

Jazyky pro reprezentaci znalostí (srovnání
formálních logických systémů)

Lukáš Grulich

21. února 2006

Obsah

1 Úvod	3
2 Předpokládané využití – příklad Sugarscope	3
3 Reprezentace znalostí	4
3.1 Principy reprezentace znalostí	4
3.2 Výroková logika	5
3.2.1 Příklad	5
3.2.2 Metody odvozování ve VL	5
3.3 Predikátová logika prvního řádu (PL1)	6
3.3.1 Příklad	6
3.3.2 Metody odvozování ve PL1	7
4 Zaklady transparentní intenzionální logiky	7
4.1 Algoritmicky nestrukturované (tj. entity prvního řádu)	8
4.2 Algoritmicky strukturované (entity vyšších řádů)	9
4.3 Příklad	9
5 Inference v TIL	10
6 Závěr	11

1 Úvod

Tato práce se věnuje jazykům využívaným pro reprezentaci znalostí v oblasti umělé inteligence, a to takovým, které je možné (a vhodné) využívat pro vytváření modelů socio-ekonomických systémů. Z hlediska teorie modelování, bude takový systém vytvořen na základě multiagentního systému (MAS) – tj. interakcí mnoha subjektů různých typů vybavených identitou, znalostmi a rozhodováním (individua, společnosti, agregovaná individua, ...) (více v [12]). Předkládaná práce se zaměřuje na srovnání reprezentace znalostí pomocí klasické výrokové a predikátové logiky a téhož, využitím transparentní intenzionální logiky. Tato práce by měla poskytnout základní přehled možností a omezení jednotlivých typů logických formálních systémů, a to na příkladě jednoduchého modelu nazvaného Sugarscope. Mým původním úmyslem bylo nejprve zcela obecně shrnout možnosti reprezentace znalostí pomocí uvedených technik a posléze nastínit principy jejich využití v socio-ekonomických modelech. Nakonec jsem zvolil jinou cestu – na začátku práce představím jednoduchý příklad, na kterém posléze demonstřuji možnosti jednotlivých jazykových systémů. Většinou neuvádím přesné formální definice logických systémů – ty je možné nalézt v literatuře. Práce by tak měla plnit funkci přehledově-srovnávací.

2 Předpokládané využití – příklad Sugarscope

Má disertační práce je zaměřená na modelování socio-ekonomických systémů. V rámci tohoto výzkumu vzniká framework pro tyto modely, který je vystaven na principech multiagentního paradigmatu. Představovaný příklad je jedním z předpokládaných aplikačních využití. Model Sugarscope patří mezi klasiku oblasti umělého života. Modeluje chování jednoduchých subjektů v prostředí, které je tvořeno čtvercovou maticí, jejíž jednotlivá políčka obsahují jídlo (cukr) a to v různém množství. Cílem subjektů (agentů) je samozřejmě přežít, pohybovat se po prostředí a hledat místa s větším množstvím potravy (a malým množstvím konkurentů), ovšem jen tak, aby jim stačily síly (které nabývají jídlem). Do systému je možné zavést subjekty vybavené různými vlastnostmi a sledovat jejich vzájemné interakce, atd. (v pozdější verzi systému byly implementovány atributy kmenů a pozorovány souboje mezi jednotlivými kmeny – strukturovanost umělé společnosti). Více o tomto modelu je možné nalézt např. v [13]. V tomto modelu se mohou tedy vyskytnout tyto znalosti:

- Která políčka obsahují cukr (popř. v jakém množství, eventuálně informace o jiných objektech prostředí)
- Znalosti o ostatních subjektech (soupeřích, partnerech) – na jakých políčkách jsou (popř. informace o nich)
- Informace o vlastních silách, energii

