá váha. V tabulce 5 pak vidíme ukázkové výsledky seřazení svých kolegů každým zaměstnancem dle oblíbenosti.

Tabulka 5 – Příklad relací s úplným pořadím

Možnosti / odpovídající	Tomáš	Klára	Jiří	Karel
Tomáš	---	1	2	3
Klára	1	---	3	2
Jiří	2	3	---	1
Karel	3	2	1	---

Posledním a nejpokročilejším typem relací jsou relace intervalové. Jak už název napovídá, jedná se o vazby zaznamenané na stupnici v rámci pevně stanoveného intervalu, který obvykle bývá od 0 do 1. Váha relace se potom určuje reálným číslem z tohoto intervalu. Příkladem by pak mohlo být určení přátelství každého zaměstnance na své kolegy, viz tabulka 6.

Tabulka 6 – Příklad relací intervalových

Možnosti / odpovídající	Tomáš	Klára	Jiří	Karel
Tomáš	---	0,85	0,47	0,13
Klára	0,78	---	0,24	0,66
Jiří	0,44	0,37	---	0,88
Karel	0,06	0,60	0,71	---

3.2 Vlastnosti sociální sítě

Než budeme pokračovat dál, vysvětlíme si několik důležitých pojmů a vlastností, kterými budeme charakterizovat celou sociální síť, její aktéry a relace mezi nimi. Hned

na začátek bychom si měli uvést rozdíl mezi sítí orientovanou (např. Facebook) a neorientovanou (např. Twitter). Orientovaná síť je taková síť, jejíž hrany jsou uspořádané dvojice oproti hranám neorientované sítě, kde to jsou dvouprvkové množiny.

Sítě dále dělíme na spojitě, nespojitě nebo egocentrické, jak ukazuje [ENDI-SNA]. Spojitá síť je typická tím, že se od jednoho aktéra dostaneme přes libovolný počet hran k aktérovi druhému. Nespojitá síť tuto vlastnost nesplňuje a existuje v ní izolovaný prvek nebo prvky. Poslední typem je síť egocentrická, ve které se nachází uzel, jenž je spojen se všemi ostatními uzly.

3.2.1 Vlastnosti uzlů sociální sítě

Co je to uzel, aktér, vrchol a přeneseně z teorie o důvěře také subjekt či entita sociální sítě již víme, zde si uvedeme nejdůležitější vlastnosti, kterými ho v síti popíšeme:

- Stupeň
- Sousedství
- Dosažitelnost
- Vzdálenost

Stupeň vrcholu sítě je velikost, která je rovna počtu hran z vrcholu vycházející. U neorientované sítě je to jednoduché a určujeme celkový stupeň uzlu. U sítí orientovaných určujeme vstupní a výstupní stupeň uzlu. Vstupní odpovídá počtu hran, které do uzlu vstupují, což nám dává informaci, kolika cestami může k takovému aktérovi dorazit zpráva. Výstupní stupeň se naopak rovná počtu hran z tohoto uzlu vystupujících, jinými slovy počet spojení, kterými může aktér komunikovat s ostatními.

Pokud jsou dva vrcholy sítě spojeny vazbou, říkáme, že spolu sousedí. Všechny vrcholy, se kterými pak daný vrchol sousedí, nazýváme sousedství či okolí vrcholu. Velikost sousedství je potom vlastně stupněm vrcholu.

Dosažitelnost mezi uzly určuje, zda existuje cesta mezi dvěma vybranými uzly a její velikost je taková, kolika hranami musí projít zpráva od jednoho aktéra, aby ji druhý přijal. Pokud taková cesta neexistuje, jsou tyto dva vrcholy nedosažitelné. Počet hran spojující dva vrcholy sítě se nazývá také vzdálenost. V neorientovaném grafu platí, že

pokud existuje cesta z uzlu 1 k uzlu 2, pak existuje i cesta z uzlu 2 k uzlu 1. U sítí orientovaných už to platit nemusí, takže ačkoli existuje cesta z uzlu 1 do uzlu 2, už nemusí existovat cesta z uzlu 2 do uzlu 1 a obráceně.

3.2.2 Vlastnosti relací sociální sítě

Relace, spojení nebo také vazby a jejich druhy jsme si popsali již v kapitole 3.1.2. Nyní se zaměříme na jejich vlastnosti:

- Reciprocita
- Tranzitivita

Reciprocita relace určuje vyváženost této vazby. Aktéry v síti podle [HARI-IST] dělíme na příjemce (z anglického sinks) a zdroje (z anglického sources). Přijímač je definován jako ten, kdo zprávy od jiných aktérů přijímá. Naopak zdroj je ten, kdo zprávy posílá. Většinou se potom v síti setkáváme s aktéry, kteří jsou zároveň přijímači a zdroji a právě reciprocita určuje vyváženost těchto dvou činností. Tyto aktéry potom nazýváme jako přenašeče (z anglického transmitters).

O tranzitivitě relací mluvíme, existuje-li cesta z uzlu 1 přes uzel 2 do uzlu 3, přičemž uzly 1 a 3 nejsou přímo spojeny. Tohoto faktu budeme moci využít při posílání doporučení na subjekt při modelování důvěry, o kterém se budeme bavit v kapitole 4.

3.2.3 Vlastnosti celé sociální sítě

Pro celou sociální síť vymežíme dvě důležité vlastnosti, kterými ji popíšeme. Mezi ně patří:

- Velikost
- Hustota

Velikost sítě je určena počtem hran a přímo se odvíjí od počtu aktérů, kteří v síti figurují. Jestliže máme n uzlů, tak počet všech vazeb je maximálně $n * (n - 1)$. V orientované síti je toto číslo konečné, v neorientované je navíc poloviční, jelikož relace mezi uzlem 1 a uzlem 2 je stejná jako mezi uzlem 2 a uzlem 1. Velikost sítě vypovídá o rychlosti, bezpečnosti a spolehlivosti s jakou je přenesena zpráva od aktéra 1 k aktérovi 2.

Hustota sítě nám říká jak moc je nebo není síť propojená vazbami a je tedy určena počtem vazeb, které se v síti nacházejí. Představme si malou firmu o 20 zaměstnancích, je nanejvýš pravděpodobné, že se všichni znají a tak bude hustota sítě určitě vysoká. Naopak velké korporace s několika stovkami zaměstnanců již hustotu tak vysokou mít nebudou.

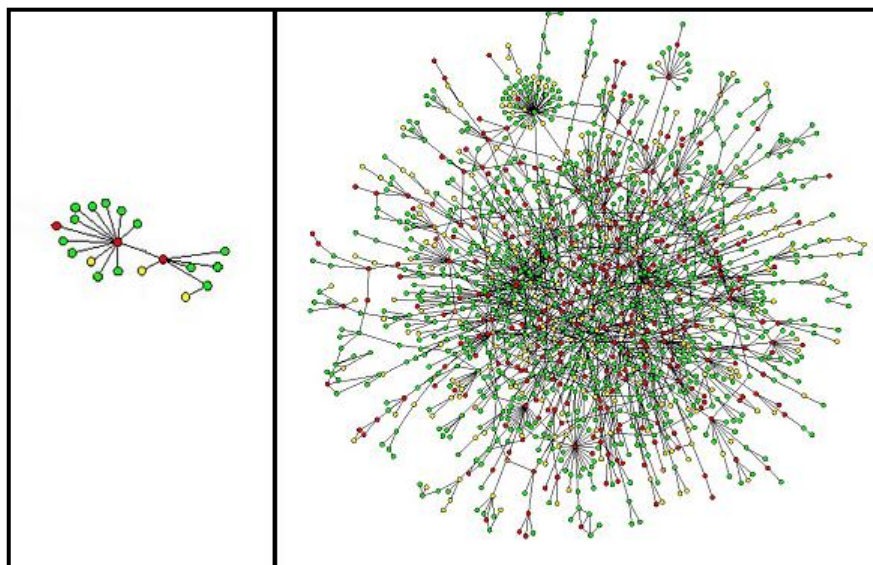
3.3 Reprezentace sociální sítě

K prezentaci sociální sítě využíváme teorie grafů. Jedním ze způsobů zobrazení sítě je vykreslení sítě jako samotného grafu. Graf potom můžeme implementovat různými způsoby:

- Matice incidence (Node-Arc Incidence Matrix)
- Matice sousednosti (Node-Node Adjacency Matrix)
- Seznamy sousednosti (Adjacency Lists)

Všechny zápisy jsou si naprosto rovny, ale každý ze zápisů má své specifické využití a záležit.

Nejjednodušším zakreslením sítě je využití grafu. Taková reprezentace však může být při větším počtu uzlů a hran nepřehledná. V našich příkladech si s ní ale většinou vystačíme. Na obrázku 7 vidíme rozdíl v přehlednosti reprezentace sítě s využitím grafu v závislosti na počtu uzlů a hran.

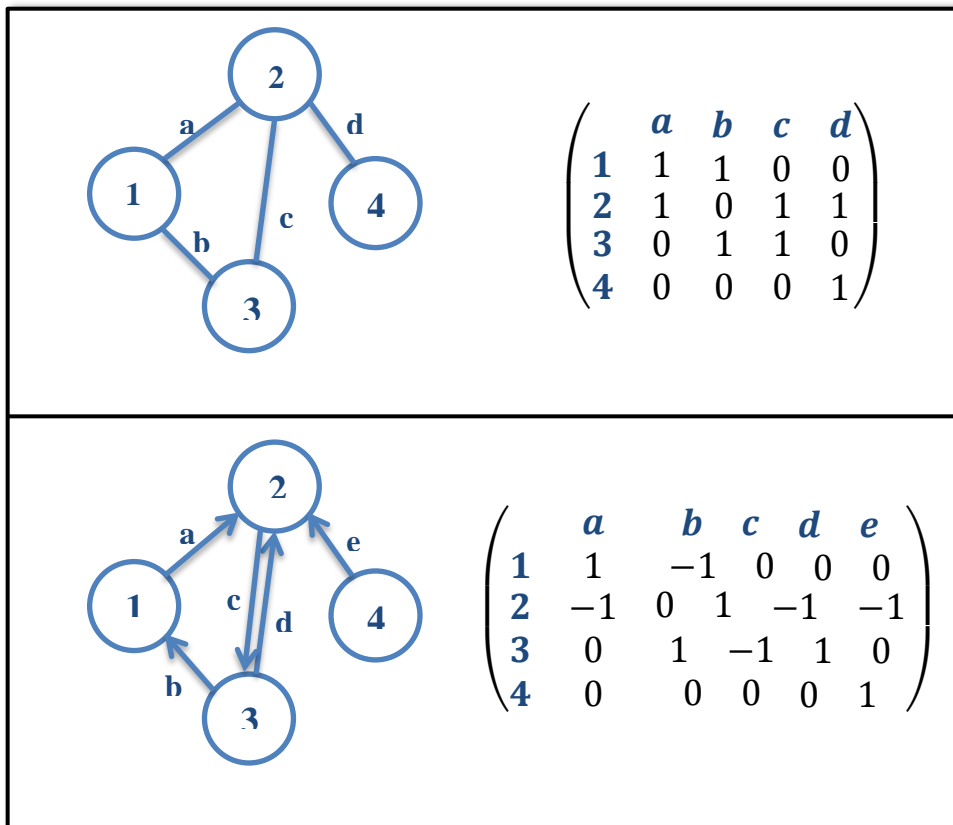


Obrázek 7 - Malá a velká sociální síť zobrazená grafem, převzato z [BORD]

Při užití matice incidence odpovídá každý řádek matice jednomu vrcholu a každý sloupec potom jedné hraně sítě. V matici vidíme, zda mezi daným vrcholem a danou hranou existuje spojení. Konkrétní zápis se trochu liší v případě neorientované a orientované sítě:

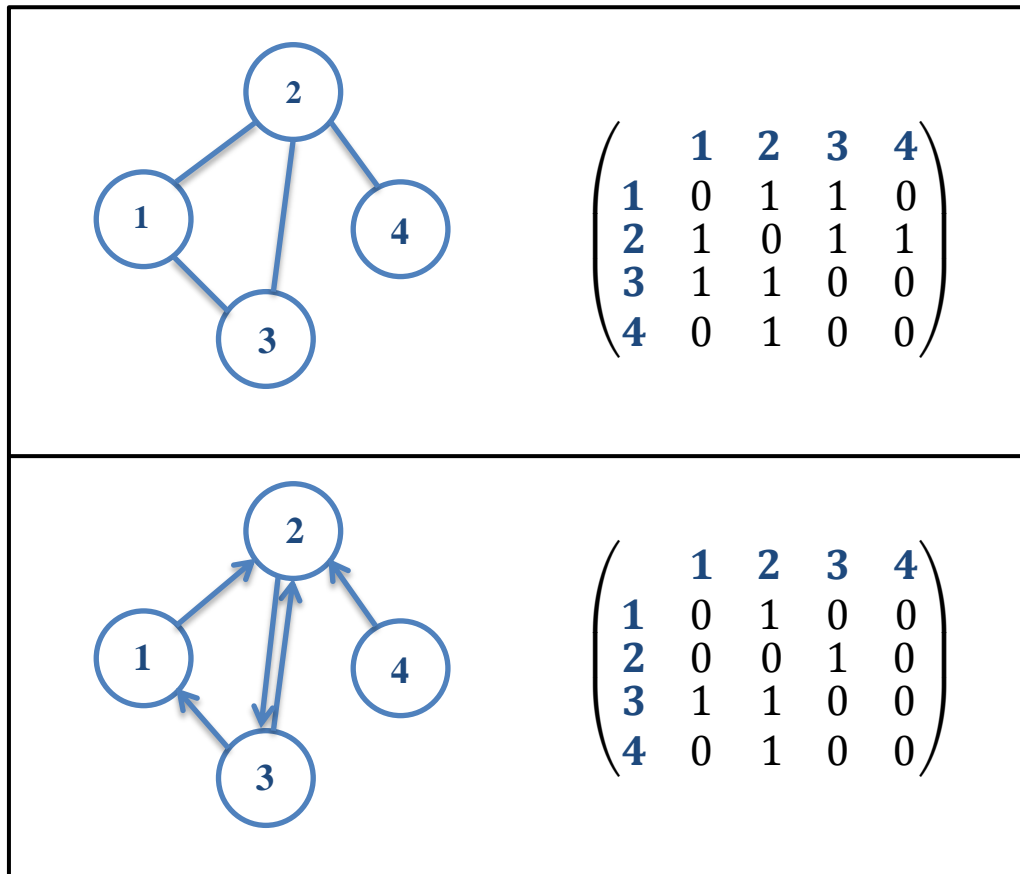
- Neorientovaná síť:
 - 1 – existence spojení mezi vrcholem sítě a hranou
 - 0 – neexistence spojení mezi vrcholem sítě a hranou
- Orientovaná síť:
 - 1 – existence spojení mezi vrcholem a hranou, hrana vede z daného vrcholu
 - 0 – neexistence spojení mezi vrcholem a hranou
 - -1 – existence spojení mezi vrcholem a hranou, hrana vede do daného vrcholu

Je důležité si uvědomit, že matice incidence neumožňuje zakreslovat smyčky a zpravidla bývá obdélníková. Příklad vidíme na obrázku 8.



Obrázek 8 - Matice incidence neorientované a orientované sítě

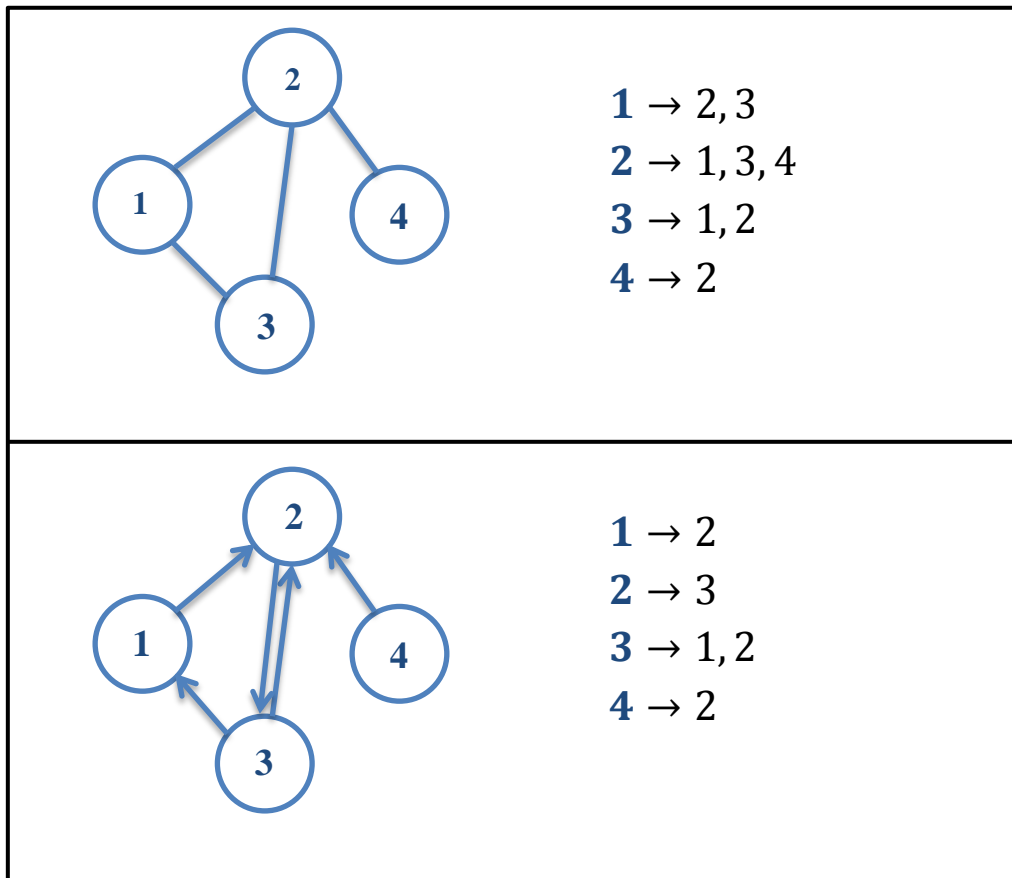
Další možností je matice sousednosti. Řádky i sloupce matice odpovídají vrcholům sítě a matice sousednosti je vždy čtvercová. Rozměr této matice odpovídá počtu uzlů sítě. Řádkový index značí vrcholy, ze kterých hrana vychází, a sloupcové, do kterých daná hrana vchází. Existenci hrany potom značíme 1 a neexistenci 0. Je zřejmé, že matice sousednosti neorientovaného grafu je symetrická¹. Příklady pak vidíme na obrázku 9.



Obrázek 9 - Matice sousednosti neorientované a orientované sítě

Listy sousednosti je jednoduchá reprezentace grafu, kdy pro každý uzel sítě existuje spojový seznam obsahující všechny uzly, do kterých vede hrana z daného uzlu. V případě neorientovaných sítí potom platí, že v každém listu jsou všechny uzly spojující daný uzel s jiným uzlem. Příklad pro orientovanou a neorientovanou síť potom vidíme na obrázku 10.

¹ Symetrická matice je taková matice, kdy záměna řádků a sloupců vytvoří stejnou matici.



Obrázek 10 - Listy sousednosti neorientované a orientované sítě

3.4 Sociální sítě v informačních technologiích

Když dnes použijeme pojem sociální sítě, všichni nejprve vzpomenou na sociální síť Facebook, ačkoliv sociální sítě existuje opravdu velké množství. V této kapitole si některé z nich popíšeme víc a to jak celosvětové, tak české. Počty uživatelů zmiňované u každé sítě byly převzaty z [TRR] a byly aktuální k 27. lednu 2012.

3.4.1 Facebook

Facebook je celosvětově rozšířená sociální síť, která slouží ke komunikaci mezi uživateli, sdílení informací, fotek, videí, hraní her či plánování a zvaní na různé akce. Jak již bylo zmíněno hned v úvodní kapitole, Facebook měl v únoru tohoto roku přes 800 miliónů aktivních uživatelů, kteří ho použili alespoň jednou měsíčně.

Funkce a aplikace na Facebooku nalezneme skoro na vše, které bychom si pro naši zábavu a pohodlí mohli přát. Dodává je jak sám poskytovatel Facebooku, tak i společnosti třetích stran. Právě s nimi se po této síti objevuje nemalé množství reklamy

a komerčního obsahu, ač Facebook sám o sobě je nekomerčním projektem vývojáře Marka Zuckerberga již od roku 2004. To byl ovšem Facebook jen sociální síť Harvardovy univerzity v USA, odtud se pak během dalších několika let rozšířil do celého světa a stal se největší a nejpoužívanější sociální sítí naší doby.

3.4.2 Twitter

Twitter je druhá nejpoužívanější sociální síť čítající přes 200 miliónů uživatelů. Twitter byl založen roku 2006 Jackem Dorseyem a dnes se o něm často hovoří jako o „SMS internetu“. Jeho hlavní myšlenkou je mikroblogging a odesílání krátkých zpráv. Tyto zprávy (hovorově „tweety“) se zobrazují na profilové stránce vkladatele a mohou si je přečíst všichni odběratelé tohoto uživatele. Pokud se uživatel sám stane odběratelem, může neustále přijímat krátké zprávy o novinkách a zajímavostech s odkazem na celé zprávy. Právě díky tomu je Twitter neocenitelným nástrojem, který využívá velké množství lidí, z nemalé části pak odborníků či interesantních společností. Leckdy se zde člověk může dozvědět o věcech, které se v klasických masmédiích objeví až mnohem později.

3.4.3 Google Plus

Nejmladší ze všech zmiňovaných sítí založená teprve v minulém roce, konkrétně 28. června 2011. Google Plus se již dnes může pyšnit 90 milióny uživatelů a stává se tak možná jednou z největších konkurencí Facebooku. Stejně jako zmiňovaný Facebook pomáhá lidem v komunikaci a sdílení informací a multimédií. Nespornou výhodou je podobnost s Twitterem, kdy uživatelé sami určují, kdo může sledovat jejich komentáře a statusy a od koho je budou získávat oni, aniž by se s dotyčným museli přátelit. Aplikace Googlu Plus jsou převážně již dříve známé aplikace Googlu, které jsou nyní spojeny přes jediný účet do jednoho většího systému či se jmenují jinak a fungují na stejném principu. Právě těmito nepřebornými možnostmi spojení této sítě s vyhledávačem, překladačem, sdílením a zpracováním dokumentů, kalendářem a mnoha dalšími se Google Plus na poli sociálních sítí velmi zviditelnil a je otázkou, zda si k němu uživatelé najdou cestu jako k Facebooku.

3.4.4 LinkedIn

LinkedIn je specifickou sociální sítí, na které se setkávají lidé, kteří svým profilem vystavují svůj profesní život a sdílejí převážně pracovní kontakty. Každý profil je vlastní CV osoby zobrazené na internetu, kde potenciální zaměstnavatel vidí uživatelské vzdělání, profesní historii a praxi, získané certifikáty a podobně. LinkedIn často využívají různé personální agentury pro své klienty. Hlavním rozdílem mezi LinkedIn a Facebook je formálnost sítí, kdy Facebook je používán hlavně pro zábavu a LinkedIn pro profesní a odborný život.

V dnešních dnech má LinkedIn přes 135 milionů uživatelů získaných od roku 2003, kdy byl server poprvé spuštěn. Zajímavostí jistě je, že během prvních šesti let LinkedIn dosáhl hranice 50 milionů a pak za další rok a půl tuto hranici zdvojnásobil. Je tedy vysoce pravděpodobné, že celkový počet uživatelů i nadále poroste.

3.4.5 Lidé.cz

Lidé.cz je známá česká sociální síť provozovaná firmou Seznam.cz, která čítá okolo 1,5 milionů uživatelů. Při srovnání se světovými sítěmi je to opravdu málo, ale v porovnání s celkovým počtem obyvatel České republiky to již tak zanedbatelné číslo není.

I na této sociální síti má každý uživatel svůj profil se svými informacemi, fotkami a videi, které může sdílet s ostatními. Ke komunikaci pak slouží posílání zpráv a různé diskuzní kroužky. V rámci této sítě existuje i seznamka s desítkami tisíc inzerátů, které můžete propojit s vlastním profilem.