3 Reprezentace znalostí

Reprezentace znalostí je jednou z oblastí Umělé inteligence. Podle [10] se zabývá údaji (daty), které slouží k tvorbě „modelu světa“ (tj. modelu sebe sama a modelu prostředí), ve kterém existují (pracují). Reprezentace znalostí se zabývá zejména *způsobem* jakým mohou být tato data reprezentována a možnostmi jejího využívání pro *inferenci* (odvozování). Jazyk (v naprosto obecném smyslu), určený pro reprezentaci znalostí lze tedy charakterizovat právě kategoriemi *vyjadřovací schopnost jazyka* (tj. jaká fakta umožňuje vyjadřovat) a *schopnost inference* (tedy možnosti a náročnost odvozování nových znalostí z již existujících). V zásadě (a v kontextu této práce) existují tři typy inferencí (více v [15]): *indukce* (zobecnování speciálních případů), *dedukce* (hledání vyplývajících závěrů, tj. hledá se pravá strana implikace) a *abdukce* (hledání předpokladů, tj. hledá se levá strana implikace). V této práci a v uváděných příkladech se zaměřím zejména na dedukci.

Obecné požadavky na reprezentaci znalostí jsou ([10]):

- Modifikovatelnost
- Modularita
- Sémantické sdružování objektů (třídy, hierarchie)
- V některých případech nemonotónost¹

3.1 Principy reprezentace znalostí

Základními používanými technikami pro reprezentaci znalostí jsou (dle [11]):

- Deklarativní: Jde o množinu tvrzení, které vyjadřují poznání nebo dokázaná fakta. Prakticky se používá implementace buď pomocí výrokové/predikátové logiky, nebo pomocí sémantických sítí.
- Procedurální: Jde o popis postupů (procedur), které udávají, jak poznávat, nebo odvozovat (využívání IF-THEN pravidel).
- Rámcová: Jde o hybridní techniku, která integruje deklarativní a procedurální metody.

V této práci se budu zabývat prostředky reprezentace znalostí nejprve pomocí klasické (tj. dvoustavové a extenzionální logiky – konkrétněji výrokové a predikátové) a poté naznačím možnosti využití transparentní intenzionální logiky. Tuto technologii jsem zvolil zejména z těchto důvodů:

- Blízkost přirozenému jazyku (zjednodušený návrh systému, možnost interakcí na úrovni přirozeného jazyka)

¹Schopnost opustit staré znalosti (pravdy) a nahradit je novými. Tedy v monotóním systému musí platit, že pokud $P_1, \dots, P_n \models Z$, pak $P_1, \dots, P_n, P_{n+1} \models Z$, pro libovolnou další premisu P_{n+1} .

- Relativní snadnost implementace pomocí funkcionálního programovacího jazyka
- Velká vyjadřovací schopnost tohoto formalismu, existující hlavně díky jeho nekonečné hierarchii

3.2 Výroková logika

Výrokovou logiku (VL) je možné vnímat jako nauku o vztazích mezi výroky², tj. o kompozici elementárních (atomických) výroků do složitějších – tedy jednak o tom *jak* (jakým způsobem) je možné výroky skládat (syntaxe) a jednak o tom *co* výsledný výrok reprezentuje, čili jaké je jeho ohodnocení. Formalismus výrokové logiky tedy zjednodušuje celý modelovaný svět na charakteristiku jednotlivých výroků (pravdivý/nepravdivý). O syntaxi a sémantice výrokové logiky je možné najít podrobné informace např. v [14], zde jen velmi zjednodušeně: V jazyce VL existují atomické výroky (P, M, S, E, G, ...), logické spojky (\neg , \vee , \wedge , \rightarrow , \leftrightarrow), konstanty (T, F) a pomocné symboly (závorky).

3.2.1 Příklad

Pokud bychom aplikovali reprezentaci znalostí založenou na výrokové logice na příklad Sugarscope, bude výsledný model velmi jednoduchý. Znalosti se budou týkat pouze aktuálního políčka (výrokovou logikou nelze jednoduše vyjadřovat objekty a jejich vlastnosti). Proměnné existující v systému tedy jsou: P (aktuální políčko je vhodné (Appropriate) k usídlení), M (je vhodné přesunout se na vedlejší políčko), S (na aktuálním políčku je dostatečné množství cukru (Sugar)), E (na aktuálním políčku konkurent (Enemy)), G (subjekt má dost energie (Energy) k přesunu o jedno políčko). Znalostní formule mohou být například tyto:

1. $S \wedge \neg E \leftrightarrow P$ (pokud je cukr a není soupeř, je vhodné)
2. $(\neg S \vee E) \wedge G \leftrightarrow M$ (pokud není cukr, nebo je soupeř, a současně je dost energie na přesun, je vhodné se přesunout)

Je evidentní, že samotná výroková logika umožní uchovávat jen několik jednoduchých znalostí – lepší rozlišovací schopnost by bylo možné zavést třeba definováním více proměnných, které by uchovávali informace o soupeřích (např. proměnné E0-E10 by nesly informaci o tom, zda existuje soupeř v dané vzdálenosti od aktuálního políčka), a jim odpovídajících pravidel, ovšem tento způsob by byl velmi netransparentní a komplikovaný.

3.2.2 Metody odvozování ve VL

Mezi několika metodami pro deduktivní inferenci v oblasti umělé inteligence a VL (jako jsou tabulkové a tablové metody) má výsadní postavení zejména

²Za výrok se považuje tvrzení, u kterého má smysl zvažovat jeho pravdivost/nepravdivost

rezoluční metoda, a to pro svou snadnou strojovou implementaci. Je založena na rezolučním pravidle $l \vee C_1, \neg l \vee C_2 \models C_1 \vee C_2$, kde l je literál, C_1, C_2 jsou klauzule VL (tj. formule v konjunktivní normální formě, více v [14], [6]). Klauzule $C_1 \vee C_2$ se nazývá rezolventou. Příkladem může být odvození výroku P (políčko je vhodné k usídlení – první tři klauzule jsou odvozené od znalostní formule č. 2 ve výše uvedeném příkladě):

1. $\neg S \vee E \vee P$
2. $\neg P \vee S$
3. $\neg P \vee \neg E$
4. S (na políčku je cukr)
5. E (na políčku je nepřítel)
6. $\neg P$ (odvozeno z 3 a 5)
7. $S \vee P$ (odvozeno z 1 a 4)

Podařilo se tedy odvodit $\neg P$, čili nevhodnost pro osídlení (z důvodu obsazení políčka soupeřem – klauzule 5). V aplikační praxi má větší smysl tzv. nepřímá rezoluce, kdy do množiny klauzulí vložíme negaci $\neg C$ klauzule C , kterou chceme dokázat – pokud se podaří odvodit prázdnou klauzuli (tj. vznikne spor), klauzule C je dokázaná.

3.3 Predikátová logika prvního řádu (PL1)

Predikátová logika je formalismus, který je schopen reprezentovat výrazy pro znalosti v mnoha oblastech. Predikátová logika vytváří model světa (který popisuje) pomocí objektů, které mohou mít různé atributy (predikáty), a vztahů mezi nimi. Jazyk PL1 obsahuje navíc oproti VL symboly pro individuové proměnné a konstanty ($f, 0, \dots$), funkční a predikátové symboly (a, s, e, \dots), kvantifikátory (\forall, \exists) a pomocný symbol „ \Leftarrow “.