3.4.6 Spolužáci.cz

Poslední zde zmíněnou sociální sítí jsou Spolužáci.cz provozované taktéž firmou Seznam.cz a přibližně i stejným počtem uživatelů jako Lidé.cz. Spolužáci.cz slouží ke spojování současných a bývalých spolužáků škol z celé České republiky. Uživatelé jsou potom rozděleni do jednotlivých tříd podle školy a ročníku a mohou mezi sebou měnit informace a multimediální soubory. Diskuze pak probíhá nejenom v rámci třídy ale i školy a mezi žáky a jejich vyučujícími.

4 Model důvěry v sociální síti

Model důvěry vzniká za účelem zkoumání důvěry na základě vytváření a vyhodnocování vztahů mezi určenými objekty. V této kapitole uvedeme nejčastější přístupy k modelům důvěry, které jsou ale více dopodrobna popsány v mé bakalářské práci [VAC-BP]. Dále si zde popíšeme některé technologie a implementační nástroje pro jejich realizaci.

4.1 Přístupy pro tvorbu modelů důvěry

Přístupů, jak reprezentovat vývoj důvěry a ukázat, jak se jednotlivé objekty ovlivňují, existuje nepřeberné množství. Je důležité si hned na začátku uvědomit, jak na důvěru chceme pohlížet a podle toho si vybrat vhodný model využívající:

- Teorie grafů
- Teorie pravděpodobnosti
- Teorie her
- Teorie rozhodování
- Teorie informace a entropie
- Teorie šíření informací a šíření epidemií

Při využití teorie grafů či teorie pravděpodobnosti na důvěru pohlížíme jako na hledání vhodné strategie pro splnění společného cíle. Popisujeme totiž jevy, které mohou, ale nemusí nastat, záleží na možnostech volby a konečný výsledek je nejistý. Pro lepší představu si uveďme například tip výsledku utkání, hod kostkou nebo sázku do loterie.

Z pohledu teorie her, bereme modelování důvěry jako analýzu konfliktních situací a snahu získat nejlepší strategii pro každou entitu, která se modelování účastní. Konfliktní situací máme na mysli jakoukoli situaci, kde okolní účastníci reagují na naše předchozí rozhodnutí. Dobrým příkladem je evoluce chování známé jako Vězňovo dilema, kdy existuje oboustranně výhodné řešení, ale je nedostupné. Viz [BJÖ-CCC].

Pokud se zabýváme racionálním výběrem z několika možností na základě kladného nebo záporného hodnocení, je vhodné využít k modelování důvěry teorii rozhodování. Rozhodování je chápáno jako proces, který vede k výběru nejlepšího řešení, kdy každé rozhodnutí produkuje konečnou volbu, akci a stanovisko. Teorie rozhodování se

využívá v mnoha modelech s problémem optimalizace výběru z několika alternativ na základě počátečních podmínek.

Při kladení důležitosti na samotné informace, jejich přenos, rušení, kódování a měření využíváme v modelech důvěry teorii informace, o které najdeme více v práci [MCE-TIC]. Komunikace ve vztahu s důvěrou má zásadní význam, díky ní mezi aktéry důvěra vzniká a dále se vyvíjí.

Posledním důležitým přístupem je šíření informací a šíření epidemií, které má základy v živé přírodě. Klasickou ukázkou je vir a jeho šíření mezi tvory, co s ním přišli do styku. Modely založené na šíření epidemií a informací se využívají pro svou jednoduchost a odolnost vůči chybě zejména ve velkých systémech velice často.

4.2 Modelování důvěry

Metodik pro modelování důvěry známe spoustu a záleží na mnoha aspektech s ohledem na samotný fenomén důvěry. Každý model je typický mírou racionality, měřitelnosti, prostředí a jeho velikostí. Rozlišujeme, zda se jedná o centralizovaný či decentralizovaný model a podobně.

Existují dva typy důvěry, pro které máme v několika ohledech odlišné modely a styly přístupu:

- Personální důvěra
- Impersonální důvěra (nepersonální, fenomenální)

Personální důvěra je typická vývojem důvěry mezi dvěma subjekty. Tyto subjekty jsou schopné důvěru tvořit či ji opřevzat. Nejčastěji to jsou lidé a zvířata.

Impersonální důvěra má charakter důvěry v nějaký fenomén. Pod fenoménem se může schovávat velké množství podobných produktů. Produkty jedné kategorie mají podobné znaky či vlastnosti, podle nichž je hodnotíme. Pro příklad uvedeme několik značek pracích prášků či různé rádiové stanice.

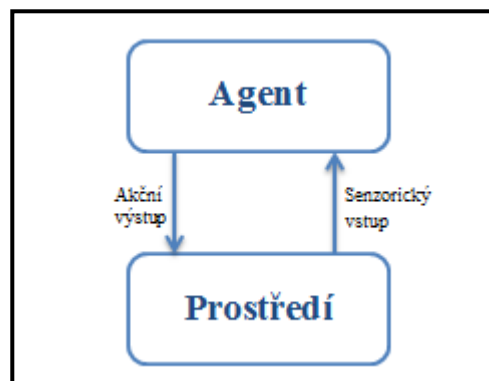
4.3 Technologie a nástroje pro tvorbu

K realizaci modelů důvěry můžeme využít několika technologií. K těm nejtypičtějším patří distribuované umělé inteligence a multiagentní systémy. Dobře pospaná je tato problematika v [NET-MSTE]. Pro naše účely se více zaměříme na

multiagentní systémy, které budou využity v softwarovém systému a několika experimentech.

4.3.1 Multiagentní systémy

Multiagentní systémy se skládají z agentů a prostředí, v němž žijí. Agent je počítačový systém schopný reagovat na prostředí, ve kterém se nachází s cílem splnit určený úkol. Výsledkem činnosti je převedení agentního systému ze současného stavu do stavu cílového. Podle způsobu transformace potom rozeznáváme různé druhy agentních systémů. Ukázka agenta a jeho prostředí vidíme na obrázku 11. Více o agentních systémech nalezneme v [KUB-IAT].



Obrázek 11 - Agent a jeho prostředí

Agenti mají tyto vlastnosti:

- Autonomnost – je proaktivní, má určitý cíl, je schopný řešit určitou úlohu bez pomoci a nadbytečné komunikace s ostatními, je schopný spolupracovat s dalšími agenty v určité komunitě, má možnost tuto komunitu opouštět či do ní vstupovat
- Reaktivita – je aktivován událostí, je schopný reakce v určitém čase
- Intencionalita – má zapamatované dlouhodobé cíle, umí dosáhnout daného cíle, má vlastní strategii, jak cíle dosáhnout
- Schopnost sociálního chování – je schopen spolupracovat pro splnění cíle, je schopen udržovat informace o dalších agentech a mít na ně vlastní názor, je schopen sdružovat se v komunitách při plnění cíle

Mezi významné nástroje pro tvorbu a modelování multiagentních systémů patří především JADE, RETSINA a SWARM.

4.3.1.1 JADE

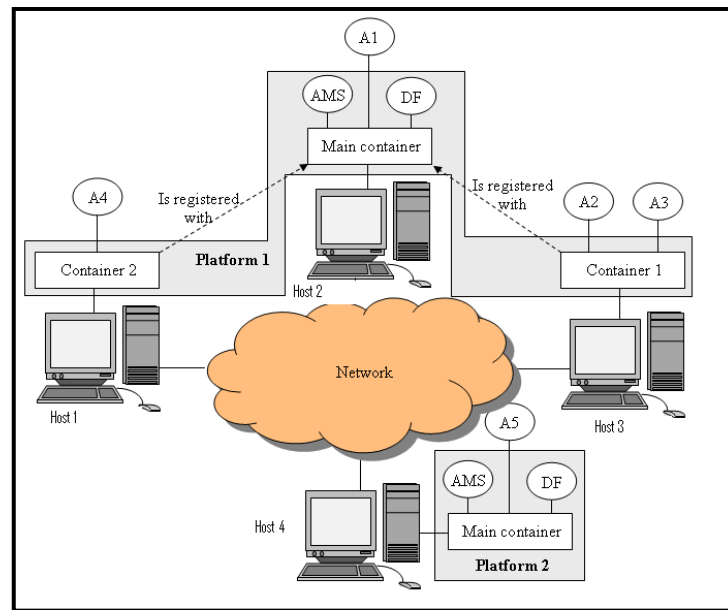
JADE (Java Agent Development Framework) je volně šiřitelná platforma pro vývoj P2P agentově založených aplikací řídicích se standardy FIPA (Foundation for Intelligent Physical Agent). JADE byl vyvinut organizací Telecom Italia Lab v programovacím jazyce Java a je poskytován pod licencí LPGL. Více informací nalezneme v [JADE].

JADE je middleware pro snadnou tvorbu multiagentních aplikací a obsahuje tyto části:

- Prostředí pro agenty – prostředí, kde jsou agenti aktivní
- Knihovny pro vývoj agentů – třídy a jejich modifikace
- Balík grafických nástrojů – správa a monitoring běžících agentů

JADE platforma je složena z agentních kontejnerů a samotní agenti jsou vytvořeni jako běžící instance JADE RE (JADE Runtime Environment). Každá platforma má jeden hlavní kontejner, který se spustí jako první a všechny ostatní se k němu zaregistrují. Hlavní kontejner má navíc další specifické vlastnosti na rozdíl od ostatních kontejnerů – spravuje tabulku kontejnerů (container table) a globální tabulku agentních deskriptorů a hostuje agenty AMS (Agent Management System) a DF (Directory Facilitator). Základní prvky architektury jsou ukázány na obrázku 12.

Agent se vytváří pomocí definice třídy dědicí od `jade.core.Agent` a je realizován jako třída zapouzdřující životní cyklus agenta. Činnost agenta je implementována jako objekt zděděný od `jade.core.behaviours.Behaviour`. Agent může provdět několik takových chování současně s tím, že plánování chování není pre-emptivní ale kooperativní. Každý agent je v souladu s FIPA specifikací označen vlastním identifikátorem. Ten je reprezentován jako instance třídy `jade.core.AID`.

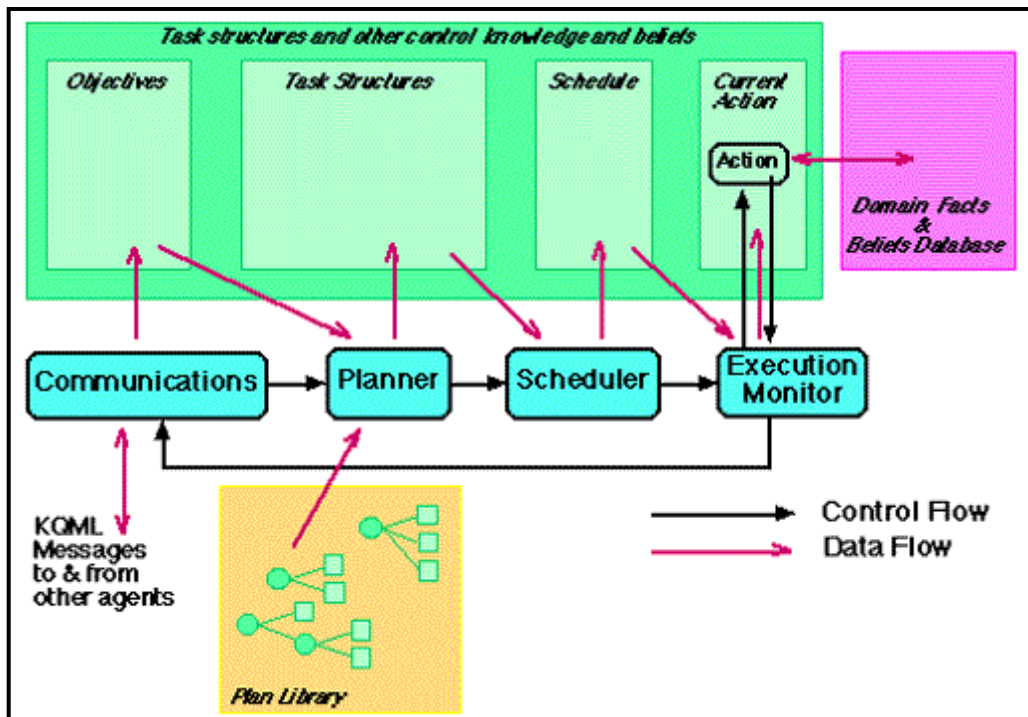


Obrázek 12 - Hlavní prvky architektury JADE, převzato z [JADE]

4.3.1.2 RETSINA

RETSINA (Reusable Environment for Task Structured Intelligent Network Agents) [RETS] je multiagentní systém vyvinutý v Laboratořích Softwarových Agentů v Institutu Robotiky na Carnegie Mellon University. Knihovny tohoto softwaru existují ve dvou verzích. První využívající programovacího jazyka C++ a používá se na zařízeních s limitními zdroji (mobilní zařízení a vestavěné systémy) a druhá v programovacím jazyce Java, která je zatím ve vývoji. Knihovny se nedají volně stáhnout a pro jejich získání je zapotřebí kontaktovat Institut Robotiky.

Celá implementace je založena na architektuře nazývané se RETSINA AFC (Agent Foundation Classes) a skládá se ze čtyř znovupoužitelných modulů – modul pro komunikaci a koordinaci, modul pro plánování, modul pro rozvrhování a modul pro zpracování. Tato architektura je zachycena na obrázku 13.



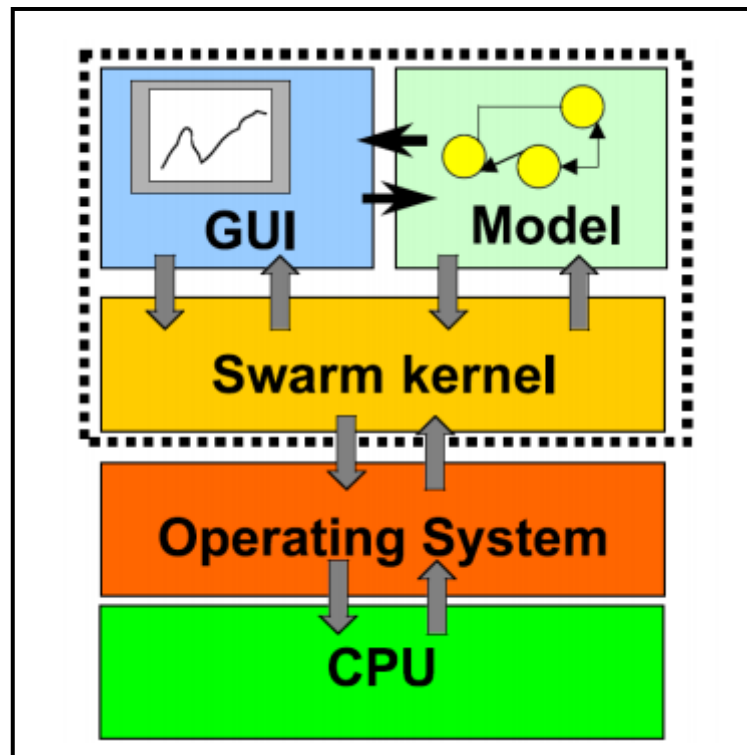
Obrázek 13 - Hlavní prvky architektury RETSINA, převzato z [RETS]

RETSINA se využívá v mnoha oblastech jako je například správa finančního portfolia, mobilní zařízení, vyhledávání informací na webu, plánování logistiky a další.

4.3.1.3 SWARM

SWARM (Swarm Intelligence) [SWA] je programový balík pro simulaci složitých multiagentních systémů vyvíjený v Institutu v Sante Fe a nyní běží v organizaci Swarm Development Group. SWARM je implementován v programovacích jazycích Objective-C a Java, je dostupný pod GNU licencí a stále se vyvíjí.

SWARM knihovny jsou znázorněny jako vrstva nad jádrem operačního systému a je tak možné přerušit simulaci a měnit vnitřní stav agentů. Ukázka virtuálního stroje je naznačena na obrázku 14.



Obrázek 14 - SWARM virtuální stroj, převzato z [SWA]

SWARM vytváří komunity agentů, které na sebe v daném prostředí navzájem působí. Jedná se o nízko-úrovňové interakce a systémy jsou označovány jako komplexní. Celý SWARM je postaven na objektovém programování. Objekty (agenti) jsou schopné udržovat informace a třídit je.

4.4 Personální model důvěry v sociální síti

Model důvěry vychází z modelu personální důvěry, který je popsán v [NET-MD], a modelu personální důvěry v sociální síti z [HAV-MSC]. Jak již bylo naznačeno, personální důvěra je typem důvěry, kterou projevuje jeden subjekt jinému subjektu či skupině subjektů.

Model důvěry z [NET-MOD] určuje míru důvěry v závislosti na velikosti důvěry nasbírané v minulosti, na počtu vzájemných kontaktů (kolikrát se subjekty setkaly), na počtu doporučení (známost založená na dalších subjektech), na všeobecné reputaci a pověsti každého subjektu a v poslední řadě na náhodné veličině představující momentální dispozici k důvěře (nálada, intuice, prostředí, situace).

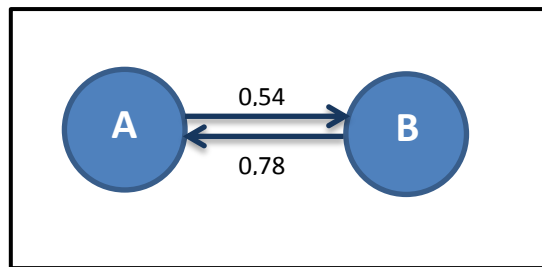
Model personální důvěry v sociální síti z [HAV-MCOD] si také zakládá na důvěře získané v minulosti, obdržení doporučení, reputaci a náhodné dispozici k důvěře.

Zde popsany model vychází hlavně z druhého zmíněného modelu, který rozšiřuje, ač většina veličin zůstává stejná a liší se jen jejich výpočet.

Míru personální důvěry v sociální síti definujeme v iteracích v časových intervalech. Míra důvěry se vyvíjí mezi dvěma subjekty s_i a s_j , které se obecně nazývají poskytovatel důvěry (trustor) a příjemce důvěry (trustee). Tuto míru důvěry vyjádříme jako hodnotu v intervalu $\langle 0, 1 \rangle$. Lze tedy psát:

$$t(s_i, s_j) = t_{ij}, \text{ tzn.: } T: S \times S \rightarrow \langle 0, 1 \rangle, \text{ kde } i, j = 1, \dots, n (i \neq j) \quad (4.4.1)$$

Model reprezentujeme jako orientovaný ohodnocený graf, kde jednotlivé uzly představují subjekty a jednotlivé hrany spojují ty subjekty, mezi kterými se důvěra utváří. Orientace hran poté naznačuje směr důvěry a odlišnost důvěr mezi dvěma subjekty, viz obrázek 15, tzn. $t_A \neq t_B$ (subjekt A důvěřuje subjektu B jinak než subjekt B subjektu A). Ohodnocení hrany udává míru důvěry i-tého subjektu j-tému.



Obrázek 15 - Asymetrie míry důvěry

Míra aktuální důvěry T_{ij} i-tého subjektu j-tému subjektu je dána takto:

$$T_{ij} = F(t_{ij}, t_{ji}, d_{ij}, r_i, g_i), \quad (4.4.2)$$

kde:

t_{ij} určuje původní míru důvěry i-tého subjektu v j-tý subjekt

t_{ji} určuje míru důvěry j-tého subjektu v i-tý subjekt

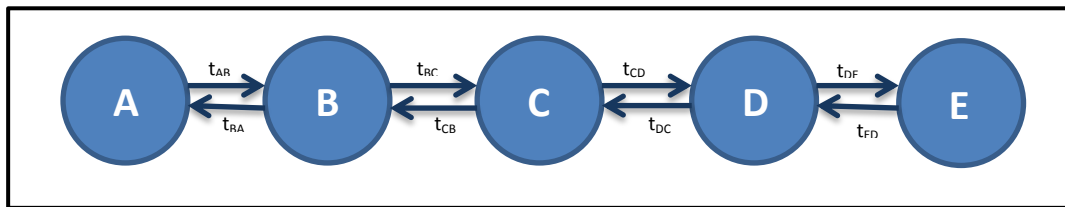
d_{ij} určuje počet doporučení j-tého subjektu i-tému jinými subjekty

r_i určuje reputaci i-tého subjektu v sociální síti

g_i určuje dispozici k důvěře podle dané nálady, situaci, intuici, atd.

Protože se důvěra vyvíjí v čase, jsou časové intervaly v modelu řešeny iteračně. Jak dlouhý je časový interval, už závisí na situaci, typu modelu a jeho použití.

Pod pojmem doporučení si můžeme představit kladné ohodnocení subjektu C subjektem B, které je předáno subjektu A. Subjekt B přitom zná jak subjekt C, tak subjekt A. Naopak subjekty A a C se neznají a jediným jejich prostředníkem je právě subjekt B. Velikost tohoto doporučení je dána mírou důvěry subjektu A a společného prvku B a důvěrou subjektu B a doporučovaným C. Celkové doporučení je průměrem všech doporučení mezi dvěma subjekty. Je vidět, že velikost doporučení se odráží na velikosti důvěry mezi prvky, čímž docílíme velikosti doporučení přesně takové, jak silný je vztah mezi těmito prvky. Tímto jsme si ukázali, jak to vypadá při využití cesty grafu o velikosti 2. Můžeme pokračovat ještě dál, viz obrázek 16.



Obrázek 16 - Jednoduchá síť s pěti aktéry

Obecně je možné, aby subjekt B udělil doporučení na subjekt D, který zná přes subjekt C, subjektu A. Velikost tohoto doporučení potom určíme naprosto stejným způsobem jako sílu důvěry mezi prvky A a B, B a C a nakonec C a D. Protože zde již může docházet k jistým zkreslením, je nutné toto doporučení upravit koeficientem vzdálenosti doporučení λ daným délkou cesty v grafu. Velikost koeficientu je zobrazena v Tabulce 7.

Tabulka 7 - Koeficient vzdálenosti doporučení

Velikost koeficientu pro výpočet velikosti doporučení	
Délka cesty v síti mezi doporučeným prvek a příjemcem doporučení	Koeficient vzdálenosti doporučení λ
2	1
3	1/2
4	1/3
5	1/4

Teoreticky by bylo možné nadále pokračovat v tomto navyšování vzdálenosti, ale v praktických výpočtech pak dochází stále k většímu zkreslení výsledku. Pokud si takovou situaci představíme v reálném životě, také si do svého rozhodování o člověku, kterého nám náš známý doporučil jako dobrého elektrikáře, o kterém mu vyprávěl jeho

kamarád, když mu jedna známá v práci říkala, že jejímu otci dělal elektřinu, nenecháme příliš zasahovat.

Doporučení d_{ij} j-tého subjektu i-tému subjektu je dáno jako

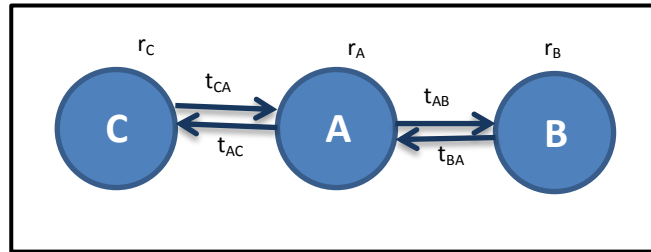
$$d_{ij} = \lambda * \frac{\sum_0^n t_{ik_1} * t_{k_1 k_2} * \dots * t_{k_{m-1} k_m} * t_{k_m j}}{n}, \quad (4.4.3)$$

kde λ je koeficient vzdálenosti doporučení, $k_{1,2,\dots,m}$, kde $k_{m=1,2,\dots}$, je označení společných prvků a n počet relevantních poskytovatelů důvěry. Při výpočtu je nutné použít změnu doporučení v jednotlivých iteracích. Platí, že

$$\Delta d_{ij} = \frac{d_{ij}^l - d_{ij}^{l-1}}{n-1}, \quad (4.4.4)$$

kde l značí právě iteraci při vývoji důvěry. Hodnota důvěry leží na intervalu $\langle 0,1 \rangle$, tj. $d_{ij} = \langle 0, 1 \rangle$.