3.3.1 Příklad

Predikátová logika umožňuje mnohem sofistikovanější implementaci znalostí o cukru a o soupeřích. V systému mohou existovat objekty, které reprezentují jednotlivá políčka (F_0 - F_n), nad kterými definujeme predikáty a (aktuální pozice), p (vhodnost pro usídlení), $m(f_1)$ (je vhodné se přesunout na políčko f_1), s (cukr), e (Enemy) a $g(i)$ (energie na překonání i políček) a funkční symbol $Dist(j)$ (vzdálenost j -tého políčka od aktuálního). Pravidla pak mohou být takováto:

- $a(f) \wedge s(f) \wedge \neg e(f) \leftrightarrow p(f)$ (pokud je cukr a není soupeř, je vhodné)
- $a(f) \wedge (\neg s(f) \vee e(f)) \wedge \exists f_1 (\neg e(f_1) \wedge g(Dist(f_1))) \leftrightarrow \neg m(f_1)$ (pokud není cukr, nebo je soupeř, a současně je dost energie na přesun, je vhodné se přesunout)

3.3.2 Metody odvozování ve PL1

Stejně jako ve výrokové logice je i zde možné efektivně použít rozšířenou rezoluční odvozování, byť je to již komplikovanější proces než ve VL. Formule musí být uvedeny v prenexní klauzulární normální formě (tj. skolemizovaná v prenexní formě s jádrem v KNF, více v [14])

Jak je vidět, predikátová logika umožňuje vyjádřit (reprezentovat) velmi pestrou paletu jevů a dějů. Ovšem i zde je možné narazit (v případě zdokonalování a rozšiřování vytvářeného umělého světa) na nepříjemná omezení. Tato omezení mohou být temporální (těžko jde například vyjádřit, že políčko bude volné ještě x časových okamžiků), subjektivě-strukturní (není možné vyjádřit, že jiný subjekt „komunikoval“ informace o volnosti políčka) a statkově-strukturní³ (jde o strukturu prostředí – například jídlem může být jak cukr, tak i obilí; v navrhovaném modelu je navíc primárním konceptem *Potřeba*, což je *naprosto abstraktní pojem, který může zahrnovat jídlo, spánek, komunikaci, atd.*). Navíc, predikátová logika umožňuje sice postihnout, která políčka jsou vhodná pro usídlení, ale nelze jednoduše rozhodnout, které políčko je pro nastěhování výhodnější (což může být v tomto případě např. to, které je blíže aktuální pozici) – chybí zde tedy hodnotící nástroj, který by tak musel být implementován mimo vlastní inferenční mechanismus. Pro základní verzi verzi modelu Sugrascope (tak jak byl uveden) ovšem predikátová logika postačuje.

4 Zaklady transparentní intenzionální logiky

Transparentní intenzionální logika patří mezi logiky neklasické⁴. Jejím zakladatelem je Pavel Tichý. V dnešní době se jí nejvíce využívá v oblasti logické analýzy přirozeného jazyka. Mezi její základní charakteristiky patří (více v [2], [3], [4], [5], [9]):

1. Realismus: Výrazy jazyka vyjadřují mimojazykové entity.
2. Parmenidův princip: Nelze mluvit o něčem, aniž bychom to nějak označili, nebo zmínili.
3. Princip kompozicionality: Význam složeného výrazu je funkcí významů jeho složek.
4. Anti-kontextualismus: Význam výrazu nezávisí na kontextu, ve kterém je užít.
5. Funkcionální princip: Abstraktní i konkrétní entity jazyk TIL vyjadřuje pomocí (parciálních) funkcí.
6. Význam je (algoritmicky) strukturován

³Statek je zde uveden v ekonomickém slova smyslu, v souladu s koncepcí uvedenou v [12]. Jde tedy o objekty, které tvoří prostředí.

⁴Za klasickou je považována taková logika, která je a) dvoustavová a b) extensionální (tj. každý výraz odkazuje na existující objekt)

7. Existence dvouúrovňové sémantiky: Mezi výrazem a označenou entitu je mezistupeň, který definuje způsob, jakým se výraz na danou entitu odvolává (smysl).
8. Využití sémantiky možných světů: TIL využívá tzv. sémantiku možných světů, tj. existence různých stavů světa (resp. logického stavového prostoru), o kterém výrazy jazyka vypovídají a tedy určitou míru nezávislosti logických mechanismů na oněch stavech.
9. Nekonečná hierarchie entit TIL a s ní související úroveň (algoritmické) typovosti výrazů jazyka
10. Jako aparát konstrukcí TIL slouží modifikovaný λ -kalkul (to souvisí s jejím funkcionálním zaměřením).