Reputace prvku udává postavení prvku mezi ostatními prvky v síti, jeho oblíbenost u ostatních a jeho pověst, kterou o něm šíří jeho známí, přátelé a nepřátelé. Reputaci každého subjektu získáme jako vážený průměr reputací daného prvku a prvku známého, který subjektu reputaci udílí. Na obrázku 17 vidíme tři subjekty A, B, C, které mezi sebou sdílí důvěru o velikosti t_{CA}, t_{AC} mezi prvky A a C a t_{AB}, t_{BA} mezi prvky A a B. Každý z aktérů má svou reputaci označenou r_i .



Obrázek 17 - Síť tří prvků s reputacemi

Pro výpočet reputace r_i využijeme vzorce

$$r_i = \frac{\sum_0^n r_i * r_m}{n}, \quad (4.4.5)$$

kde r_i je reputace udílená prvku i prvkem m , r_m je reputace prvku m a n je počet udělených reputací. Reputace r_i nabývá hodnot v intervalu mezi 0 a 1, tj. $r_i = \langle 0,1 \rangle$.

Aktuální dispozice k důvěře g_i je důležitá složka ovlivňující důvěru. Je závislá na momentální náladě a intuici subjektu, na okolním prostředí či situaci. Právě tato veličina

se může u subjektů velmi často měnit. Stejně jako ostatní veličiny i aktuální dispozice k důvěře je hodnota z intervalu $\langle 0,1 \rangle$.

Výsledná důvěra T_{ij} mezi dvěma subjekty je potom dána jako součet dosavadní míry důvěry t_{ij} a přírůstku (úbytku) důvěry:

$$T_{ij} = t_{ij} + \Delta t_{ij}. \quad (4.4.6)$$

Přírůstek (úbytek) důvěry je potom dán jako:

$$\Delta t_{ij} = \sqrt{t_{ij} * t_{ji}} * \Delta d_{ij} * w_{d_i} * r_i * w_{r_i} * g_i * w_{g_i}. \quad (4.4.7)$$

Geometrický průměr $\sqrt{t_{ij} * t_{ji}}$ je reprezentace trendu ve vývoji důvěry mezi i-tým a j-tým subjektem. Veličiny d_{ij} , r_i a g_i byly již podrobně popsány výše a zbývající veličiny w_d , w_r a w_g jsou váhy ohodnocení (váhové koeficienty ohodnocení). Konkrétně w_d je váhový koeficient ohodnocení vlivu počtu doporučení, jedná se o konstantní hodnotu z intervalu $\langle 0,1 \rangle$, w_r je váhový koeficient ohodnocení vlivu reputace, jedná se také o konstantu nabývající hodnot $\langle 0,1 \rangle$, a w_g je váhový koeficient ohodnocení vlivu dispozice k důvěře, která je též konstantního charakteru z intervalu $\langle 0,1 \rangle$.

Výsledná rovnice pro výpočet aktuální velikosti míry důvěry T_{ij} má tvar:

$$T_{ij} = t_{ij} + \sqrt{t_{ij} * t_{ji}} * \Delta d_{ij} * w_{d_i} * r_i * w_{r_i} * g_i * w_{g_i}, \quad (4.4.8)$$

kteřou ještě můžeme upravit do tvaru:

$$T_{ij} = t_{ij} + \sqrt{t_{ij} * t_{ji}} * \left(\lambda * \frac{(\sum_0^n t_{ik_1} * t_{k_1 k_2} * \dots * t_{k_{m-1} k_m} * t_{k_m j})^l - d_{ij}^{l-1}}{n} \right) * w_{d_i} * \left(\frac{\sum_0^n r_i * r_m}{n} \right) * w_{r_i} * g_i * w_{g_i}. \quad (4.4.9)$$

5 Realizace softwarového systému

V této kapitole bude popsán návrh a realizace softwarového systému, který s využitím multiagentního systému JADE popisuje problematiku modelování důvěry v sociální síti. Při implementaci aplikace je přihlédnuto k dříve popsaným metodám a je využit model z kapitoly 4.4.

5.1 Analýza aplikace

Aplikace je realizována v programovacím jazyce Java jako práce navazující na oborový projekt, jenž byl také v Javě zadán, a s využitím javovské knihovny JADE pro vytvoření multiagentního systému. Software splňuje následující body:

- Obsahuje grafické rozhraní pro snadnější a příjemnější ovládání aplikace.
- Načítá data z textového souboru nebo generuje vlastní data.
- Počítá důvěru vznikající mezi jednotlivými aktéry (= agenty) v jednotlivých iteracích na základě zadaných dat.
- Vypisuje výsledky na obrazovku, ukládá je do textového souboru a do XML souboru.

5.2 Architektura

Softwarový systém je rozdělen do několika menších částí, tak aby spolu byly vždy celky, které spolu spolupracují. Takováto architektura má výhodou v tom, že každou z těchto částí můžeme libovolně měnit bez ohledu na úpravy v blocích ostatních a činíme tak celou aplikaci nezávislou a přehlednou.

Aplikace je rozdělena do osmi balíčků splňující známou konvenci vycházející z názvu domény. Doména je složená z jednotlivých prvků od nejvyššího až po nejnižší řád. Přehled všech balíčků je přehledně zobrazen tabulce 8.

Tabulka 8 - Struktura balíků aplikace

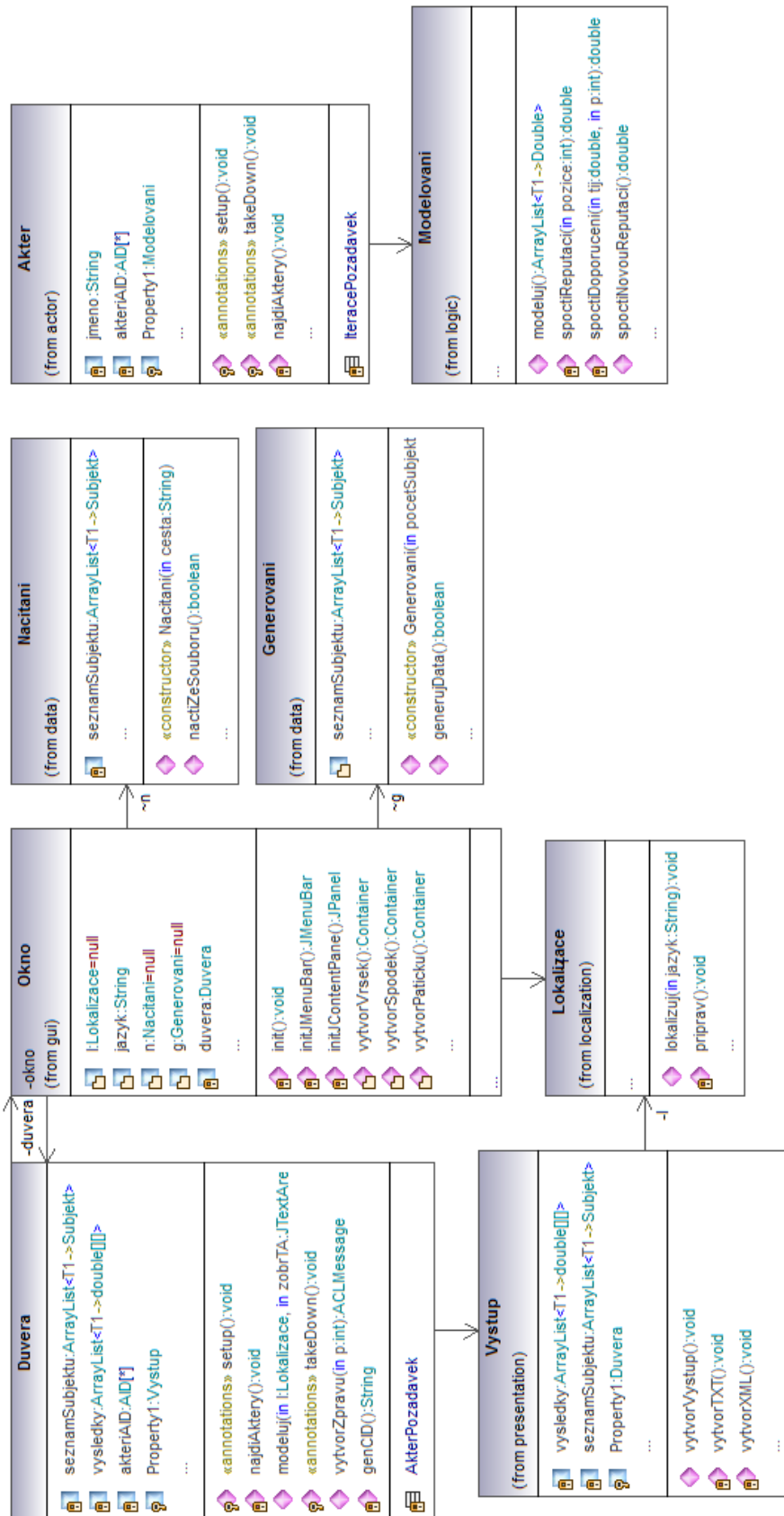
Název balíku	Popis balíku
cz.zcu.kiv.trust	Obsahuje třídu Duvera spouštějící GUI a agenta koordinátora, který řídí modelování důvěry.
cz.zcu.kiv.actor	Obsahuje třídu Akter prezentující entitu v síti agentů při modelování důvěry.
cz.zcu.kiv.trust.data	Obsahuje třídy Generovani, Nacitani a Subjekt, které slouží k získání a uložení dat do objektů, ze kterých se síť skládá.
cz.zcu.kiv.trust.gui	Obsahuje třídu Okno, která zobrazuje uživatelské rozhraní a obsluhuje ho.
cz.zcu.kiv.trust.localization	Obsahuje třídu Lokalizace. Ta slouží k lokalizaci do českého nebo anglického jazyka.
cz.zcu.kiv.trust.logic	Obsahuje třídu Modelovani, která zajišťuje hlavní logiku s výpočty pro každého aktéra.
cz.zcu.kiv.trust.presentation	Obsahuje třídu Vystup. Tato třída vypisuje data na obrazovku, do textového souboru a do XML souboru.
cz.zcu.kiv.trust.util	Obsahuje třídu Utility, kde se nachází obslužné metody usnadňující často se opakující práci.

5.2.1 UML diagram tříd

Pro lepší představu, které třídy spolu navzájem kooperují a co s čím jak souvisí, je na obrázku 18 zobrazen UML diagram tříd aplikace. Pro nedostatek místa jsou v něm zakresleny jen ty nejdůležitější části.

Celá aplikace je složena z deseti tříd, z nichž každá má svou specifickou funkci. Dále program obsahuje dalších několik tříd pro obsluhu tlačítek, služby uživatelského prostředí anebo chování agentů a jejich požadavků.

Celý systém tvoří dva typy agentů. První z nich je jeden jediný agent koordinátor, přes kterého uživatel komunikuje se systémem a ovládá ho. Druhým typem jsou agenti aktéři, což jsou samotné subjekty v sociální síti.



Obrázek 18 - Zjednodušený UML diagram tříd

5.2.2 Agent koordinátor

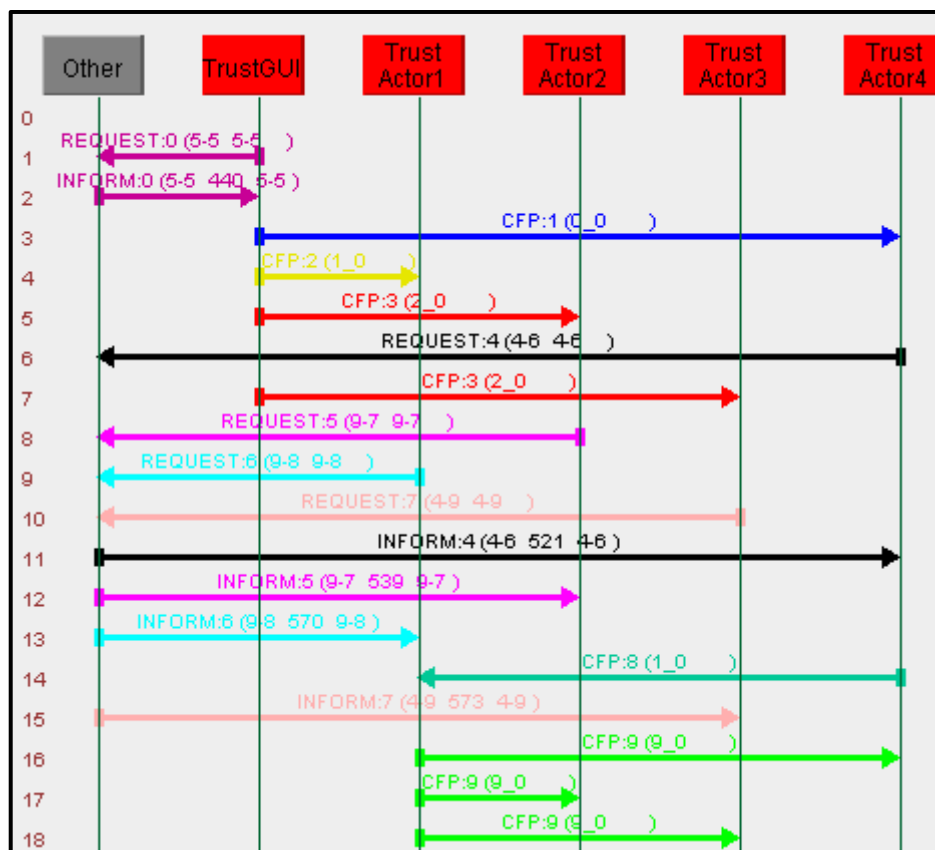
Agent koordinátor obsahuje třídy Duvera, Generovani, Nacitani, Subjekt, Okno, Lokalizace, Vystup a Utility. Jeho hlavním úkolem je zprostředkovat uživateli grafické rozhraní. V tomto rozhraní uživatel vybere způsob načtení dat a spustí modelování. Agent koordinátor získaná data rozešle každému z agentů aktérů a oni mu posílají zpátky výsledky modelování důvěry. Tento cyklus se opakuje, dokud se nespočítá důvěra pro všechny iterace. Když agent koordinátor získá všechny výsledky, vypíše je na obrazovku a uloží do textového a XML souboru.

5.2.3 Agent aktér

Agent aktér využívá ke své činnosti třídy Akter, Subjekt, Modelovani a Utility. Každý agent zastupuje právě jeden subjekt v sociální síti. Jeho činnost začíná registrací do systému a poté čeká na příjem zprávy od agenta koordinátora. Zpráva obsahuje všechny údaje o konkrétním agentovi. Agent aktér data postupně zpracovává a komunikuje při tom s ostatními agenty aktéry, se kterými si vyměňuje všechny chybějící informace. Pokud získá všechny potřebné hodnoty, spočítá aktuální důvěru a vrátí výsledky agentu koordinátorovi.

5.2.4 Komunikace agenta koordinátora a agenta aktéra

Ukázka komunikace mezi agentem aktérem a agentem koordinátorem v jednotlivých iteracích je vidět na obrázku 19. Průběh komunikace je sledovaný pomocí Sniffer agenta poskytovaného knihovnou JADE. Celá komunikace je potom přiložena na CD přiložené k této práci.



Obrázek 19 - Komunikace mezi agenty

Význam jednotlivých zpráv z obrázku 19 je popsán v tabulce 9, kdy barvy šipek znamenají zprávu se stejným identifikačním číslem.

Tabulka 9 - Popis komunikace mezi agenty

Pořadí zprávy	Performativ	Odesílatel	Příjemce	Význam
1.	REQUEST	GUI	DF	Vyhledání agentů aktérů
2.	INFORM	DF	GUI	Seznam agentů aktérů
3.	CFP	GUI	Aktér 4	Data pro aktéra
4.	CFP	GUI	Aktér 1	Data pro aktéra
5.	CFP	GUI	Aktér 2	Data pro aktéra
6.	REQUEST	Aktér 4	DF	Vyhledání agentů aktérů

7.	CFP	GUI	Aktér 3	Data pro aktéra
8.	REQUEST	Aktér 2	DF	Vyhledání agentů aktérů
9.	REQUEST	Aktér 1	DF	Vyhledání agentů aktérů
10.	REQUEST	Aktér 3	DF	Vyhledání agentů aktérů
11.	INFORM	DF	Aktér 4	Seznam agentů aktérů
12.	INFORM	DF	Aktér 2	Seznam agentů aktérů
13.	INFORM	DF	Aktér 1	Seznam agentů aktérů
14.	CFP	Aktér 4	Aktér 1	Data pro aktéra
15.	INFORM	DF	Aktér 3	Seznam agentů aktérů
16.	CFP	Aktér 1	Aktér 4	Data pro aktéra
17.	CFP	Aktér 1	Aktér 3	Data pro aktéra
18.	CFP	Aktér 1	Aktér 2	Data pro aktéra

5.3 Implementace

Nyní se budeme zabývat kompletním popisem implementace softwarového systému pro modelování důvěry v sociální síti.

5.3.1 Třída Duvera

Třída Duvera je hlavní třídou reprezentující agenta koordinátora. Dědí z nadřazené třídy jade.core.Agent knihovny JADE. Překrývá dvě metody rodičovské třídy, kterými jsou setup() a takeDown().

Metoda setup() je startovací metoda agenta a její tělo je prováděno hned při startu. Setup() spouští grafické uživatelské rozhraní reprezentované třídou Okno. Defaultním jazykem je čeština.

Metoda takeDown() má na starosti ukončení agenta a vypnutí grafického uživatelského rozhraní.

Třída `Duvera` obsahuje i další metody. Jednou z nich je veřejná metoda `modeluj(Lokalizace l, ...)`, která je volána po stisknutí tlačítka `Modeluj` grafického uživatelského rozhraní. V ní jsou nejprve pomocí metody `najdiAktery()` nalezeni agenti aktéři, kterým jsou později v cyklu rozeslána data ke zpracování. Zpráva pro aktéry je vytvořena pomocí metody `vytvorZpravu()` a identifikační číslo zprávy je generováno metodou `genCID()` převzaté z `www.iro.umontreal.ca`. Pro rozesílání a komunikaci s agentem aktérem je vytvořeno speciální chování `AkterPozadavak(this, zprava)`. Toto kompozitní chování je odvozeno od rodičovské třídy `SequentialBehaviour`. Ve zděděné `onStart()` metodě je odeslána zpráva a připraveno chování pro přijetí odpovědi s výsledky. Výsledky se po příjmu uloží a chování se ukončí. Modelování tímto způsobem probíhá ve všech iteracích, dokud nejsou vymodelovány všechny iterace. Nakonec je vytvořena třída `Vystup`, kam se výsledky předají pro prezentaci.

Tabulka 10 - Metody a atributy třídy `Duvera.java`

Duvera.java	
Odděděná od	<code>jade.core.Agent</code>
Metody	<code>protected void setup()</code> - vytvoří instanci třídy <code>Okno</code> a spustí grafické rozhraní
	<code>private void najdiAktery()</code> - najde všechny zaregistrované agenty aktéry a uloží si jejich adresy
	<code>public void modeluj(Lokalizace l, JTextArea zobrTA, ArrayList<Subjekt> seznamSubjektu, double[][] maticeDuvery, ArrayList<double[][]> maticeReputaci, int[][] sitAkteru)</code> - spouští se po stisku tlačítka <code>Modeluj</code> a odesílá a přijímá zprávy s daty se všemi agenty aktéry
	<code>protected void takeDown()</code> - odregistruje agenta koordinátora po ukončení uživatelského rozhraní
	<code>private ACLMessage vytvorZpravu(int p)</code> - vytvoří ACL zprávu
	<code>private String genCID()</code> - vygeneruje unikátní identifikátor zprávy
Atributy	<code>private static final long serialVersionUID</code> - sériové označení třídy
	<code>private Okno okno</code> - instance třídy <code>okno</code>
	<code>private ArrayList<double[][]> vysledky</code> - list s matice aktuální důvěry v iteracích
	<code>private double[][] aktualniDuvera</code> - matice aktuální důvěry pro aktuální iteraci

	private int aktualniIterace - označení aktuální iterace
	private AID[] aktériAID - seznam aktérů
	private int prijatych - počet přijatých odpovědí od aktérů pro danou iteraci

5.3.2 Třída Okno

Celé grafické rozhraní aplikace je implementováno v této třídě. Ke svému správnému chodu využívá balíků `java.awt` a `javax.swing`. Třída `Okno` dědí z nadřazené třídy `javax.swing.JFrame`.

Při zavolání konstruktoru s parametrem jazyku se jako první načte lokalizace celého rozhraní a poté se inicializuje uživatelské prostředí (obrázek GUI je přiložen v přílohách této práce). Celé prostředí je složeno z menu a pole rozděleného do tří částí.

Menu obsahuje dvě položky `Program` a `Nápověda`. Každá z položek je dále rozvíitelná a najdeme v nich tlačítka pro vyčištění textového pole s výsledky, nastavení jazyku, ukončení aplikace či informace o programu. Každé z těchto tlačítek má svoji klávesovou zkratku:

- Vyčistit konzoli – `CTRL+C`
- Změna jazyka – `CTRL+L`
- Ukončení aplikace – `CTRL+E`
- O programu – `CTRL+A`

Hlavní panel okna je rozdělen do tří částí a využívá `BorderLayout`. První vrchní panel obsahuje `JRadioButton` s možností výběru načtení dat ze souboru nebo náhodného generování. V případě načtení dat z textového souboru je využit `FileChooser` a `FileFilter`, které otevřou nové okno s průzkumníkem a filtrem na textový soubor. V opačném případě generování musíme zadat počet subjektů a počet iterací, pro které se mají data generovat. Počet subjektů v takovém případě musí odpovídat počtu spuštěných agentů aktérů.

Ve druhé části okna nalezneme textové pole, do kterého se vypisují všechny důležité zprávy a výsledky. Pro lepší přehlednost a možnost prohlídky starších zpráv obsahuje toto textové pole posuvník. V poslední třetí části se jsou umístěna tlačítka pro vyčištění konzole nebo ukončení aplikace.

Každé z tlačítek pro svou funkci implementuje vnitřní třídu dědicí od `AbstractAction` a obsahuje konstruktor třídy. V konstruktoru je přidána každé akci klávesová zkratka, jak bylo zmíněno výše. Dále vnitřní třída obsahuje metodu `actionPerformed()` zajišťující funkci tlačítka. Nejsložitější funkčnost se skrývá pod tlačítkem `Modeluj`. V obsluhující metodě se vytvoří instance třídy `Nacitani` nebo `Generovani` podle zvoleného způsobu a načte se seznam všech aktérů, matice sítě aktérů, matice počáteční důvěry a seznam matic reputací. Poté se volá metoda `modeluj()` ze třídy `Duvera` agenta koordinátora.

Tabulka 11 - Metody a atributy třídy `Okno.java`

Okno.java	
Odděděná od	<code>javax.swing.JFrame</code>
	<code>public Okno(String jazyk, Duvera duvera)</code> - konstruktor třídy s lokalizací
	<code>private void init()</code> - inicializace grafického rozhraní
	<code>private JMenuBar initJMenuBar()</code> - vytvoření menu grafického rozhraní
	<code>private JPanel initJContentPane()</code> - vytvoření vnitřního panelu grafického rozhraní
	<code>Container vytvorVrsek()</code> - vytvoření kontejneru s volbou načtení dat a tlačítkem pro spuštění modelování
	<code>Container vytvorSpodek()</code> - vytvoření kontejneru s textovým polem pro výpis výsledků a svislým posuvníkem
	<code>Container vytvorPaticku()</code> - vytvoření kontejneru s tlačítkem pro vyčištění textového pole a ukončení aplikace
	<code>private Component nastav(Component c, int x, int y, int s, int v, double rs, double rv, int vyp, int k)</code> - metoda pro umístění komponenty v grafickém uživatelském rozhraní
	<code>private void vytvorAkce()</code> - vytvoří akce pro jednotlivá tlačítka
Metody	<code>private void pridelAkce()</code> - přidělí akce jednotlivým tlačítkům
	<code>private final String getCestaSouboru()</code> - vrátí cestu k načítanému textovému souboru
	<code>private final int getPocetSubjektu()</code> - vrátí počet subjektů při generování vstupních dat
	<code>private final int getPocetIteraci()</code> - vrátí počet iterací při generování náhodných dat

Atributy	private static final long serialVersionUID - sériové označení třídy
	private static final int SIRKA, VYSKA - konstanty pro doporučenou šířku a výšku okna rozhraní
	private static final int NOR, SOU, WES, EAS, BOT, NON, CEN, HOR - konstanty pro GridBagConstraints
	private GridBagConstraints gbl - rozvržení grafického rozhraní
	private GridBagConstraints - rozvržení grafického rozhraní
	private JPanel vrsekPN, spodekPN, patickaPN - panely grafického rozhraní
	private JRadioButton zeSouboruRBT, generovatNahodneRBT - rádiová tlačítka grafického rozhraní
	private JLabel cestaLB, JLabel generovaniLB, pocetSubjektuLB, pocetIteraciLB - textová návěští grafického rozhraní
	private JTextField poleCestaTF, pocetSubjektuTF, pocetIteraciTF - textová pole grafického rozhraní
	private JButton zdatCestuBTN, modelujBTN, vycistiBTN, konecBTN - tlačítka grafického rozhraní
	private static JTextArea zobrTA - textové pole pro výpis výsledků grafického rozhraní
	private String jazyk - řetězec s jazykem grafického rozhraní

5.3.3 Třída Lokalizace

Třída Lokalizace má jediný úkol a to převést grafické uživatelské rozhraní do jednoho ze dvou jazyků – čeština, angličtina. Jazyk pevně určuje Locale a využívá ResourceBundle. Přes ResourceBundle jsou načteny lokalizační soubory a jednotlivé výrazy jsou uloženy do hash mapy, ze které jsou dále vybírány. Vytvoření instance této třídy probíhá ještě před vykreslením grafického rozhraní a při každé změně jazyka dochází k novému vykreslení.

Tabulka 12 - Metody a atributy třídy Loklizace.java

Lokalizace.java	
Odděděná od	---
Metody	public Lokalizace(String jazyk) - konstruktor třídy s parametrem jazyku
	public void lokalizuj(String jazyk) - načte lokalizační soubor podle zvoleného jazyka

Atributy	private void priprav() - připravý hash mapu s texty dle zvoleného jazyka
	public String getVypis(int klic) - vrátí text dle pořadového čísla v hash mapě
	private Locale lokalizace - Locale určující jazyk
	private ResourceBundle lokalizacniSoubor - ResourceBundle s přiloženým souborem dle zvoleného jazyka
	private HashMap<Integer, String> textyRozhrani - hash mapa s texty dle zvoleného jazyka

5.3.4 Třída Nacitani

Instance třídy Nacitani je vytvořeno při volbě načítání dat z textového souboru. Přímou v konstruktoru metody je parametr s cestou k souboru. Textový soubor má přesně stanovenou strukturu, jinak je uživateli na obrazovku vypsané chybové hlášení. Struktura textového souboru je zobrazena ve sloupcích pro úsporu místa na obrázku 20. K zpracování souboru je naimplementována metoda nactiZeSouboru() využívající BufferedReader a FileReader z knihovny java.io.*. Data jsou při načítání rovnou upravena a uložena pro pozdější použití v privátních proměnných třídy. K proměnným se dostaneme přes get() metody.

```

subjects:      trust:      receiver: E      0.5 - - 0.9 -
A              - 0.7 0.3 - -   D-C-A           0.8 - - 0.9 -
0.7            0.8 - - 0.6 -   S               - 0.7 0.7 - 0.6
0.6 0.6 0.6   0.4 - - 0.7 -   receiver: A     - - - 0.8 -
               - 0.5 0.8 - 0.6   B-D C-D C-D-E   4
B              - - - 0.7 -   receiver: B     - 0.6 0.4 - -
0.5            0.8 - - 0.9 -   D-E            0.5 - - 0.9 -
0.6 0.6 0.6   recommendation: 0.8 - - 0.9 -
C              0              reputation:      - 0.7 0.7 - 0.6
0.6            receiver: C    0              - - - 0.8 -
0.6            D-E           - 0.6 0.4 - -   5
0.6 0.6 0.6   1              0.5 - - 0.9 -   - 0.6 0.4 - -
               receiver: A    0.8 - - 0.9 -   0.5 - - 0.9 -
D              B-D-E C-D-E   - 0.7 0.7 - 0.6 0.8 - - 0.9 -
0.8            receiver: E   - - - 0.8 -     - 0.7 0.7 - 0.6
0.6 0.6 0.6   D-B-A         1              - - - 0.8 -
               2              - 0.6 0.4 - -
E              receiver: B   0.5 - - 0.9 -   state:
0.4            D-E           0.8 - - 0.9 -   0.8 0.7 0.8 0.4 0.5 0.7
0.6 0.6 0.6   receiver: C   - 0.7 0.7 - 0.6 0.4 0.5 0.6 0.1 0.1 0.1
               D-E           - - - 0.8 -     0.5 0.5 0.5 0.5 0.5 0.5
network:      3              2              0.9 0.6 0.3 0.2 0.1 0.1
- 1 1 0 0     receiver: B   - 0.6 0.4 - -   0.1 0.2 0.3 0.1 0.1 0.1
1 - 0 1 0     D-E           0.5 - - 0.9 -
1 0 - 1 0     receiver: C    0.8 - - 0.9 -
0 1 1 - 1     D-E           - 0.7 0.7 - 0.6
0 0 0 1 -     4              - - - 0.8 -
               receiver: C    3
               D-E           - 0.6 0.4 - -
    
```

Obrázek 20 - Struktura textového souboru ve sloupcích

Tabulka 13 - Metody a atributy třídy Nacitani.java

Nacitani.java	
Odděděná od	---
Metody	public Nacitani(String cesta) - konstruktor třídy s parametrem cesty k textovému souboru
	public boolean nactiZeSouboru() - načítá a zpracuje textový soubor, vrací true v případě úspěchu, jinak false
	private double[][] parseMatice(int id, String radka, String[] pomocne) - parseruje matici hodnot double z textového souboru
	public ArrayList<Subjekt> getSeznamSubjektu() - vrací seznam načtených subjektů
	public int[][] getSitAkteru() - vrací matici aktérů sítě
	public double[][] getMaticiDuvery() - vrací matici počáteční důvěry
	public ArrayList<double[][]> getMaticeReputaci() - vrací seznam matic reputací v jednotlivých iteracích
	private String cestaSouboru - cesta k souboru, který se bude načítat
	private BufferedReader bfr - BufferedReader pro načtení textového souboru
	private ArrayList<Subjekt> seznamSubjektu - seznam načtených subjektů
Atributy	private int[][] sitAkteru - matice udávající sousedy subjektů v síti
	private double[][] maticeDuvery - matice počáteční důvěry
	private ArrayList<double[][]> maticeReputaci - seznam matice reputací v jednotlivých iteracích
	private ArrayList<Double> poleDispozice - seznam dispozic subjektu

5.3.5 Třída Generovani

Funkcí třídy je generování pseudonáhodných dat, která jsou potom využita pro modelování důvěry v jednotlivých iteracích. Při uživatelem zvolené variantě generování dat je volána metoda generujData(), kdy systému stačí znát počet subjektů v síti a počet iterací, po které se má modelovat. Metoda generujData() využívá dalších šesti privátních metod generujSubjekty(), generujSitAkteru(), generujMaticiDuvery(), generujMaticiReputace(), generujPoleDispozice() a konečně generujSeznamDoporuceni(). Každá z metod dle svého názvu srozumitelně říká, co má za úkol. Při generování konkrétních hodnot jsou využity

metody generujInt() a generujDoubleGauss(). Jak název druhé zmíněné metody napovídá, vygenerované hodnoty typu double jsou dle Gaussovského normálového rozdělení, aby byla data blízká skutečnosti a nejvíce hodnot bylo ze středu zvoleného intervalu. Využívá se metoda nextGaussian() z balíku java.util.Random. Hodnoty a data jsou uložena v privátních proměnných třídy a lze se k nim dostat přes get() metody.

Tabulka 14 - Metody a atributy třídy Generovani.java

Generovani.java	
Odděděná od	---
Metody	public Generovani(int pocetSubjektu, int pocetIteraci) - konstruktor třídy s parametry počtu subjektů a počtu iterací pro modelování
	public boolean generujData() - generuje a ukládá data do proměnných, v případě úspěchu vrací true, jinak false
	private void generujSubjekty() - generuje aktéry sítě, jejich id, jméno, reputaci a váhy doporučení, dispozice a reputace
	private void generujSitAkteru() - generuje síť aktérů
	private void generujMaticiDavery() - generuje matici počáteční důvěry
	private ArrayList<String> generujSeznamDoporuceni(Subjekt s) - generuje seznam doporučení pro každý subjekt v každé iteraci
	private void generujMaticiReputace() - generuje matici reputace
	private void generujPoleDispozice() - generuje pole dispozice pro každý subjekt ve všech iteracích
	private int generujInt(int param) - generuje hodnotu integer od 0 do hodnoty stanovené parametrem
	private double generujDoubleGauss() - generuje hodnotu double dle Gaussova normálového rozdělení
	public ArrayList<Subjekt> getSeznamSubjektu() - vrací seznam načtených subjektů
	public int[][] getSitAkteru() - vrací matici aktérů sítě
	public double[][] getMaticiDavery() - vrací matici počáteční důvěry
	public ArrayList<double[][]> getMaticeReputaci() - vrací seznam matic reputací v jednotlivých iteracích
	Atributy

	private int pocetIteraci - počet iterací, po které se bude modelovat důvěra
	private ArrayList<Subjekt> seznamSubjektu - seznam načtených subjektů
	private int[][] sitAkteru - matice udávající sousedy subjektů v síti
	private double[][] maticeDuvery - matice počáteční důvěry
	private ArrayList<double[][]> maticeReputaci - seznam matic reputací v jednotlivých iteracích
	private ArrayList<Double> poleDispozice - seznam dispozic subjektu

5.3.6 Třída Subjekt

Subjekt je datová třída, která uchovává informace o každém aktérovi při modelování důvěry v síti. Aktér má vždy svůj jednoznačný identifikátor, jméno, svou reputaci, váhové koeficienty počtu doporučení, reputace a dispozice k důvěře, seznam doporučení, které obdržel, a nakonec dispozici k důvěře ve všech krocích modelování. Všechny zmíněné vlastnosti aktéra se dají nastavit přes set() metody a získat přes get() metody.

Tabulka 15 - Metody a atributy třídy Subjekt.java

Subjekt.java	
Odděděná od	---
	public Subjekt(int id, String jmeno, double reputace, double wd, double wr, double wg) - konstruktor třídy s parametry
Metody	public Subjekt() - konstruktor třídy bez parametrů
	public int getId() - vrací identifikátor aktéra
	public int setId(int id) - nastaví identifikátor aktéra
	public String getJmeno() - vrací jméno aktéra
	public void setJmeno(String jmeno) - nastaví jméno aktéra
	public double getReputace() - vrací reputaci aktéra
	public void setReputace(double reputace) - nastaví reputaci aktéra

Atributy	public double getWD() - vrací váhu doporučení aktéra
	public void setWD(double wd) - nastaví váhu reputace aktéra
	public double getWR() - nastaví váhu reputace aktéra
	public void setWR(double wr) - nastaví váhu reputace aktéra
	public double getWG() - vrací váhu dispozice k důvěře aktéra
	public void setWG(double wg) nastaví váhu dispozice k důvěře aktéra
	public HashMap<Integer, ArrayList<String>> getSeznamDoporuceni() - vrací seznam doporučení aktéra v jednotlivých iteracích
	public void setSeznamDoporuceni(int iterace, ArrayList<String> sd) - nastaví seznam doporučení aktéra v jednotlivých iteracích
	public ArrayList<Double> getPoleDispozice() - vrací seznam dispozice k důvěře aktéra v jednotlivých iteracích
	public void setPoleDispozice(ArrayList<Double> pd) - nastaví seznam dispozice k důvěře aktéra v jednotlivých iteracích
	private int id - unikátní identifikátor aktéra
	private String jmeno - jméno aktéra
	private ArrayList<Double> reputace - seznam reputací aktéra v jednotlivých iteracích
	private double wd - váha doporučení aktéra
	private double wr - váha reputace aktéra
	private double wg - váha dispozice k důvěře aktéra
private HashMap<Integer, ArrayList<String>> seznamDoporuceni - seznam doporučení aktéra v jednotlivých iteracích	
private ArrayList<Double> poleDispozice - seznam dispozice k důvěře aktéra v jednotlivých iteracích	

5.3.7 Třída Akter

Tato třída zastupuje agenta aktéra a spouští se nezávisle na dříve popsaných třídách. Protože je třída Akter odděděná od jade.core.Agent má implementovány metody setup() zajišťující běh agenta a takeDown() volané při odhlášení a ukončování aktéra. Život agenta aktéra spočívá v přihlášení se do multiagentního systému a čekání na přijetí zprávy od koordinátora. K tomuto účelu obsahuje Akter cyklické chování pro obsluhu

zpráv, kdy u každé příchozí zprávy rozhoduje, zda přišla od koordinátora nebo jiného aktéra. Odesílatel první přijaté zprávy je koordinátor s daty, se kterými agent aktér dále pracuje, aby spočítal aktuální důvěru ke všem známým aktérům. Z tohoto důvodu komunikuje s ostatními aktéry a získává i další chybějící data. Pokud má agent zpracované všechny zprávy a uložené vše potřebné, proběhne výpočet důvěry a výsledky jsou odeslány zpět agentu koordinátorovi. Chování se cyklicky opakuje pro všechny iterace.

Správnou činnost třídy Akter zajišťují metody `parseZpravaOdGui()` a `parseZpravaOdAktera()`. Jak již názvy metod napovídají, je jejich úkolem zpracovávat přijaté zprávy od koordinátora a aktéra. Aby mohl aktér komunikovat i s ostatními agenty v systému, využívá pro jejich nalezení metodu `najdiAktery()`.

Tabulka 16 - Metody a atributy třídy Akter.java

Akter.java	
Odděděná od	jade.core.Agent
Metody	protected void setup() - zajišťuje činnost agenta po jeho spuštění – registrace do systému, příjem a obsluhu zpráv
	protected void takeDown() - odregistruje agenta ze systému a ukončí ho
	private void parseZpravaOdGui(String zprava) - zpracuje zprávu od agenta koordinátora
	private ArrayList<String> parseZpravaOdAktera(String zprava) - zpracuje zprávu od agenta aktéra
	private void ulozZpravu(ArrayList<String> al) - uloží data ze zprávy od jiného agenta aktéra do hash mapy, jsou v ní uloženy všechny zprávy od všech aktérů
	private void najdiAktery() - nalezne ostatní agenty aktéry v síti
	private static final long serialVersionUID - sériové označení třídy
Atributy	private int id - unikátní identifikátor aktéra
	private double ri - reputace aktéra
	private String jmeno - jméno aktéra
	private double wd - váha doporučení aktéra
	private double wr

- váha reputace aktéra
private double wg
- váha dispozice k důvěře aktéra
private ArrayList<Double> duvera
- seznam důvěry k ostatním aktérům
private ArrayList<Double> reputace
- seznam reputace ostatních aktérů
private ArrayList<String> doporuceni
- seznam doporučení aktéra na ostatní aktéry
private ArrayList<Integer> sousede
- seznam sousedů aktéra
private ACLMessage zpravaGUI
- zpráva s výsledky pro agenta koordinátora
private AID[] akteriAID
- seznam ostatních aktérů v systému
private int pocetAkteru
- počet aktérů v systému
private int pocetPrijatychZprav
- počet přijatých zpráv od aktérů
private HashMap<Integer, ArrayList<String>> prijateZpravy
- přijaté zprávy od ostatních agentů

5.3.8 Třída Modelovani

Po vytvoření instance třídy Modelovani se v konstruktoru inicializují všechny důležité proměnné a seznamy, které jsou třeba při modelování důvěry, tzn. data získaná při načtení ze souboru či náhodně vygenerovaná.

Třída obsahuje jen jednu veřejnou metodu modeluj(). Po jejím zavolání se v jednoduchém cyklu spočítá aktuální důvěra mezi aktérem, který metodu zavolal a všemi známými prvky v síti. Pro spočtení veličin jako je doporučení a reputace jsou v programu implementovány metody spoctiReputaci() a spoctiDoporuceni(), které vrací hodnotu s vypočtenou reputací či doporučením. V metodě modeluj() je nutné hlídat, možnost existence doporučení na prvek, který aktér zatím nezná. Dále také okrajové hodnoty důvěry rovnající se nule a ukládány jako jednotka sto-biliardtina a stejně tak i hodnoty rovny či větší než jedna. Vypočtená aktuální důvěry se ukládá do listu, který metoda vrací.

Tabulka 17 - Metody a atributy třídy Modelovani.java

Modelovani.java	
Odděněná od	---
Metody	public Modelovani(int id, String jmeno, double wd, double wr, double wg, double dispozice, ArrayList<Double> duvera, ArrayList<Double> reputace, ArrayList<String> doporuzeni, ArrayList<Integer> sousede, HashMap<Integer, ArrayList<String>> data) - konstruktor třídy
	public ArrayList<Double> modeluj() - vypočte aktuální důvěru aktéra ke všem ostatním známým aktérům v síti
	private double getDuvera(int i) - vrací důvěru aktéra podle identifikátoru
	private double spoctiReputaci(int pozice) - spočte velikost reputace prvku podle jeho pozice
	private double spoctiDoporuzeni(double tij, int p) - spočte velikost doporučení podle pozice a cesty
	private static double[] LAMBDA - konstanta pole koeficientu doporučení
	private int id - unikátní identifikátor aktéra
	private String jmeno - jméno aktéra
	private double wd - váha doporučení aktéra
	private double wr - váha reputace aktéra
Atributy	private double wg - váha dispozice k důvěře aktéra
	private ArrayList<Double> duvera - seznam důvěry k ostatním aktérům
	private ArrayList<Double> reputace - seznam reputace ostatních aktérů
	private ArrayList<String> doporuzeni - seznam doporučení aktéra na ostatní aktéry
	private HashMap<Integer, ArrayList<String>> data - seznam zpráv od ostatních aktérů dle jejich identifikátoru

5.3.9 Třída Vystup

Činností třídy Vystup je vytvořit prezentaci výsledků a to na obrazovku, do textového souboru a do XML souboru. Třída obsahuje jednu veřejnou metodu vytvorVystup(), která volá dvě metody privátní – vytvorTXT() a vytvorXML() – a nakonec vypíše výsledky na obrazovku.

Metoda vytvorTXT() vytvoří textový soubor, do kterého vypíše zpracovaná data a výsledky. Pro zápis do souboru je využit PrintWriter, BufferedWriter, OutputStreamWriter a FileOutputStreamWriter z knihovny java.io.*. Je použito kódování UTF-8 a výstupní soubor se nazývá vystup.txt.

Metoda vytvorXML() dělá to stejné, jen se liší typ výstupního souboru, kterým je soubor s příponou XML. Pro práci je využita technologie StAX, což nám umožňuje rychlý sekvenční nízko-úrovňový zápis velkých XML dokumentů. Výstupním souborem je vystup.xml.

Tabulka 18 - Metody a atributy třídy Vystup.java

Vystup.java	
Odděděná od	---
Metody	public Vystup(Lokalizace l, JTextArea zobrTA, ArrayList<double[][]> vysledky, ArrayList<Subjekt> seznamSubjektu, double[][] maticeDuvery, ArrayList<double[][]> maticeReputaci, int[][] sitAkteru) - konstruktor třídy se všemi daty
	public void vytvorVystup() - veřejná metoda pro vytvoření výstupu
	private void vypis(double vysl) - upravuje výpis do definované podoby
	private void vytvorTXT() - vytvoří textový soubor s výsledky
	private void vypisTXT(Printwriter pw, double vysl) - upravuje výpis textového souboru do definované podoby
	private void vytvorXML() - vytvoří XML soubor
	Atributy
private JTextArea zobrTA - textové pole grafického rozhraní	
private ArrayList<double[][]> vysledky - seznam s matice důvěry v jednotlivých iteracích	
private ArrayList<Subjekt> seznamSubjektu - seznam všech subjektů	
private double[][] maticeDuvery - matice počáteční důvěry	
private ArrayList<double[][]> maticeReputaci - seznam matic reputací v jednotlivých iteracích	
private int[][] sitAkteru - matice udávající sousedy subjektů v síti	

5.3.10 Třída Util

Pomocná třída s často používanými funkcemi k usnadnění práce. Konkrétně se jedná o funkce na převod řetězec do hodnoty integer nebo double – `stringToInt()` nebo `stringToDouble()` – na základě přijatého řetězce.

Tabulka 19 - Metody a atributy třídy Util.java

Akter.java	
Odděděná od	---
Metody	<code>public static int stringToInt(String s)</code> - převede řetězec do hodnoty integer
	<code>public static double stringToDouble(String s)</code> - převede řetězec do hodnoty double
Atributy	---

6 Experimenty s modelem důvěry v sociální síti

V této kapitole si ukážeme experiment rozdělený do tří částí s modelem důvěry v sociální síti. Nejprve si popíšeme získání dat na konkrétně zvolené síti Facebook popsané v kapitole 3.4.1 a poté se zaměříme na práci s daty v modelu důvěry z kapitoly 4.4, k čemuž je využit softwarový systém z kapitoly 5.

6.1 Zisk reálných dat

Jak bylo uvedeno výše, data byla získávána na sociální síti Facebook od října roku 2011 do dubna roku 2012 ve vytvořené skupině s několika účastníky. Jejich počet se postupně zvyšoval, a proto je experiment rozdělen do třech částí. Prvních sedm týdnů byla skupina čtyřčlenná, dalších sedm týdnů sedmičlenná a posledních sedm týdnů dvanáctičlenná. Každý účastník se vstupem do této skupiny odevzdal uvítací dotazník, ve kterém odpověděl na otázky, díky nimž se určily hodnoty váhy doporučení, reputace a dispozice k důvěře. Dále každý odpověděl na základní otázky ohledně dalších účastníků, zdali se znají nebo ne a jak moc a jak druhého účastníka hodnotí. Ukázkový dotazník je vložen jako příloha na konci této práce. Všechny vyplněné dotazníky jsou potom k nalezení na přiloženém CD.

Účastníci měli za úkol ve vytvořené skupině komunikovat na různá témata a více se poznávat. Každý týden, který byl zvolen jako iterace, potom odevzdali dotazník s hodnocením ostatních a své momentální nálady. Díky tomu byly získávány hodnoty reputace a nálady subjektu ve všech iteracích. Toto se opakovalo po sedm iterací, kdy někteří z účastníků přivedli do skupiny svého známého, kamaráda nebo zajímavou osobnost a napsali na něj krátké doporučení pro ostatní. Na základě doporučení si mohli ostatní utvořit názor na nově příchozí. Všichni nováčci stejně jako jejich předchůdci vyplnili vstupní dotazník a vše se opakovalo dalších sedm týdnů. Stejně tak probíhala i poslední třetí část experimentu s konečnými dvanácti aktéry. Poslední dotazník navíc obsahoval od každého zhodnocení všech ostatních přispěvatelů za celou dobu strávenou ve skupině.

Získaná data byla použita pro zkoumání důvěry v modelu důvěry v sociální síti a konečné výsledky byly porovnány s celkovým hodnocením účastníků skupiny.

Celý experiment byl samotnými přispěvateli hodnocen velmi pozitivně a komunikace v této skupině pokračuje i po jejím oficiálním ukončení.

6.2 První část experimentu – čtyřčlenná skupina

6.2.1 Základní hodnoty

Experiment obsahuje čtyři aktéry A, B, C a D, kteří jsou zachyceni v matici sousednosti S . V této matici patří každému prvku A, B, C a D jeden řádek a jeden sloupec, kdy jednička značí vztah mezi dvěma subjekty získaný někdy dříve.