Ve zmíněné nekonečné hierarchii entit rozlišujeme v zásadě dva typy:

4.1 Algoritmicky nestrukturované (tj. entity prvního řádu)

Jde o entity prvního řádu, jsou jimi *atomické* (bázové) a *molekulární* entity. Celá hierarchie je definovaná nad určitou bází daných symbolů jazyka. Pro analýzu přirozeného jazyka, i pro potřeby reprezentace znalostí jednotlivých subjektů v MAS se zdá být vhodné využití této bázi ⁵:

- Pravivostní hodnoty $o = \{True, False\}$
- Individua jazyka ι (univerzum)
- Množina časových okamžiků / reálných čísel τ
- Množina možných světů ω

Nad touto bází atomických prvků jsou definovány molekulární entity, jimiž jsou parciální funkce typu $\alpha_1 x \dots x \alpha_n \rightarrow \beta$, kde $\alpha_1, \dots, \alpha_n, \beta$ jsou třídy entit (typy) prvního řádu. Obvykle se používá opačná notace, tj. $(\beta \alpha_1 \dots \alpha_n)$. Nejvýznamnější funkcí tohoto typu je *intenze*, což je funkce z možných světů do chronologií prvků daného typu $\alpha : ((\alpha\tau)\omega)$, zkráceně $\alpha_{\tau\omega}$ ⁶.

Intenze tedy odpovídají empirickým výrazům (závisí na logickém stavu světa, na ohodnocení jednotlivých entit), extenze naopak výrazům, které jsou na stavu světa nezávislé (např. individuové úřady – v našem příkladě například nepřítel). To je i princip zmiňované dvouúrovňové sémantiky – díky existenci/neexistenci

⁵Či spíše – vzhledem k tomu, že výzkum v oblasti využití TIL pro reprezentaci znalostí v MAS není příliš daleko, zdá se být vhodné využití ověřenou konfiguraci, která se využívá v oblasti zpracování přirozeného jazyka

⁶Tento zjednodušený zápis asi pochopitelněji vyjadřuje fakt, že aktuální hodnota prvku α_{now} typu α je funkcí aktuálního světa $\omega_{now} \in \omega$ a aktuálního času $\tau_{now} \in \tau$. V souladu s tím se používá notace pro deklarování typů výrazů tak, že např. *IsEnemyOnField/(oi)* znamená, že konstrukce *IsEnemyOnField* je typu (oi) , tedy vstupní parametr je typu ι (univerzum) a výstupní typu o (boolean); podle další z konvencí rovněž zápis $w \rightarrow \omega$ znamená, že proměnná w konstruuje hodnotu typu ω

intensionálního sestupu (tj. závislosti na wt) je hned zjevné, zda výraz reprezentuje intenzi, nebo extenzi. Současně je takto implementována typová kontrola. Formalismem, který ji implementuje je λ -kalkul s typy (typed lambda calculus) – nedovoluje provést aplikaci abstrakce s nekorektním typem parametru.

4.2 Algoritmicky strukturované (entity vyšších řádů)

Entity řádu n vzniknou konstrukcemi nad typy řádu $n-1$, tj. např. entity druhého řádu vznikají konstrukcemi nad typy řádu 1 (značíme *1). Existují tyto základní typy konstrukcí:

1. Proměnné (x, y, \dots) konstruuji v závislosti na valuaci v , tj. v -konstruuji.
2. Trivializace 0X konstruuje entitu X beze změny.
3. Uzávěr $[\lambda x_1 \dots x_n C]$ konstruuje funkci typu $(\beta \alpha_1 \dots \alpha_n)$, kde $\alpha_1 \dots \alpha_n$ jsou typy proměnných $x_1 \dots x_n$ a β je typem konstrukce C . Uzávěr vyjadřuje definici funkce (abstrakci).
4. Kompozice (aplikace) $[CX_1 \dots X_n]$ konstruuje hodnotu funkce (konstrukce C) s parametry $X_1 \dots X_n$.