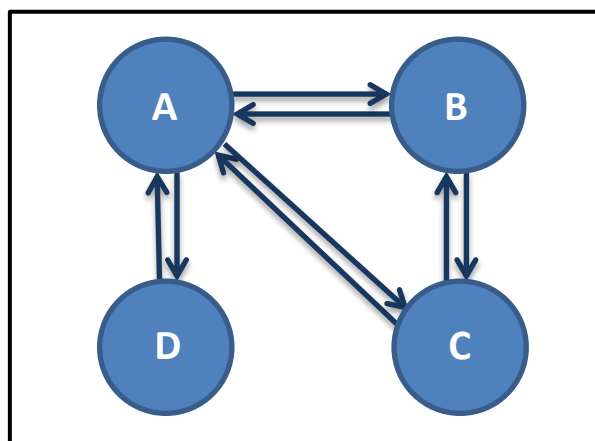
$$S = \begin{pmatrix} - & 1 & 1 & 1 \\ 1 & - & 1 & 0 \\ 1 & 1 & - & 0 \\ 1 & 0 & 0 & - \end{pmatrix}$$

Spojení jména prvku s řádkem a sloupcem matice je pro větší přehlednost zachyceno v následující tabulce 20.

Tabulka 20 - Prvky sítě a jejich pozice v matici v první části experimentu

Prvek	A	B	C	D
Pozice prvku	0	1	2	3

Grafický pohled na síť vidíme na obrázku 21. Informace v něm uvedené přesně odpovídají hodnotám v matici sousednosti S .



Obrázek 21 - Sociální síť aktérů v první části experimentu

Jak je na první pohled zřejmé nejvíce kontaktů má prvek A. Nazýváme ho tedy centrálním prvem sítě. Bez tohoto spojovacího prvku by daná síť nemohla vzniknout,

protože prvek D by se nedostal do kontaktu s prvky B a C. Z toho i vyplývá, že nejvíce izolovaným prvkem je právě aktér D, který se zná pouze s centrálním prvkem A.

Každý z prvků A, B, C a D má své váhy doporučení, reputace a dispozice k důvěře. Tyto hodnoty jsou znázorněny v tabulce 21 a po celou dobu experimentu se nemění.

Tabulka 21 - Váha doporučení, reputace a dispozice každého aktéra sítě v první části experimentu

Prvek	A	B	C	D
Váha doporučení	0.6	0.4	0.4	0.4
Váha reputace	0.2	0.4	0.6	0.4
Váha dispozice k důvěře	0.5	0.7	0.6	0.6

Protože se v sociální skupině bude bavit každý s každým je matice počáteční důvěry T dána takto:

$$T = \begin{pmatrix} - & 0.4 & 0.4 & 0.2 \\ 0.2 & - & 0.6 & 0.08 \\ 0.2 & 0.6 & - & 0.36 \\ 0.2 & 0.16 & 0.24 & - \end{pmatrix}$$

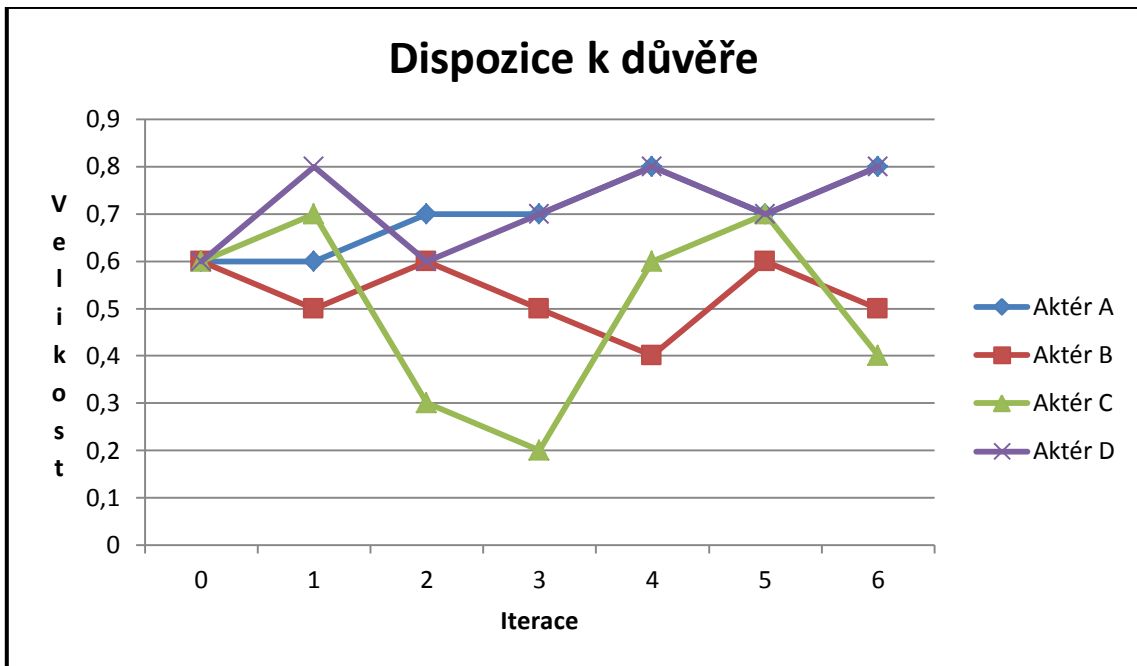
Počáteční důvěra mezi prvky, které se neznají je určena z jejich počáteční reputace a ovlivněna váhou reputace. To je pochopitelné vzhledem k tomu, že žádný jiný kontakt spolu tato diáda prvků do té doby neměla.

Protože při modelování důvěry závisí v každé iteraci na momentální dispozici k důvěře, jsou v tabulce 22 uvedeny její hodnoty pro každý krok.

Tabulka 22 - Dispozice k důvěře v jednotlivých iteracích v první části experimentu

Iterace	0	1	2	3	4	5	6
Aktér A	0.6	0.6	0.7	0.7	0.8	0.7	0.8
Aktér B	0.6	0.5	0.6	0.5	0.4	0.6	0.5
Aktér C	0.6	0.7	0.3	0.2	0.6	0.7	0.4
Aktér D	0.6	0.8	0.6	0.7	0.8	0.7	0.8

Tabulka 22 uvádí, že aktér C trpí velkými výkyvy dispozice k důvěře na rozdíl od svých kolegů. Nejstabilnější se jeví aktéři A a D, u nichž dochází jen k malým změnám oproti normálu. Pro větší přehlednost je ještě dispozice k důvěře znázorněna v grafu 1.

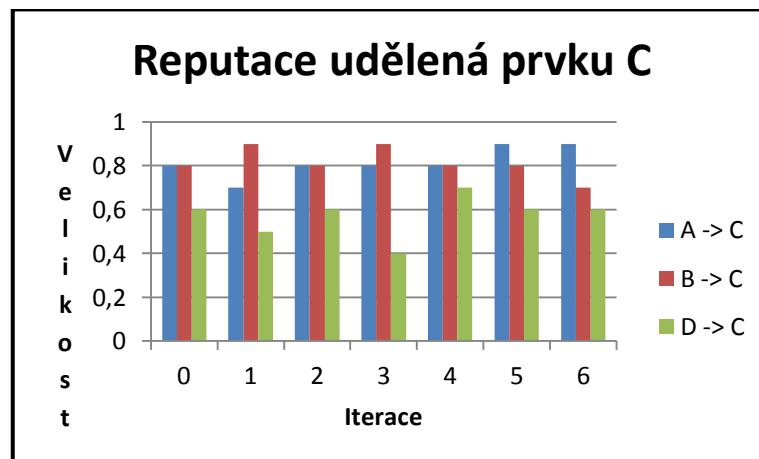


Graf 1 - Vývoj dispozice k důvěře v první části experimentu

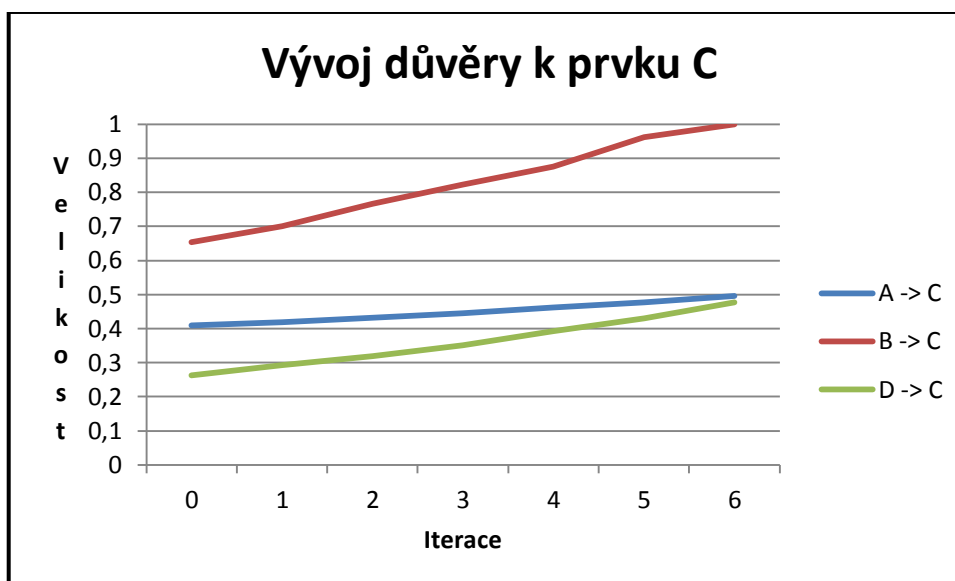
6.2.2 Výsledky první části experimentu

Vývoj důvěry proběhl v sedmi iteracích, které trvaly dohromady 58 dní. Výsledky vývoje modelování důvěry lze sledovat v grafu 7, který se nachází na konci této kapitoly. My si předtím popíšeme několik specifických situací, ke kterým během modelování důvěry došlo.

Z obecného hlediska je nejdůvěryhodnějším prvkem aktér C, který je v síti velice oblíben a každou iteraci obdrží od svých kolegů vysokou reputaci. Právě proto jeho důvěryhodnost u všech aktérů nejvíce roste. Reputaci udílenou prvku C můžeme nalézt v grafu 2 a vývoj důvěry v grafu 3.



Graf 2 - Reputace udělená prvku C ostatními aktéry v první části experimentu



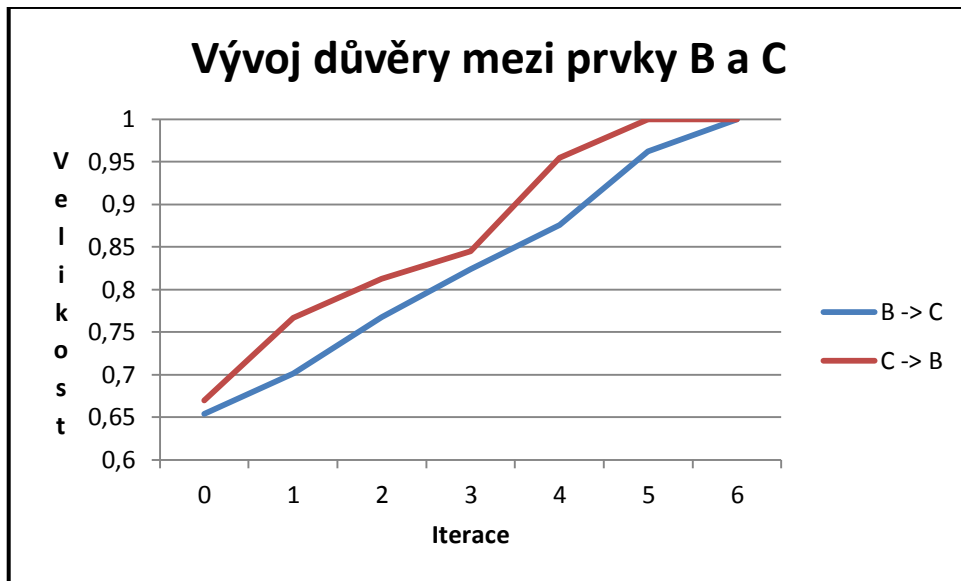
Graf 3 - Vývoj důvěry k prvku C u ostatních aktérů v první části experimentu

Nejmenší nárůst důvěry k prvku C prodělal prvek A, protože důvěryhodnost v prvek C roste hlavně na základě reputace, kterou prvek A neuznává (váha reputace prvku A je nízká).

Prvky B a D s průměrnou hodnotou váhy reputace si potom vedli jinak. I přesto největší nárůst důvěry nastal u prvku B. Tento fakt se odráží od vysoké počáteční důvěry. Oba aktéři se znali již před začátkem experimentu a byli dobrými přáteli. To znamenalo, že trend důvěry mezi subjekty B a C byl podstatně vyšší než mezi prvky D a C a důvěra rostla rychleji.

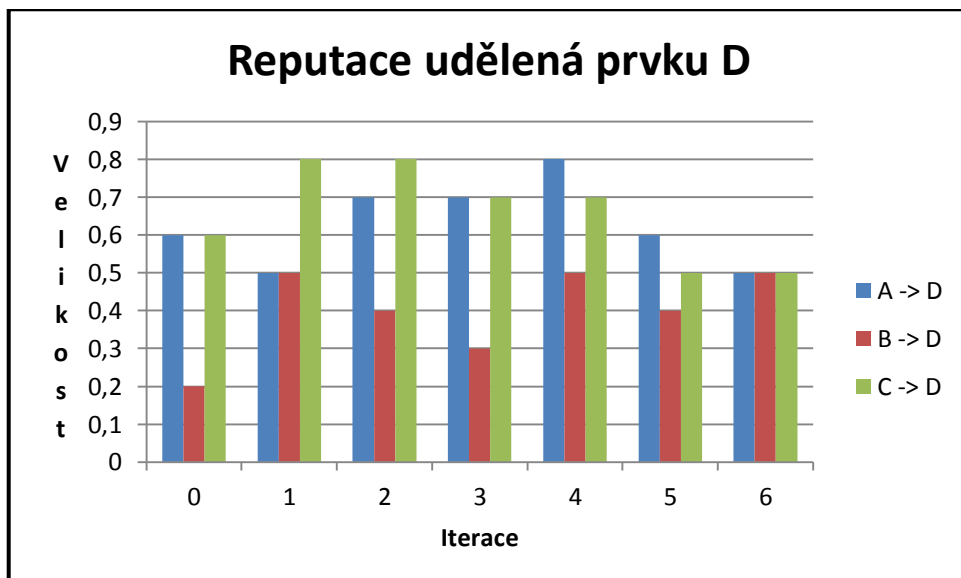
Jak již bylo naznačeno, nejpevnější vztah mezi sebou mají prvky B a C, které již v poslední iteraci první části můžeme označit za nejlepší přátele. Nárůst důvěry k prvku C byl již vysvětlen dříve. Na rozdíl od nárůstu velikosti míry důvěry k subjektu C prvku B byl vcelku lineární. Odlišností byly výkyvy hodnot dispozice k důvěře prvku C. Vývoj důvěry mezi prvky B a C lze vidět v grafu 4.

Opakem k nejdůvěryhodnějšímu prvku C je prvek D. Tento aktér během první části experimentu nenašel společnou řeč s prvkem B, který je v kolektivu oblíben. To se hodně odrazilo na obdržené reputaci od prvku B. V pozdějších iteracích se nechali ovlivnit prvkem B i ostatní aktéři a obdržaná reputace pro prvek D začala klesat i od nich. Výsledkem je nejmenší důvěryhodnost prvku D. Graf obdržené reputace nalezneme pod číslem 5 a vývoj důvěry k prvku D pod číslem 6.

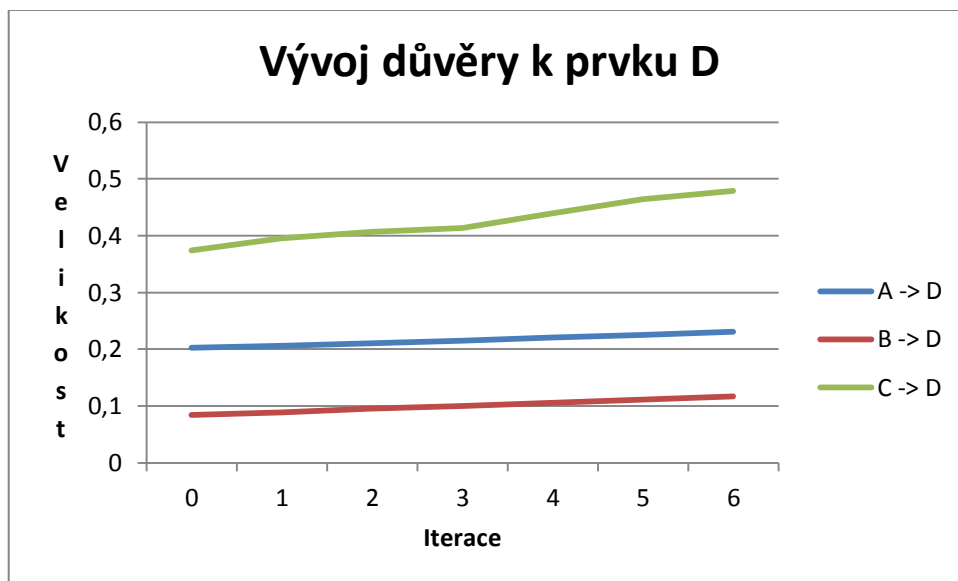


Graf 4 - Vývoj důvěry mezi prvky B a C v první části experimentu

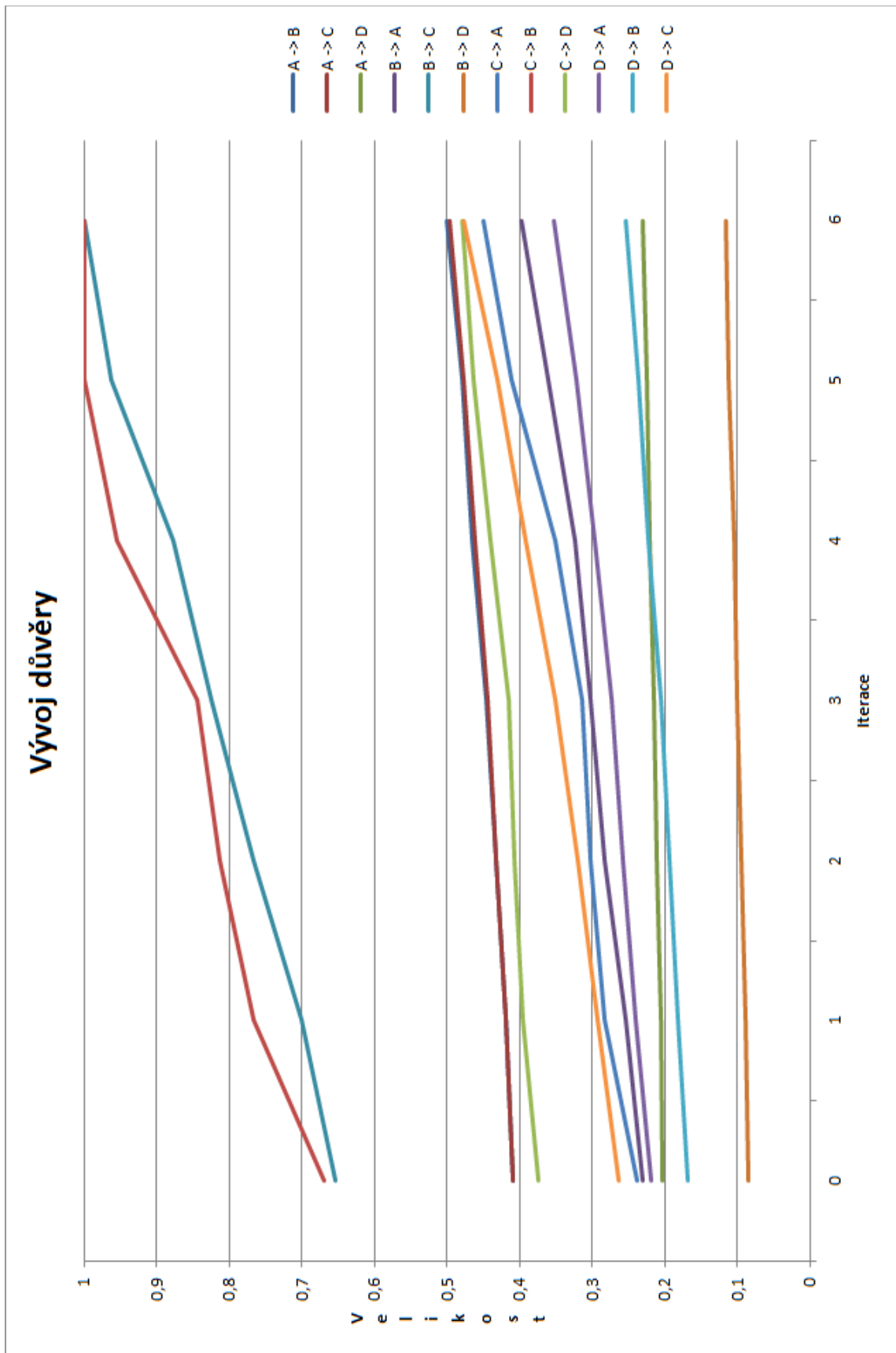
Mezi ostatními prvky nedocházelo k žádným velkým změnám a nárůst důvěry byl pomalý a vcelku konstantní. V této části, jak jsme si mohli povšimnout, záleží hlavně na reputaci udílené od ostatních prvků a váze reputace. Odchytky mezi nárůsty jsou pak způsobeny hlavně kolísavou hodnotou dispozice k důvěře. Vývoj všech prvků lze shlédnout v grafu 7.



Graf 5 - Reputace udělená prvku D ostatními aktéry v první části experimentu



Graf 6 - Vývoj důvěry k prvku D u ostatních aktérů v první části experimentu



Graf 7 - Vývoj důvěry všech aktérů v první části experimentu

6.3 Druhá část experimentu – sedmičlenná skupina

6.3.1 Základní hodnoty

Ve druhé části experimentu se síť rozrostla na sedm členů, přičemž první čtyři aktéři zůstávají stále stejní. Sociální síť si nejprve popíšeme maticí sousednosti S .

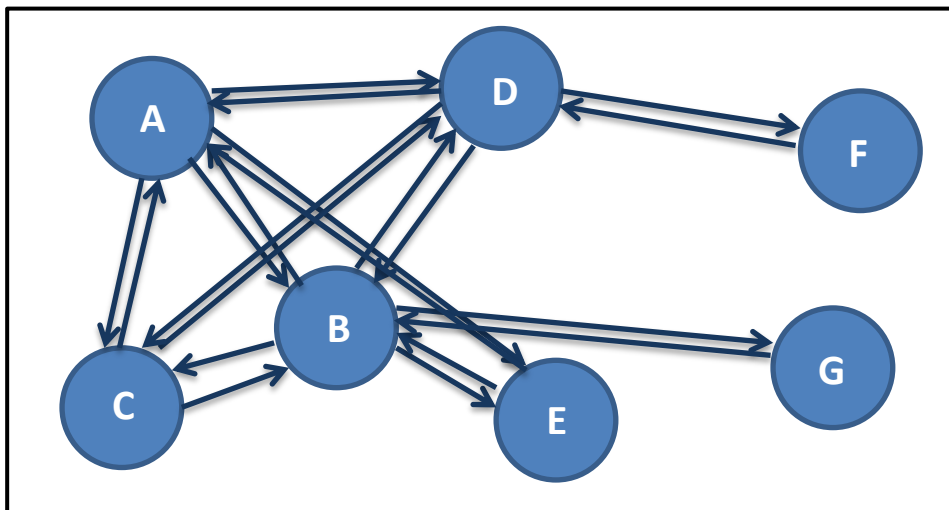
$$S = \begin{pmatrix} - & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & - & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & - & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & - & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & - & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & - & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & - \end{pmatrix}$$

Spojení jednotlivých prvků s pozicí řádků a sloupců v matici sousednosti S je zachyceno v tabulce 23.

Tabulka 23 - Prvky a jejich pozice v matici ve druhé části experimentu

Prvek	A	B	C	D	E	F	G
Pozice prvku	0	1	2	3	4	5	6

Pro větší přehlednost je na obrázku 28 síť všech aktérů a jejich vazeb. Sociální síť přesně odpovídá matici sousednosti S .



Obrázek 22 - Sociální síť aktérů ve druhé části experimentu

Z obrázku sociální sítě aktérů je patrné, že aktéři z minulé části experimentu vytvořili jádro sítě. Centrální uzlem celé sítě i nadále zůstává prvek B, který se zná s novým aktérem E a zároveň přivedl aktéra G. Jediný prvek, se kterým se zná jen přes uzel D, je subjekt F. Dva ze tří nových aktérů F a G se stali nejizolovanějšími prvky celé této sítě. Stupeň uzlu je roven jedné. Matice počáteční důvěry má tvar:

$$\begin{pmatrix} - & 0.499 & 0.496 & 0.231 & 1.0 & 0.146 & 0.1 \\ 0.397 & - & 1.0 & 0.116 & 0.2 & 0.292 & 0.2 \\ 0.45 & 1.0 & - & 0.48 & 0.408 & 0.438 & 0.3 \\ 0.353 & 0.255 & 0.476 & - & 0.292 & 0.8 & 0.2 \\ 1.0 & 0.2 & 0.292 & 0.2 & - & 0.292 & 0.2 \\ 0.28 & 0.288 & 0.292 & 0.8 & 0.292 & - & 0.2 \\ 0.14 & 0.144 & 0.146 & 0.1 & 0.136 & 0.146 & - \end{pmatrix}$$

Každému z aktérů přísluší i další počáteční veličiny, kterou je váha doporučení, váha reputace a váha dispozice k důvěře. Tyto veličiny jsou po zbytek experimentu neměnné na rozdíl od veličin ostatních závislých na iteraci modelování důvěry. Jednotlivé váhy pro každý subjekt jsou uvedeny v tabulce 24.

Tabulka 24 - Váha doporučení, reputace a dispozice každého aktéra sítě ve druhé části experimentu

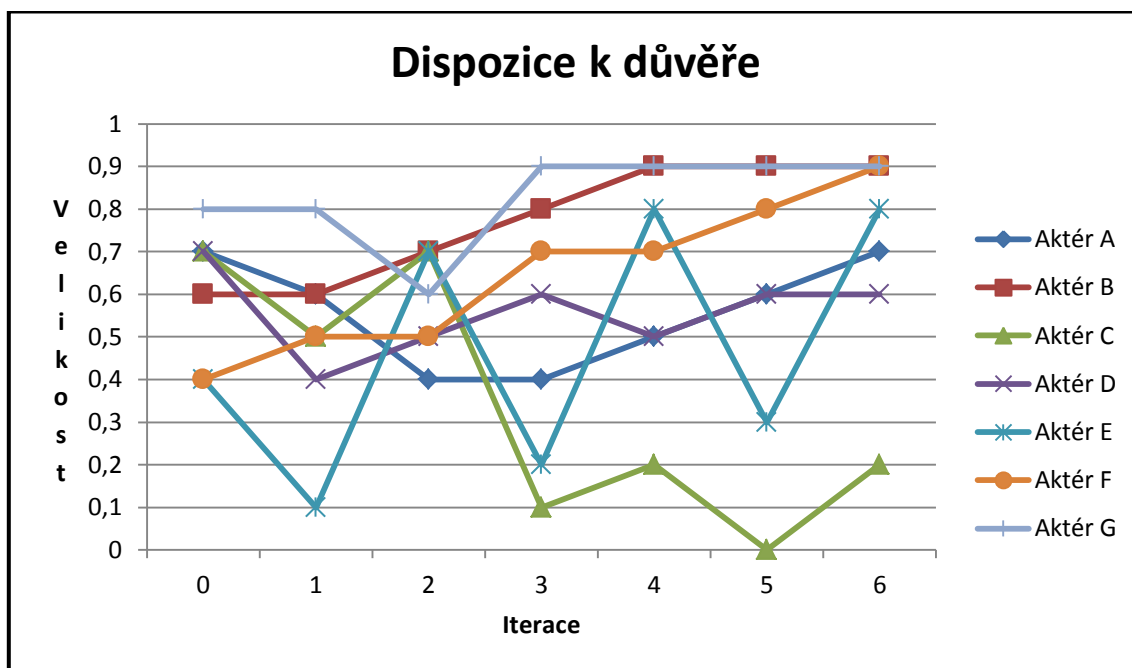
Prvek	A	B	C	D	E	F	G
Váha doporučení	0.6	0.4	0.4	0.4	0.4	0.4	0.8
Váha reputace	0.2	0.4	0.6	0.4	0.4	0.4	0.2
Váha dispozice k důvěře	0.5	0.7	0.6	0.6	0.7	0.6	0.4

Poslední důležitou veličinou, kterou je třeba si uvést je dispozice k důvěře. Tato veličina se každou iterací mění, a proto je zaznamenána v tabulce 25 v několika po sobě jdoucích krocích.

Tabulka 25 - Dispozice k důvěře v jednotlivých iteracích ve druhé části experimentu

Iterace	0	1	2	3	4	5	6
Aktér A	0.7	0.6	0.4	0.4	0.5	0.6	0.7
Aktér B	0.6	0.6	0.7	0.8	0.9	0.9	0.9
Aktér C	0.7	0.5	0.7	0.1	0.2	0.0	0.2
Aktér D	0.7	0.4	0.5	0.6	0.5	0.6	0.6
Aktér E	0.4	0.1	0.7	0.2	0.8	0.3	0.8
Aktér F	0.4	0.5	0.5	0.7	0.7	0.8	0.9
Aktér G	0.8	0.8	0.6	0.9	0.9	0.9	0.9

Pro větší představivost byly dispozice k důvěře vyvedeny na následující spojnicový graf 8. Z grafu jasně vyplývá pokles a následný nárůst dispozice u aktéra A, pomalu se zvyšující dispozice u aktéra B, velký propad dispozice u aktéra C, průměrné kolísavé hodnoty dispozice aktéra D, nízké hodnoty s občasnými výkyvy u aktéra E, zvyšující se dispozici aktéra F a konečně nárůst dispozice u aktéra F.

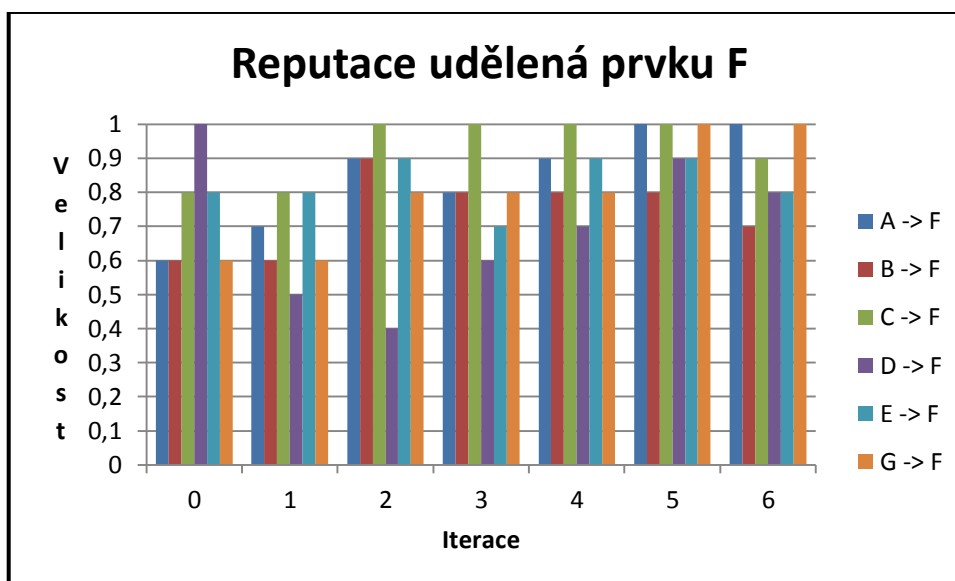


Graf 8 - Vývoj dispozice k důvěře ve druhé části experimentu

6.3.2 Výsledky druhé části experimentu

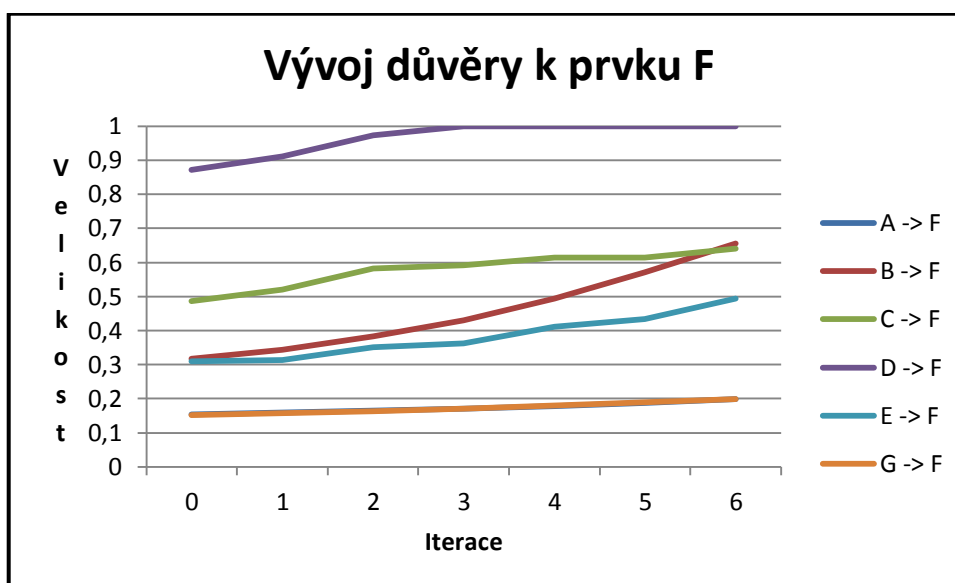
Druhá část experimentu probíhala tak jako první v sedmi iteracích a trvala celkem 61 dní. Subjekty A, B, C a D spolu v rámci sociální sítě strávili již 58 dní. Zbýlé tři subjekty se musely začlenit do komunity, která se tedy již znala. Jako tomu bylo v kapitole 6.2.2 s výsledky vývoje důvěry první části experimentu, je i na konci této kapitoly uveden podrobný graf 15 s vývojem důvěry všech subjektů.

První se podíváme na prvek, který díky své vysoké reputaci získával u všech ostatních aktérů nejrychleji na své důvěryhodnosti. Je to jeden z nových aktérů v sociální síti a to konkrétně uzel F. V grafu 9 můžeme vidět, že kromě první a druhé iterace nedostal subjekt F nikdy menší reputaci než 0.6. To ve výsledném vývoji důvěry mělo za následek velmi rychlý nárůst míry důvěry u prvků s vysokou váhou reputace a střední růst u prvků ostatních.



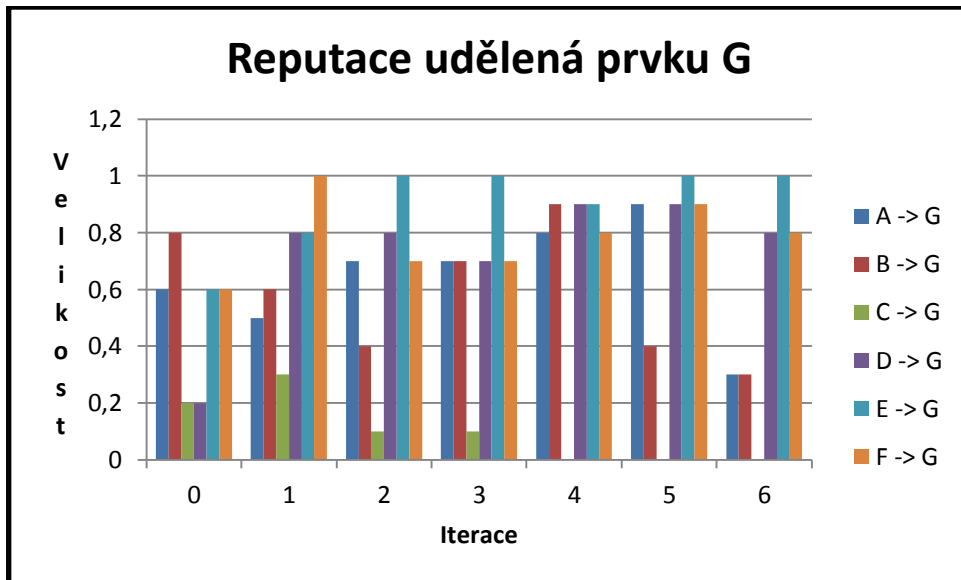
Graf 9 - Reputace udělená prvku F ostatními aktéry v druhé části experimentu

Vývoj míry důvěry subjektu F je zakreslen v grafu 10. Další zajímavostí je i rozdíl růstu důvěry k prvku F u prvků B a C, které mají podobné sklony ke zvyšování důvěry. U prvku C se ale míra důvěry zvyšuje podstatně pomaleji kvůli velmi nízké dispozici k důvěře, která v případě třetí až šesté iterace nabývá minimálních hodnot a vývoj důvěry je tak skoro zanedbatelný. Vezmeme-li ještě v potaz modelování míry důvěry prvku E k prvku F. To by mohlo být také velmi rychlé, ale musel by aktér F dostávat doporučení od prvku D jako tomu je u předchozích zmiňovaných B a C.



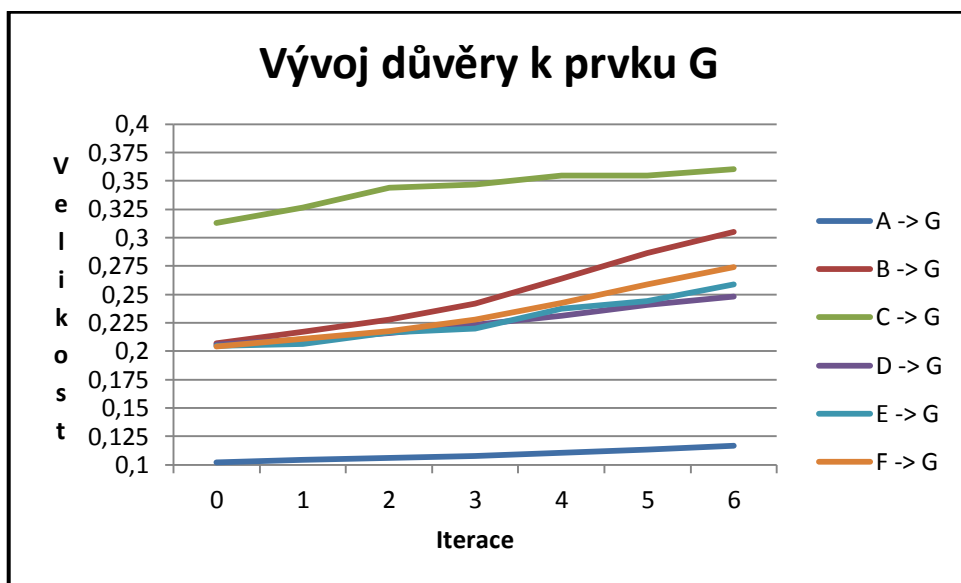
Graf 10 - Vývoj důvěry k prvku F u ostatních aktérů v druhé části experimentu

Nejizolovanějším a zároveň nejméně oblíbeným prvkem se v druhé části experimentu stal prvek G, kterého sice doporučil jeden z hlavních prvků sociální sítě aktér B, ale u všech aktérů měl nízkou počáteční důvěru a v průběhu byl některými subjekty velmi negativně hodnocen, viz graf 11.



Graf 11 - Reputace udělená prvku G ostatními aktéry v druhé části experimentu

Vývoj důvěry je tedy na rozdíl od minulého příkladu velmi malý. Největší růst je zaznamenán u aktéra B, který prvek doporučil. Tato situace je popsána grafem 12.



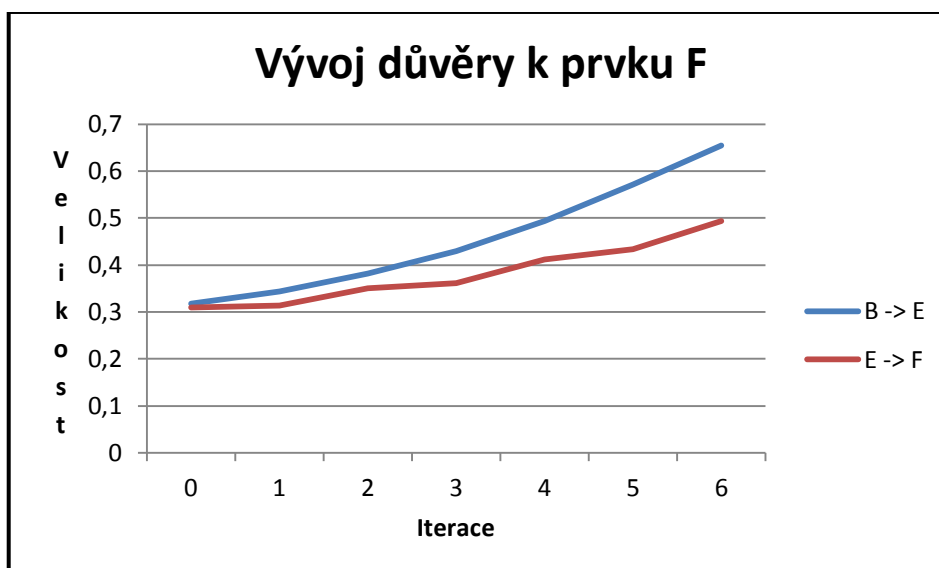
Graf 12 - Vývoj důvěry k prvku F u ostatních aktérů v druhé části experimentu

Dále se zaměříme na ovlivnění modelování důvěry v závislosti na váze doporučení, váze reputace a váze dispozice k důvěře. Nejlepším příkladem jsou prvky B a G. G má sice vysokou váhu doporučení, ale žádné doporučení v druhé části experimentu nedostává. Proto ji můžeme zanedbat a brát aktéra G jako prvek s velmi nízkými váhami. Prvek B je naopak aktér se středně velkými váhami, v naší sociální síti dokonce s jedněmi z největších. Oba aktéři mají i většinu zbylých veličin velmi podobnou a přesto je nárůst míry důvěry k prvku A diametrálně odlišný právě kvůli velikosti vah reputace a dispozice, viz graf 13.



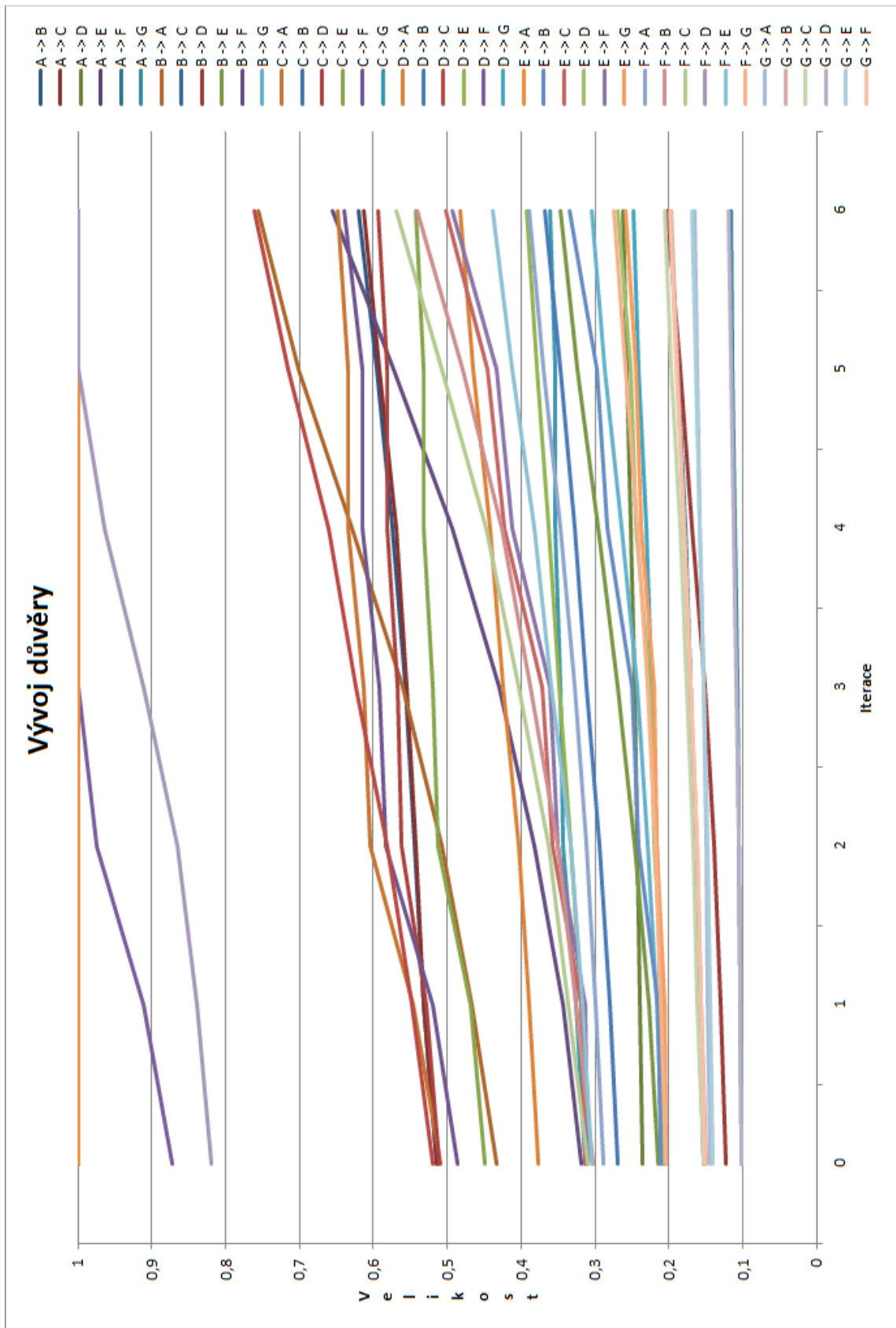
Graf 13 - Vývoj důvěry k prvku A u aktérů B a G v druhé části experimentu

Posledním zajímavým úkazem je působení doporučení na vývoj důvěry mezi dvěma subjekty znázorněný v grafu 14. Vezmeme aktéry B a E a jejich vývoj míry důvěry k subjektu F. Aktér B dostává první tři iterace na aktéra F doporučení od prvku C, který je aktérovi B velmi blízkým přítelem a jeho doporučení má tak velkou váhu. Na rozdíl od něj aktér E žádná doporučení nepřijímá. Hodnoty váhy doporučení, reputace a dispozice mají oba aktéři skoro stejné. Zatímco míra důvěry mezi prvky E a F vzroste o 0.2, mezi prvky B a F o celých 0.35. Je tedy zřejmé, jak moc ovlivní zisk doporučení od dobrého přítele vývoj důvěry.



Graf 14 - Vývoj důvěry k prvku F u aktérů B a E v druhé části experimentu

Ve druhé části experimentu jsme si ukázali několik zajímavých situací, ke kterým při modelování důvěry došlo a jak dokáží jednotlivé veličiny ovlivnit její vývoj mezi dyádami prvků. V grafu 15 jsou vidět všechny výsledky této části experimentu.



Graf 15 - Vývoj důvěry všech aktérů v druhé části experimentu

6.4 Třetí část experimentu – dvanáctičlenná skupina

6.4.1 Základní hodnoty

Finální třetí část experimentu se pyšní sociální sítí o dvanácti členech, z nichž jich je sedm stejných z minulé části experimentu. Tak jako tomu bylo v předchozím případě, i tady někteří účastníci z druhé části přivedli nové aktéry. Sociální síť si nejprve zapíšeme do matice sousednosti S .