Přesné rekurzivní definice jazyka TIL je možné nalézt v [5].

4.3 Příklad

Jako individuovou komponentu báze pro konstrukce je zvolena množina ι , která obsahuje všechny individuové jazykové koncepty, které mají v tomto prostředí smysl jak konkrétní, tak abstraktní (tj. zejména jednotlivá políčka Field0-FieldN, cukr Sugar, soupeře Enemy, pak vyšší konstrukce EnemyOnField, ValueOfField, ...)⁷ Konstrukce konceptů mohou být například takovéto:

- ${}^0Field0 - FieldN / \iota_{\tau\omega}, {}^0Enemy / \iota_{\tau\omega}, {}^0Sugar / \iota_{\tau\omega}, w \rightarrow \omega, t \rightarrow \tau$
- ${}^0IsOnField / (ou)_{\omega\tau}$
- ${}^0ActualField / (\iota)_{\omega\tau}$
- ${}^0FieldPosition / (\tau\iota)_{\omega\tau}$
- ${}^0AND / (ooo)$ ⁸
- ${}^0NOT / (oo)$
- ${}^0IF / (\iota o \iota)$
- $Distance / (\tau\iota)_{\omega\tau} = \lambda w \lambda t \lambda F_1 \lambda F_2 [{}^0Minus [{}^0FieldPosition_{wt} F_1] [{}^0FieldPosition_{wt} F_2]]$ ⁹

⁷Zde by bylo možné odchýlit se od univerzálnosti a „přirozeno-jazykovosti“ TIL, a zvolit několik individuových tříd – jednu pro políčka, další pro subjekty, atd. Znalostní reprezentace by tak získala silnější typovost a asi i snazší strojovou implementaci, ale na úkor obecnosti a rozšířitelnosti systému.

⁸Toto jsou tradičně chápané abstrakce And, Not, If ... else

⁹Zde se předpokládá existence konceptu (funkce) obvyklém významu

- $EnemyOnField / (ot)_{\omega\tau} = \lambda w \lambda t \lambda F [{}^0IsOnField_{wt} F {}^0Enemy_{wt}]$
- $SugarOnField / (ot)_{wt} = \lambda w \lambda t \lambda F [{}^0IsOnField_{wt} F {}^0Sugar_{wt}]$
- $ValueOfField / (\tau\iota) = \lambda w \lambda t \lambda F [{}^0IF[\lambda w \lambda t \lambda F [{}^0IsOnField_{wt} F {}^0Enemy_{wt}]]_{wt}]$
 ${}^0999[{}^0IF[\lambda w \lambda t \lambda F [{}^0IsOnField_{wt} F {}^0Sugar_{wt}]]_{wt}]_{wt} [(\tau\iota)_{\omega\tau} = \lambda w \lambda t \lambda F_1 \lambda F_2$
 $[{}^0Minus[{}^0FieldPosition_{wt} F_1]_{wt} [{}^0FieldPosition_{wt} F_2]] F_1 {}^0ActualField]_{wt} {}^0999]_{wt}]_{wt}$
₁₀