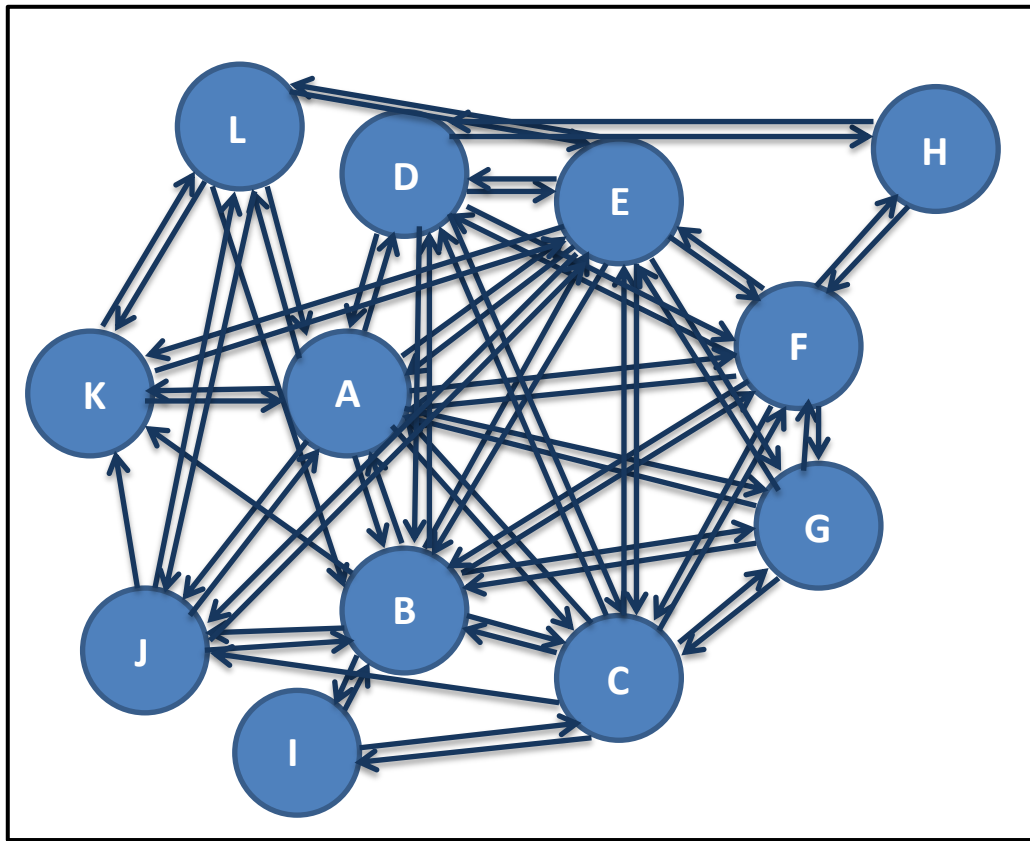
$$S = \begin{pmatrix} - & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & - & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & - & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & - & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & - & 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & - & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & - & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & - & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & - & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & - & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & - & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & - \end{pmatrix}$$

Pro úplnost jsou pozice prvků s jejich jmény poznamenány v následující tabulce 26.

Tabulka 26 - Prvky a jejich pozice v matici ve třetí části experimentu

Prvek	A	B	C	D	E	F	G	H	I	J	K	L
Pozice prvku	0	1	2	3	4	5	6	7	8	9	10	11

Nakonec si ještě zobrazíme celou sociální síť jako graf s dvanácti uzly a všemi vazbami mezi prvky, které se spolu již před začátkem modelování znají.



Obrázek 23 - Sociální síť aktérů ve třetí části experimentu

Sociální síť aktérů je již pro dvanáct aktérů značně nepřehledná. Důležitým faktem je, že aktéři A, B, C, D, E, F a G z druhé části experimentu tvoří jádro celé sítě. Nejdůležitějšími aktéry sítě jsou subjekty A a B, kteří jsou centrálními prvky sítě. To je dáno dvěma věcmi:

- oba aktéři A i B jsou v síti od samého počátku,
- oba aktéři A i B přizvali do sociální sítě nejvíce nových účastníků.

Nejizolovanějšími uzly sítě jsou dva z nových aktérů, konkrétně subjekt H a I. Každých z nich má po dvou sousedech. Matice počáteční důvěry T má tvar:

$$T = \begin{pmatrix} - & 0.595 & 0.591 & 0.257 & 1.0 & 0.188 & 0.114 & 0.116 & 0.12 & 0.8 & 1.0 & 0.8 \\ 0.702 & - & 1.0 & 0.185 & 0.324 & 0.572 & 0.287 & 0.232 & 0.8 & 0.2 & 0.2 & 0.312 \\ 0.634 & 1.0 & - & 0.581 & 0.532 & 0.615 & 0.355 & 0.348 & 0.2 & 0.2 & 0.402 & 0.468 \\ 0.466 & 0.348 & 0.715 & - & 0.379 & 1.0 & 0.241 & 0.6 & 0.24 & 0.256 & 0.268 & 0.312 \\ 1.0 & 0.298 & 0.446 & 0.252 & - & 0.434 & 0.244 & 0.232 & 0.24 & 0.8 & 0.6 & 0.8 \\ 0.37 & 0.48 & 0.507 & 1.0 & 0.412 & - & 0.259 & 0.6 & 0.24 & 0.256 & 0.268 & 0.312 \\ 0.163 & 0.188 & 0.196 & 0.116 & 0.164 & 0.189 & - & 0.116 & 0.12 & 0.128 & 0.134 & 0.156 \\ 0.288 & 0.26 & 0.252 & 0.6 & 0.276 & 0.8 & 0.164 & - & 0.24 & 0.256 & 0.268 & 0.312 \\ 0.576 & 0.8 & 0.2 & 0.488 & 0.552 & 0.56 & 0.328 & 0.464 & - & 0.512 & 0.536 & 0.624 \\ 0.8 & 0.13 & 0.126 & 0.122 & 0.8 & 0.14 & 0.082 & 0.116 & 0.12 & - & 0.2 & 0.8 \\ 0.6 & 0.13 & 0.126 & 0.122 & 0.2 & 0.14 & 0.082 & 0.116 & 0.12 & 0.134 & - & 0.2 \\ 0.8 & 0.2 & 0.378 & 0.366 & 0.8 & 0.414 & 0.42 & 0.348 & 0.36 & 0.8 & 0.6 & - \end{pmatrix}$$

Aby byl výčet veličin úplný, uvedeme zde i váhy doporučení, váhy reputace a váhy dispozice k důvěře pro každý prvek, viz tabulka 27. Tyto veličiny jsou po celou dobu modelování důvěry neměnné.

Tabulka 27 - Váha doporučení, reputace a dispozice každého aktéra sítě ve třetí části experimentu

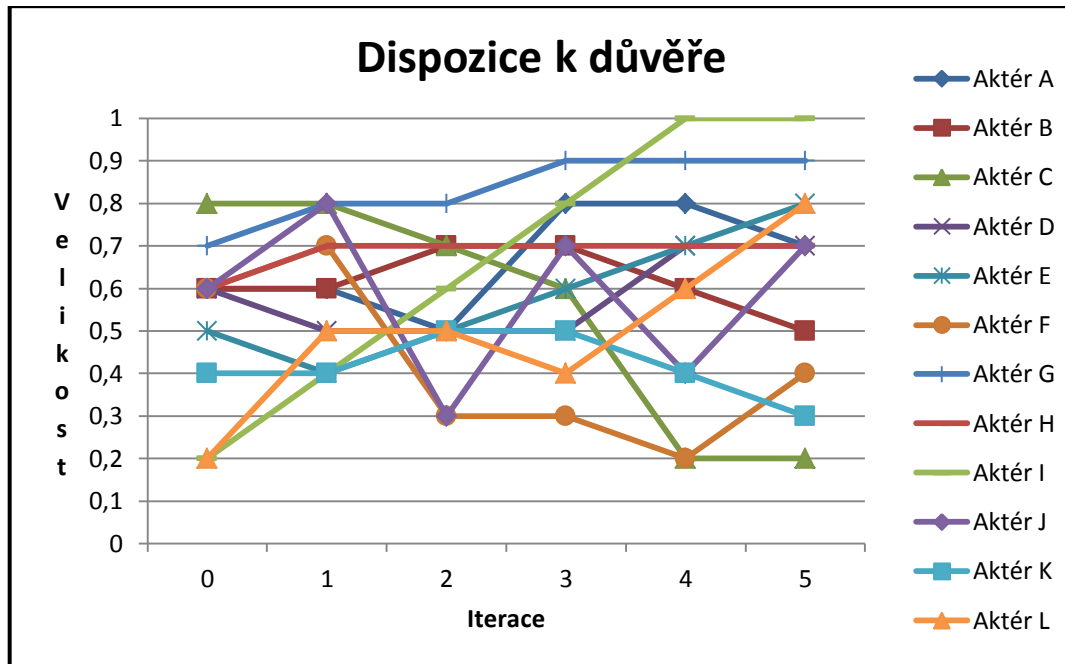
Prvek	A	B	C	D	E	F
Váha doporučení	0.6	0.4	0.4	0.4	0.4	0.4
Váha reputace	0.2	0.4	0.6	0.4	0.4	0.4
Váha dispozice k důvěře	0.5	0.7	0.6	0.6	0.7	0.6
Prvek	G	H	I	J	K	L
Váha doporučení	0.8	0.2	0.8	0.8	0.0	0.4
Váha reputace	0.2	0.4	0.8	0.2	0.2	0.6
Váha dispozice k důvěře	0.4	0.5	0.8	0.3	0.6	0.4

Poslední, co zbývá uvést, jsou dispozice k důvěře pro každý uzel sociální sítě v jednotlivých iteracích. Všechny hodnoty jsou vypsány v tabulce 28.

Tabulka 28 - Dispozice k důvěře v jednotlivých iteracích ve třetí části experimentu

Iterace	0	1	2	3	4	5	6
Aktér A	0.6	0.6	0.5	0.8	0.8	0.7	0.6
Aktér B	0.6	0.6	0.7	0.7	0.6	0.5	0.7
Aktér C	0.8	0.8	0.7	0.6	0.2	0.2	0.8
Aktér D	0.6	0.5	0.5	0.5	0.7	0.7	0.7
Aktér E	0.5	0.4	0.5	0.6	0.7	0.8	0.8
Aktér F	0.6	0.7	0.3	0.3	0.2	0.4	0.5
Aktér G	0.7	0.8	0.8	0.9	0.9	0.9	0.9
Aktér H	0.6	0.7	0.7	0.7	0.7	0.7	0.7
Aktér I	0.2	0.4	0.6	0.8	1.0	1.0	0.2
Aktér J	0.6	0.8	0.3	0.7	0.4	0.7	0.2
Aktér K	0.4	0.4	0.5	0.5	0.4	0.3	0.3
Aktér L	0.2	0.5	0.5	0.4	0.6	0.8	0.8

Grafické znázornění dispozice k důvěře lze spatřit v následujícím grafu 16. U prvků jako jsou například E, F, I, K a L se dispozice velmi často výrazně mění. Ostatní prvky ji mají většinou konstantní nebo jim lineárně roste. Takové prvky jsou při modelování důvěry stálejší a důvěra k jiným subjektům nabývá pravidelněji.

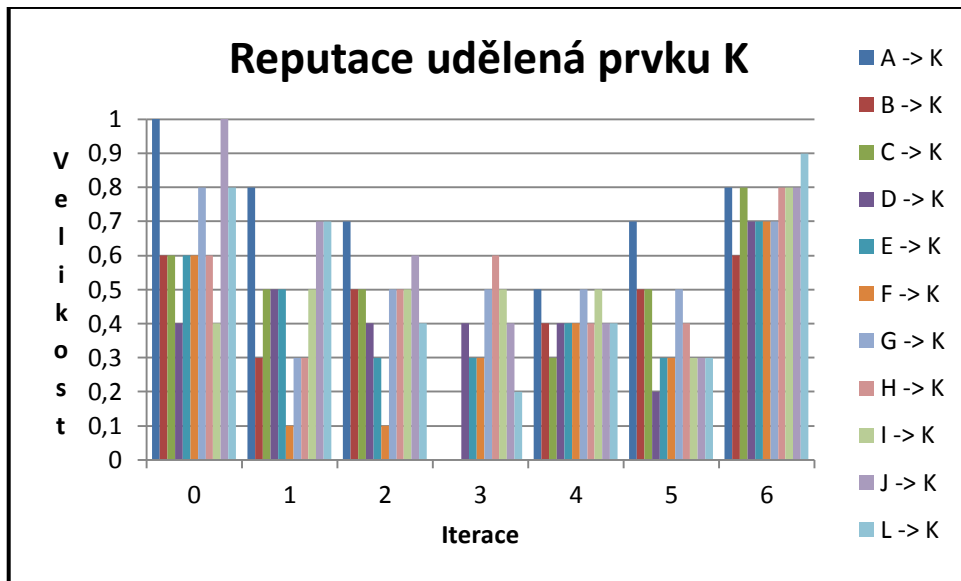


Graf 16 - Vývoj dispozice k důvěře ve třetí části experimentu

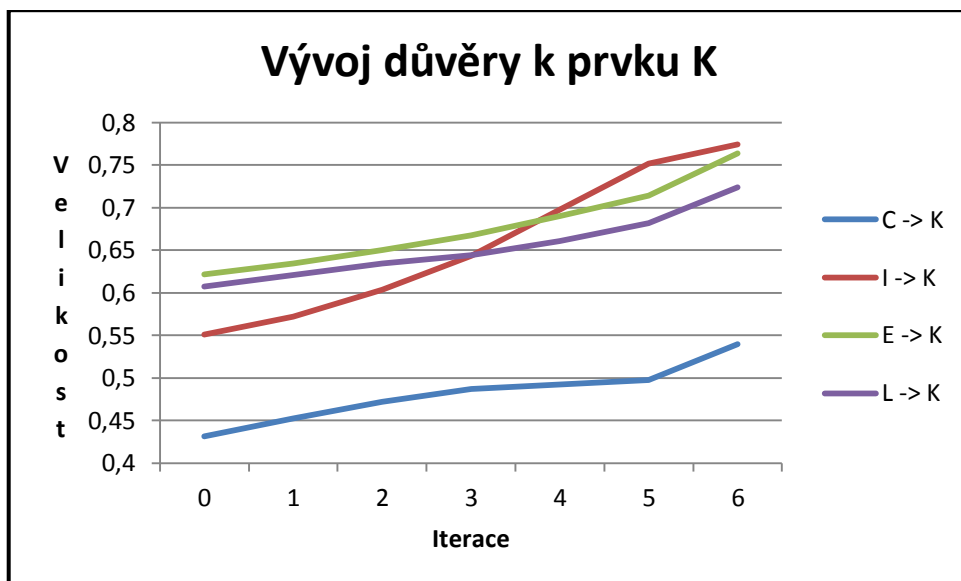
6.4.2 Výsledky třetí části

Poslední třetí část experimentu probíhala s dvanácti aktéry v sociální síti a trvala 65 dní. Tak jako v dřívějších kapitolách i zde se z nasbíraných dat modelovala míra důvěry po iteracích. Pro velké množství výsledků zde není uveden celkový graf s vývojem důvěry a výsledky lze nalézt pouze na CD přiloženém v této práci. Protože již bylo ovlivnění vývoje důvěry všech veličiny popsáno v minulých dvou částech experimentu, zaměříme se v tomto úseku hlavně na speciální případy příhodně zadaných dat.

Jako první si povšimneme udělování reputací prvku K, kdy zpočátku velikost reputace jen klesá v některých případech dokonce až k nulové hranici a v poslední iteraci prudce stoupne, viz graf 17. To vypovídá o malé aktivitě aktéra v předchozích iteracích a velkém zapojení do diskuze v iteraci poslední. Velikost míry důvěry k prvku K v posledním kroku tak díky nenadálé aktivitě stoupl. Pro příklad bylo vybráno jen několik příkladů s aktéry, u kterých se změna projevila nejvíce (dáno vysokou momentální dispozicí a váhou reputace a dispozice k důvěře). Vývoj důvěry aktérů C, E, I a L k subjektu K jsou v grafu 18.

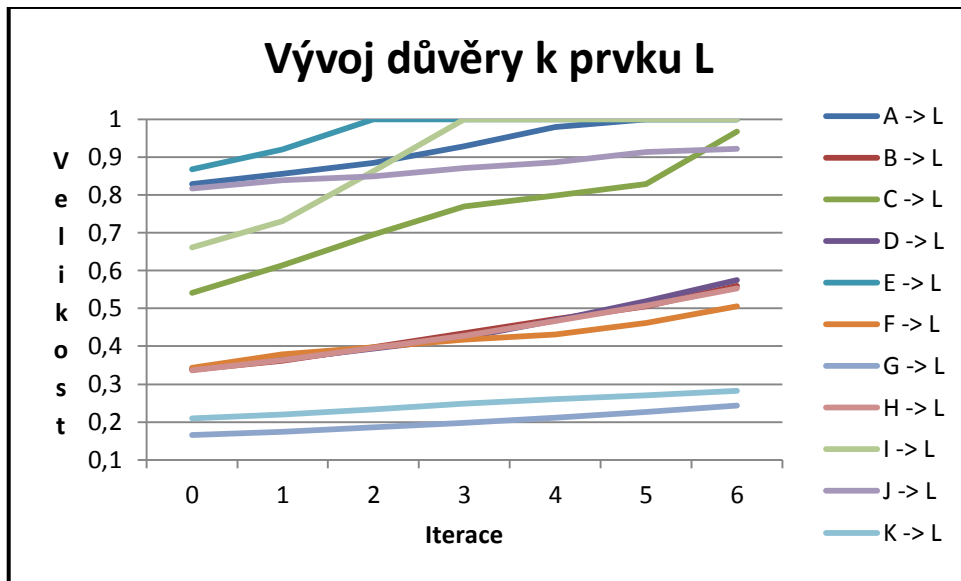


Graf 17 - Reputace udělená prvku K ostatními aktéry ve třetí části experimentu



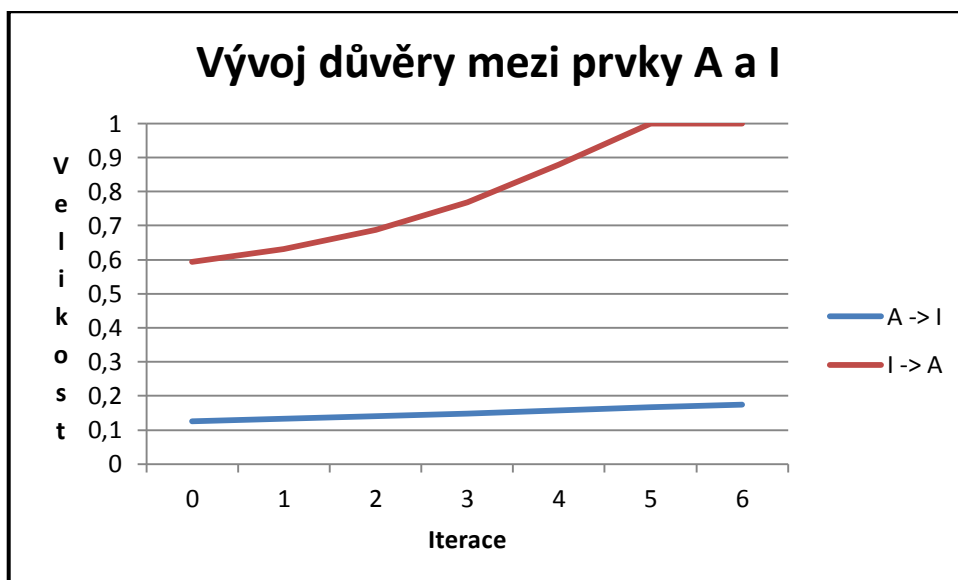
Graf 18 - Vývoj důvěry k prvku K u aktérů C, I, E a L ve třetí části experimentu

Další zajímavostí třetího úseku experimentu je neobyčejný nárůst míry důvěry u nového aktéra v sociální síti. Konkrétně se jedná o aktéra L, který díky dobrému doporučení od subjektu A, obstojné dispozici, dobré pověsti a velké oblíbenosti udílené ostatními prvky ve všech iteracích, dosáhl největšího nárůstu míry důvěry za sedm iterací ve všech třech částech experimentu. Hodnoty míry důvěry v jednotlivých iteracích jsou v grafu 19.



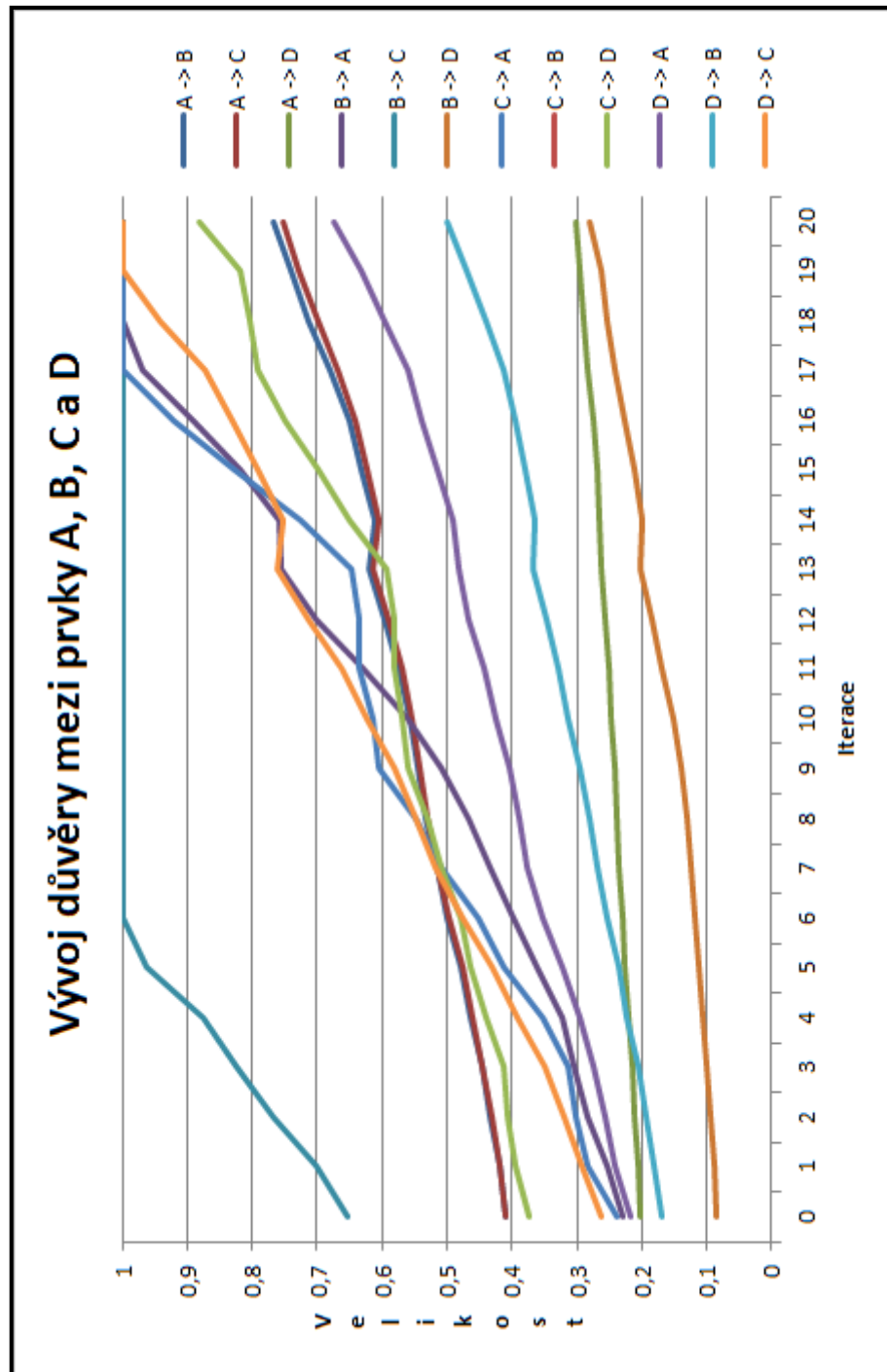
Graf 19 - Vývoj důvěry k prvku L u všech aktérů ve třetí části experimentu

Ve třetí části experimentu si můžeme taky povšimnout klasické vlastnosti důvěry a tou je asymetričnost, kdy jeden prvek důvěřuje druhému a druhý prvnímu ne. Příkladem jsou například aktéři A a I. Vývoj jejich důvěry je zachycen v grafu 20. V tomto případě se jedná o asymetričnost z důvodů nadprůměrné reputace prvku A a vysoké váhy reputace prvku I a obrácené situace.



Graf 20 - Vývoj Důvěry mezi prvky A a I ve třetí části experimentu

Nakonec se podíváme na vývoj důvěry prvních čtyř prvků A, B, C a D, které byly součástí této sociální sítě celých 184 dní od první pasáže pokusů až k třetí poslední. Následující graf zachycuje vývoj důvěry ve všech 21 iteracích.