5 Inference v TIL

Pro vysvětlení inferenčního principu v transparentní intenzionální logice je nutné definovat pojem β – redukce (λ – transformace). Jde v podstatě o provedení funkce (abstrakce) s konkrétním parametrem (známé z λ -kalkulu), přesněji o přechod z konstrukce typu aplikace, na jiný typ. Tedy, nechť C je konstrukce obsahující volné proměnné x_1, \dots, x_n typů $\alpha_1, \dots, \alpha_n$ a C_1, \dots, C_n týchž typů (tedy $\alpha_1, \dots, \alpha_n$). Pak $C(x_1/C_1, \dots, x_n/C_n)$ je výsledná konstrukce, která vznikla z konstrukce C po řadě substitucí konstrukcí C_1, \dots, C_n za proměnné x_1, \dots, x_n (samozřejmě nesmí dojít ke kolizi proměnných)¹¹. Příkladem může být odvození hodnoty políčka (toto ohodnocení je dáno v podstatě vzdáleností daného políčka F od aktuálního, pakliže na něm není soupeř, a je na něm cukr):

1. $[\lambda w \lambda t [{}^0IsOnField_{wt} {}^0Field5_{wt} {}^0Enemy_{wt}]_{wt}]_{wt} = False$ – na políčku Field5 není soupeř
2. $[\lambda w \lambda t [{}^0IsOnField_{wt} {}^0Field5_{wt} {}^0Sugar_{wt}]_{wt}]_{wt} = True$ – na políčku Field5 je cukr
3. $[\lambda w \lambda t [{}^0FieldPosition_{wt} {}^0Field5_{wt}]_{wt}]_{wt} = 5$ – pozice políčka Field5 je 5
4. $[\lambda w \lambda t [{}^0FieldPosition_{wt} {}^0ActualField_{wt}]_{wt}]_{wt} = 2$ – pozice aktuálního políčka je 2
5. $[ValueOfField {}^0Field5_{wt}] = \lambda w \lambda t \lambda F [{}^0IF[\lambda w \lambda t \lambda F [{}^0IsOnField_{wt} F {}^0Enemy_{wt}]]$
 ${}^0Field5_{wt}]_{wt}] {}^0999[{}^0IF[\lambda w \lambda t \lambda F [{}^0IsOnField_{wt} F {}^0Sugar_{wt}]]_{wt} {}^0Field5_{wt}]_{wt}$
 $[(\tau\iota)_{\omega\tau} = \lambda w \lambda t \lambda F_1 \lambda F_2 [{}^0Minus [{}^0FieldPosition_{wt} F_1]_{wt}$
 $[{}^0FieldPosition_{wt} F_2]] {}^0Field5_{wt} {}^0ActualField]_{wt} {}^0999]_{wt}]_{wt}$
 $\longrightarrow \beta + ekvivalence\ 3\ a\ 4$
6. $[ValueOfField {}^0Field5_{wt}] = \lambda w \lambda t [{}^0IF[[]^0IsOnField_{wt} {}^0Field5_{wt} {}^0Enemy_{wt}]_{wt}]_{wt}$
 ${}^0999 [{}^0IF [[]^0IsOnField_{wt} {}^0Field5_{wt} {}^0Sugar_{wt}]_{wt}$
 ${}^0Field5_{wt}]_{wt} [[{}^0Minus {}^05 {}^02] {}^0ActualField]_{wt} {}^0999]_{wt}]_{wt}$
 $\longrightarrow \beta + ekvivalenc\ 1\ a\ 2$

¹⁰Konstanta 0999 v tomto případě značí „zarážku“, tj. velké ohodnocení políčka v případě že na něm není cukr, popř. na něm je soupeř

¹¹Už některé složitější konstrukce, které byly zmíněny v uvedeném příkladě, použily β -redukci (z důvodu zjednodušení)

7. $[ValueOfField^0 Field5_{wt}] = [^0 IF False^0 999 [^0 IF True^0 3^0 999]_{wt}]_{wt}$
8. $[ValueOfField^0 Field5_{wt}] = ^0 3$

Výsledné ohodnocení políčka $^0 Field5$ je tedy $^0 3$. Jak je vidět, konstrukce (a tedy i reprezentované znalosti) mohou být velmi bohaté a je tedy možné pomocí mechanismů TIL implementovat i ty části, které by při využití klasické logiky musely být mimo logickou inferenci.