Graf 21 - Vývoj důvěry mezi prvky A, B, C a D od začátku experimentu

6.5 Zhodnocení experimentu

Jak bylo zmíněno v kapitole 6.1, celý experiment končil posledním dotazníkem, ve kterém každý ze zúčastněných vyplnil celkové hodnocení každého aktéra za celou dobu, co s ním strávil ve skupině. V následující tabulce 29 je porovnána poslední vypočtená hodnota míry důvěry každého prvku (= MD) s odpovědí vyplněnou v dotazníku (= O), kdy v každém řádku je hodnotící prvek a ve sloupci prvek, který je hodnocen. Výsledky jsou si velice podobné, a proto můžeme prohlásit, že model personální důvěry v sociální síti dosahuje dobrých výsledků.

Tabulka 29 - Porovnání výsledků modelování s odpověďmi v dotaznících

	A		B		C		D		E		F	
	MD	O	MD	O	MD	O	MD	O	MD	O	MD	O
Aktér A	-	-	0.77	0.9	0.75	0.9	0.30	0.4	1.0	0.9	0.26	0.3
Aktér B	1.0	1.0	-	-	1.0	1.0	0.28	0.4	0.58	0.7	0.99	0.8
Aktér C	1.0	1.0	1.0	0.8	-	-	0.88	0.7	1.0	1.0	1.0	0.8
Aktér D	0.67	0.8	0.5	0.5	1.0	0.9	-	-	0.57	0.5	1.0	0.9
Aktér E	1.0	0.9	0.54	0.6	0.82	0.9	0.36	0.4	-	-	0.77	0.8
Aktér F	0.48	0.6	0.7	0.6	0.74	0.9	1.0	0.9	0.6	0.7	-	-
Aktér G	0.2	0.3	0.25	0.3	0.26	0.5	0.14	0.2	0.22	0.3	0.25	0.1
Aktér H	0.39	0.4	0.4	0.3	0.41	0.6	0.76	0.6	0.42	0.3	1.0	0.8
Aktér I	1.0	0.9	1.0	0.9	0.58	0.4	0.76	0.7	1.0	0.8	1.0	0.9
Aktér J	0.9	0.8	0.15	0.3	0.15	0.2	0.13	0.2	0.9	0.9	0.16	0.3
Aktér K	0.74	0.7	0.16	0.3	0.17	0.2	0.14	0.2	0.27	0.4	0.18	0.2
Aktér L	1.0	0.9	0.35	0.4	0.63	0.5	0.46	0.6	1.0	1.0	0.63	0.7
	G		H		I		J		K		L	
	MD	O	MD	O	MD	O	MD	O	MD	O	MD	O
Aktér A	0.12	0.3	0.14	0.2	0.18	0.1	0.99	1.0	1.0	1.0	1.0	1.0
Aktér B	0.32	0.3	0.32	0.3	1.0	0.9	0.31	0.4	0.28	0.3	0.56	0.6
Aktér C	0.48	0.6	0.36	0.6	0.33	0.5	0.33	0.2	0.54	0.6	0.97	1.0
Aktér D	0.26	0.5	0.76	0.8	0.4	0.4	0.35	0.2	0.34	0.2	0.58	0.6
Aktér E	0.27	0.2	0.32	0.2	0.46	0.5	1.0	0.9	0.76	0.7	1.0	0.9
Aktér F	0.28	0.3	0.74	0.6	0.37	0.3	0.33	0.5	0.32	0.4	0.51	0.7
Aktér G	-	-	0.13	0.2	0.16	0.3	0.15	0.2	0.15	0.3	0.24	0.3
Aktér H	0.18	0.1	-	-	0.39	0.3	0.34	0.3	0.33	0.3	0.55	0.5
Aktér I	0.4	0.5	0.72	0.7	-	-	0.89	0.8	0.77	0.9	1.0	0.9
Aktér J	0.09	0.0	0.13	0.3	0.14	0.1	-	-	0.21	0.4	0.92	1.0
Aktér K	0.09	0.1	0.13	0.2	0.15	0.3	0.16	0.2	-	-	0.28	0.2
Aktér L	0.45	0.4	0.43	0.3	0.57	0.5	1.0	1.0	0.72	0.7	-	-

7 Závěr

Cílem této práce bylo prozkoumat jeden ze způsobů pojetí důvěry a jejího modelování a dále se zaměřit na sociální síť. Důvěra a sociální síť jsou fenoménem moderní doby a zasahují do mnoha oblastí, zejména do informačních technologií. Bylo nutné spojit modelování důvěry se sociální sítí a zkusit využít dostupných simulačních prostředků založených na multiagentních technologiích. Protože jsem se s multiagentními technologiemi a modelováním důvěry v sociální síti setkal poprvé, mají první kapitoly zpravidla informativní charakter.

Kapitoly 2, 3 a 4 řadím do teoretické části, které popisují a vysvětlují základní pojmy a principy, které jsou pro sociální síť a modelování důvěry nezbytné. Jsou v nich představeny dostupné simulační prostředky založené na multiagentních technologiích, zejména JADE, a navržen model důvěry v sociální síti. Tím byly pokryty první tři body zadání.

Další kapitola popisuje softwarové řešení problému modelování důvěry v sociální síti v jazyce Java s využitím knihovny JADE. Při jeho tvorbě jsem si procvičil programátorské dovednosti jako je práce s třídami a objektovým programováním, navržení GUI a přepínatelné lokalizace, tak jako jsem se naučil základní práci se simulačním prostředkem JADE. Aplikace splňuje základní funkcionalitu modelu důvěry, ale bohužel pro svou specifickou nemá široké možnosti uplatnění.

V poslední kapitole 6 jsem model podrobil testům s reálně nasbíranými daty ze sociální sítě Facebook. V jednotlivých experimentech jsem demonstroval vztah veličin podílejících se na utváření důvěry či poukázal na specifické situace vzniklé při modelování. Nakonec jsem porovnal výsledky spočtené navrženým modelem důvěry se získanými daty z dotazníků. Hodnoty byly víceméně shodné, a proto se model důvěry jeví jako správný. Aplikace z kapitoly 5 a sada experimentů s diskusí výsledků vychází z posledního bodu zadání.

K práci je přiloženo CD, na němž se nachází tento dokument v elektronické podobě, výsledky experimentů, všechny nasbírané dotazníky, spustitelné verze programu pro všechny tři části experimentu s jejich vstupními daty, zdrojové soubory, ukázka komunikace mezi agenty a vygenerovaná dokumentace.

Přehled zkratk

GUI	Graphic User interface je grafické uživatelské rozhraní
GNU	General Public License je licence pro svobodný software
LPGL	Lesser General Public License je upravená licence GNU pro svobodný software
P2P	Peer-to-peer je označení typu počítačových sítí, kdy spolu komunikují přímo jednotliví klienti
SMS	Short message service je služba krátkých textových zpráv dostupné na většině mobilních telefonů
StAX	Sun Java Streaming XML je XML knihovna jazyka Java
UML	Unified Modelling Language je grafický jazyk
XML	Extensible markup language je obecný značkovací jazyk

Literatura

- [BJÖ-CCC] Björn b.: Chaos, cheating and cooperation: potential solutions to the Prisoner's Dilemma, Theodor-Boveri-Institut für Biowissenschaften, Würzburg, Germany, 1996.
- [HARI-ISN] Hanneman R., Riddle M: Introduction to Social Networks, Department of Sociology, UCLA, Riverside, 2005
- [GOLD] Golbeck J. A.: Computing and Applying Trust in Web-based Social Networks, College Park, Diplomawork, 2005.
- [EITRU] Kamvar S. D., Schlosser M. T., Garcia-Molina H.: The EigenTrust Algorithm for Reputation Management in P2P Networks, Stanford University, 2002.
- [PETPV] Petermann F.: Psychologie des Vertrauens, Göttingen: Hogrefe, 3. Auflage, 1996.
- [MARSH] Marsh S. P.: Formalising Trust as a Computational Concept, Department of Computing Science and Mathematics, University of Stirling, 1994.
- [ABRAH] Abdul-Rahman A., Hailes S.: Supporting Trust in Virtual Communities, Department of Computer Science, University College London, 2000.
- [TH-VKP] Thomas A.: Vertrauen im interkulturellen Kontext aus Sicht der Psychologie, Göttingen: Vandenhoeck & Ruprecht, 2008.
- [DEMOT] Almenarez F., Marín A., Díaz D., Sanchez J.: Developing a Model for Trust, Management in Pervasive Devices, in proceedings of PerSec 2006 (Third IEEE International Workshop on Pervasive Computing and Communication Security Held in conjunction with IEEE PerCom 2006), Pisa, Italy, 2006.
- [NET-MD] Netrvalová A.: Modelování důvěry, disertační práce, Katedra Informatiky a výpočetní techniky fakulty aplikovaných věd Západočeské univerzity v Plzni, 2009.
- [NET-MSTE] Netrvalová A.: Modelling and Simulation of Trust Evolution, The State of the Art and Concept of PhD. Thesis, Technical Report No. DCSE/TR-2006-02, University of West Bohemia, Pilsen, 2006.
- [HAV-MS] Havel L.: Modelování sociálních sítí a ovlivňování důvěry, bakalářská práce, Katedra Informatiky a výpočetní techniky fakulty aplikovaných věd Západočeské univerzity v Plzni, 2011.

- [KUB-IAT] Kubík, A.: Inteligentní agenty a tvorba aplikačního software na bázi multiagentových systémů, Computer Press, 2002
- [MCE-TIC] McEliece R.: The Theory of Information and Coding, California Institute of Technology, USA, 2002.
- [URB] Urbánek Š.: Modelling and Simulation of Trust Evolution in Complex Systems, Master Thesis, STU – Faculty of Informatics and Information Technologies, Bratislava, 2004.
- [VAC-BP] Václavek V.: Softwarový systém pro provádění experimentů s modely důvěry, Bakalářská práce, Katedra informatiky, Západočeská univerzita v Plzni, Plzeň, 2010.
- [SOCMO] Social Computing & Cultural Modeling, IEEE Intelligent Systems, IEEE Computer Society, July/August 2011, www.computer.org/intelligent.
- [TRCO] Trustworthy Computing, IEEE Security & Privacy, Building Dependability, Reliability and Trust, IEEE Computer Society, July/August 2011, www.computer.org/security.

Online zdroje

- [DBOOK] Dealb%k
URL: <<http://dealbook.nytimes.com/>>
[citováno 26. února 2012]
- [HEUR] Heuréka
URL: <<http://www.heureka.cz/>>
[citováno 13. března 2012]
- [NCGIA] NCGIA Core Curriculum in Geographic Information Science
URL: <<http://www.therealtimereport.com/>>
[citováno 1. dubna 2012]
- [BORD] Bordalier Institute
URL: <<http://www.bordalierinstitute.com/>>
[citováno 5. Března 2012]
- [JADE] Java Agent Development Framework
URL: <<http://jade.tilab.com/>>
[citováno 3. dubna 2012]
- [RETS] Intelligent Software Agents
URL: <http://www.cs.cmu.edu/~softagents/retsina_agent_arch.html/>
[citováno 3. dubna 2012]
- [SWA] Swarm Development Group
URL: <<http://www.swarm.org/>>
[citováno 3. dubna 2012]
- [FRDIC-TR] The Free Dictionary by Farlex
URL: <<http://www.thefreedictionary.com/trust>>
[citováno 27. února 2012]
- [FRDIC-SN] The Free Dictionary by Farlex
URL: <<http://encyclopedia.thefreedictionary.com/Social+network/>>
[citováno 15. března 2012]

Přílohy

Struktura výstupních souborů:

Struktura výstupního textového a XML souboru třídy Vystup.java popsané v kapitole 5.3.9. vypadá takto:

```

VYSLEDKY
Matice duvery          Sit akteru          Subjekty
-1  1  1  1          jmeno subjektu: A
0. iterace            1 -1  1  1          identifikator: 0
-   0.400 0.400 0.200 1  1 -1  1          reputace: 0.8
0.200 -   0.600 0.080 1  1  1 -1
0.200 0.600 -   0.360
0.200 0.160 0.240 -   Matice reputace          dispozice k duvere: 0.600 0.600 0.700
0.200 0.160 0.240 -   0. iterace          vahovy koeficient poctu doporuzeni: 0.6
1. iterace            -   0.800 0.800 0.600          vahovy koeficient reputace: 0.2
-   0.409 0.409 0.203 0.800 -   0.800 0.200          vahovy koeficient dispozice: 0.5
0.230 -   0.654 0.084 0.800 1.000 -   0.600
0.239 0.669 -   0.374 0.800 0.400 0.600 -
0.218 0.169 0.263 -
    
```

Obrázek 24 - Ukázka výstupního textové souboru

```

<?xml version="1.0" encoding="UTF-8"?>
<!--xml_vystup_-_duvera_v_socialni_siti-->
- <duvera_v_siti>
  - <subjekt jmeno="A" id="0">
    - <duvera>
      - <iterace cislo="0">
        <k_subjektu jmeno="A">-1.0</k_subjektu>
        <k_subjektu jmeno="B">0.40908490792468477</k_subjektu>
        <k_subjektu jmeno="C">0.40908490792468477</k_subjektu>
        <k_subjektu jmeno="D">0.202632</k_subjektu>
      </iterace>
      - <iterace cislo="1">
        <k_subjektu jmeno="A">-1.0</k_subjektu>
        <k_subjektu jmeno="B">0.4198426880808112</k_subjektu>
        <k_subjektu jmeno="C">0.4186738087695054</k_subjektu>
        <k_subjektu jmeno="D">0.20619169267843646</k_subjektu>
      </iterace>
    </duvera>
  </subjekt>
    
```

Obrázek 25 - Ukázka výstupního XML souboru

Vstupní dotazník:

Jméno:

Jste spokojen se svou momentální životní situací (bydlení, finance, práce, politika, atd.)?

- | | |
|----------------------------|-----|
| a) Vůbec nejsem spokojen | 0.0 |
| b) Jsem nespokojen | 0.2 |
| c) Spíše jsem nespokojen | 0.4 |
| d) Spíše jsem spokojen | 0.6 |
| e) Jsem spokojen | 0.8 |
| f) Jsem absolutně spokojen | 1.0 |

Ovlivňuje Vaše nálada Vaše rozhodování?

- | | |
|---|-----|
| a) Vůbec ne – je jedno zda jsem rozčilený nebo úplně šťastný, stále se rozhoduji stejně | 0.0 |
| b) Ne – jsou výjimeční chvíle, kdy tomu je jinak | 0.2 |
| c) Spíše ne – většinou moje nálada neovlivňuje moje rozhodnutí | 0.4 |
| d) Spíše ano – většinou moje nálada ovlivňuje moje rozhodnutí | 0.6 |
| e) Ano – jsou výjimečné chvíle, kdy tomu tak není | 0.8 |
| f) Rozhodně ano – když jsem veselý, říkám ano, když jsem rozčilený/smutný, říkám ne | 1.0 |

Dáváte při rozhodování a řešení obtížných i každodenních problémů na svou intuici a vnitřní hlas?

- | | |
|-----------------|-----|
| a) Nikdy | 0.0 |
| b) Většinou ne | 0.2 |
| c) Spíše ne | 0.4 |
| d) Spíše ano | 0.6 |
| e) Většinou ano | 0.8 |
| f) Vždy | 1.0 |
-

Jak moc je pro Vás důležitá pověst někoho jiného? Když o někom uslyšíte jen špatné/dobré věci, ovlivní to Váš úsudek na něj, i když jste ho předtím vůbec neznali?

- a) Reputace pro mě není vůbec důležitá, nezajímám se, co se o kom povídá 0.0
- b) Reputace neovlivní mé smýšlení o druhých, nejsem zaujatý před naším setkáním, snažím se nemyslet na to, co jsem slyšel 0.2
- c) Reputace spíše neovlivní mé smýšlení o druhých, nejsem zaujatý před naším setkáním, ale pamatuji na to, co o druhém vím 0.4
- d) Reputace spíše ovlivní mé smýšlení o druhých, jsem trochu zaujatý před naším prvním setkáním, mám se na pozoru 0.6
- e) Reputace ovlivní mé smýšlení o druhých, napovídá mi, jaký člověk je a co si o něm mám myslet 0.8
- f) Reputace je velmi důležitá, sám si na ní hodně zakládám 1.0

Jak moc Vám záleží na doporučení výrobku, služby, pracovníka nebo úplně cizího člověka od Vašich známých a přátel?

- a) Nepotřebuji od nikoho rady a doporučení, řídím se výhradně vlastním rozumem 0.0
- b) Většinou se řídím vlastním rozumem a výjimečně dám na radu či doporučení svých přátel 0.2
- c) Většinou se řídím vlastním rozumem, rady a doporučení přátel mi můžou pomoci, když se nemohu rozhodnout 0.4
- d) Většinou dávám na rady a doporučení svých přátel, i když si nejsem úplně jistý, zda bych se tak rozhodl sám 0.6
- e) Vždy dávám na rady a doporučení svých přátel, pokud mají nějakou zkušenost, řešili podobný problém, atd. 0.8
- f) Vždy dávám na rady a doporučení svých přátel, nezajímá mě nic jiného 1.0

Otázky na ostatní účastníky facebookového diskuzního kroužku

Jak moc znáte XYZ?

- a) Vůbec se neznáme – nikdy jsme se neseťkali 0.0

- | | |
|--|-----|
| b) Spíše ne – setkali jsme se, mluvili jsme spolu, máme společné známé | 0.2 |
| c) Trochu ano – známe se již z nějaké diskuze, setkali jsme se, atd. | 0.4 |
| d) Známe se – občas se vidáme, popovídáme si spolu | 0.6 |
| e) Jsme přátelé – vidáme se pravidelně, bavíme se spolu | 0.8 |
| f) Jsme nejlepší přátelé – řešíme spolu všechno | 1.0 |

Jaké máte zkušenosti s XYZ (podle toho jak se znáte, podle facebookového účtu, podle prvního dojmu)?

- | | |
|--|-----|
| a) Velmi špatné zkušenosti | 0.0 |
| b) Špatné zkušenosti | 0.2 |
| c) Neutrální zkušenosti tíhnoucí spíše k těm špatným | 0.4 |
| d) Neutrální zkušenosti tíhnoucí spíše k těm dobrým | 0.6 |
| e) Dobré zkušenosti | 0.8 |
| f) Velmi dobré zkušenosti | 1.0 |

Dotazník po každé iteraci + přídavek posledního dotazníku:

Jméno: Subjekt A

Jak se poslední dobou máte a jakou máte náladu (týden)?

- | | |
|----------------------------|-----|
| a) Vůbec nejsem spokojen | 0.0 |
| b) Jsem nespokojen | 0.2 |
| c) Spíše jsem nespokojen | 0.4 |
| d) Spíše jsem spokojen | 0.6 |
| e) Jsem spokojen | 0.8 |
| f) Jsem absolutně spokojen | 1.0 |

Jak se Vám poslední dobou daří (týden)?

- | | |
|---|-----|
| a) Na co sáhnu to, zkazím | 0.0 |
| b) Většinou mi to nevychází | 0.2 |
| c) Sem tam něco vyjde | 0.4 |
| d) Sem tam něco nevyjde | 0.6 |
| e) Většinou mi vše vychází | 0.8 |
| f) Všechno, co jsem udělal, se vydařilo | 1.0 |

Jak byste ohodnotil XYZ podle předešlé diskuze? Známkujte od 1 do 10, 1 značí nejhorší a 10 nejlepší známku.

Subjekt B –

Subjekt C –

Jak byste ohodnotil XYZ za celou dobu, co jste s ní/m strávil ve skupině Modelování důvěry? Známkujte od 1 do 10, 1 značí nejhorší a 10 nejlepší známku.

Subjekt B –

Subjekt C –

Uživatelský manuál:

Aplikace je vytvořena v jazyce Java a využívá knihovnu JADE. Pro její bezproblémový běh je zapotřebí verze Javy alespoň 1.6 Update 22, který je ke stažení z oficiálních stránek <http://java.com/en/download/index.jsp> a knihovnu JADE verze 4.1.1, kterou lze nalézt na domovské stránce <http://jade.tilab.com>.

Adresářová struktura je následující:

- bin – přeložené a spustitelné soubory
- data – dotazníky
- diploma – tento dokument v elektronické podobě
- doc – javadoc dokumentace
- examples – vstupní soubory experimentů a jejich výsledky
- src – zdrojové kódy aplikace

Ke spuštění není potřeba žádné instalace, jen je nutné nastavit správně cestu k JADE knihovně. Proměnná CLASSPATH musí obsahovat lokální adresář a knihovny nástroje JADE. Nastavení proměnné se provede příkazem:

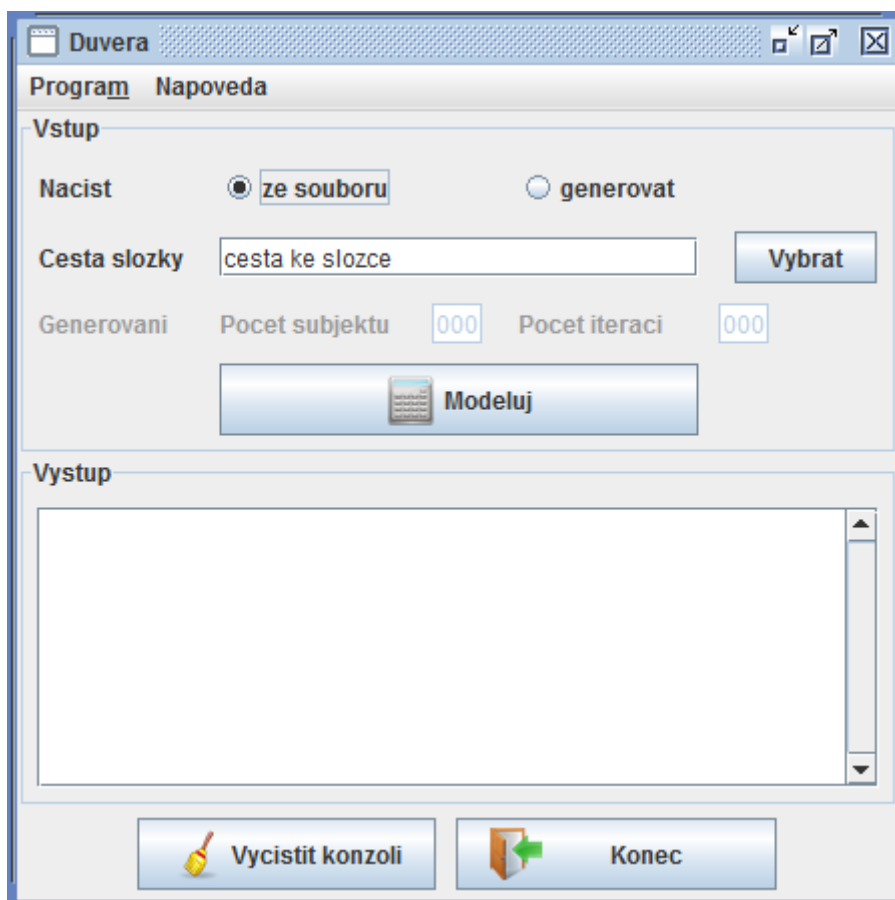
```
set CLASSPATH=%CLASSPATH%;.;c:\jade\lib\jade.jar;  
c:\jade\lib\commons-codec\commons-codec-1.3.jar
```

Aplikaci je potom možné spustit dvojklikem na soubor run_exp1.exe pro první část experimentu, run_exp2.exe pro druhou část experimentu a run_exp3.exe pro třetí část experimentu. V případě běhu s jiným počtem agentů je nutné si všechny agenty ručně spustit.

Po spuštění aplikace se zobrazí hlavní okno aplikace zobrazené na obrázku 26. Ovládání je velice intuitivní a pro obyčejného uživatele by nemělo představovat žádný problém. Přehled činností:

- Načtení dat – výběr jedné z možností, buď náhodné generování, nebo načtení z textového souboru

- Modelování – při stisku tlačítka Modeluj se spustí výpočet aplikace, na obrazovce se vypíše výsledky a vytvoří se textový a XML soubor s výsledky
- Vyčistit konzoli – vyčistí textové okno aplikace
- Změna jazyka – změní jazyk na anglický nebo český dle výběru
- O programu – vypíše informace o programu
- Konec – ukončí program



Obrázek 26 - Grafické rozhraní aplikace