6 Závěr

V práci jsem naznačil možnosti reprezentace znalostí pomocí tří typů logických systémů. Příklady byly úmyslně voleny mírně odlišné, aby vynikly rozdíly mezi jednotlivými jazyky. Transparentní intenzionální logika je jistě mocným nástrojem pro analýzu přirozeného jazyka – v těchto intencích se zatím pohyboval výzkum jejích aplikačních možností. Její využití v oblasti reprezentace znalostí (nutno podotknout, že v uvedené kategorizaci – deklarativní a procedurální – stojí tato technika na hranici obou) pro multiagentní systémy zatím není důkladně zpracováno. Tato práce se pokusila naznačit možné směřování k využití jejího potenciálu pro modelování socio-ekonomických systémů pomocí multiagentních systémů. Věřím, že právě v multiagentních systémech, kde hraje významnou roli komunikace mezi subjekty (agenty, hráči)¹², struktura jejich vnějšího (prostředí) a vnitřního (znalosti) světa, může najít možnosti uplatnění.

Reference

- [1] Hadacz, Horák. Knowledge representation and reasoning with TIL. In Knowledge-based software engineering. IOS Press. 2000.
- [2] Duží, Materna. Constructions. Online:
http://til.phil.muni.cz/text/constructions_duzi_materna.pdf
- [3] Duží, Materna. Fundamentals of Transparent Intensional Logic (TIL) and Procedural Theory of Concepts. Online:
<http://labis.vsb.cz/labib/files/til.pps>
- [4] Duží, Marie. Jazyk a pojmy.
Online: <http://www.cs.vsb.cz/duzi/Jazyk>
- [5] Duží, Marie. Principy logické analýzy jazyka. Online:
<http://www.cs.vsb.cz/duzi/LogickaAnalyza.doc>.

¹²Modely tohoto typu leží na rozhraní teorie her (pokud možno vítězit nad soupeři), multiagentních systémů (částečná spolupráce mezi subjekty) a kognitivní vědy (princip emergence), proto je komplikované volit jednoznačnou terminologii

- [6] Duží, Marie. Matematická logika.
Online: <http://www.cs.vsb.cz/duzi/Matlogika.doc>.
- [7] Hajičová, Sgall. Linguistic meaning and knowledge representation in automatic understanding of natural language.
online: <http://acl.ldc.upenn.edu/C/C80/C80-1011.pdf>
- [8] Chrz, Tomáš. Application of intensional logic to knowledge representation.
Online: <http://acl.ldc.upenn.edu/C/C82/C82-2015.pdf>
- [9] Peregrin, Jaroslav. Úvod do teoretické sémantiky.
- [10] Štěpánková, Olga: Reprezentace znalostí. (Studijní materiály).
Online: http://cyber.felk.cvut.cz/gerstner/teaching/xkui/sbirka/XKUI04_RepZn.ppt.
- [11] Matiašová, Anděla. Reprezentace znalostí v umělé inteligenci. Ústav informatiky PEF MZLU. Online:
http://old.mendelu.cz/veda/mendelnet/pef/mendelnet1999/MATIASOVA_ANDELA/
- [12] Grulich, Lukáš. Model socio-ekonomického systému. Interní materiál FIT - pracovní verze.
- [13] Jager, Wander. Modelling Consumer Behaviour. Phd thesis. Groningen. 2000. Online:
<http://dissertations.ub.rug.nl/FILES/faculties/ppsw/2000/w.jager/thesis.pdf>
- [14] Lukasová, Alena. Logické základy umělé inteligence. 1. Výroková a predikátová logika. Ostrava. 1999.
- [15] Dvořák, Pavel. Úvod do inference. Bakalářská práce. FI MU. 2002.