

Posudek disertační práce
RNDr. Pavel Surýnek. Constraint Programming in Planning

Předložená disertační práce se věnuje řešení velice aktuálních a těžkých problémů umělé inteligence z oblasti plánování a splnitelnosti pomocí metody Programování s omezujícími podmínkami (nebo metod a heuristik jimi motivovanými). Plánovací problémy jsou zadány jako počátečné a (kýžené) koncové pozice různých zdrojů a agentů a řešení je ve formě posloupnosti akcí (pokud existuje). Metoda řešení omezení se v této práci aplikuje na odvozený graf úlohy (planning graph).

Práce je rozdělena do 5 kapitol, 2 příloh a obsahuje jedno CD. Po úvodní kapitole následuje kapitola „Classical Planning and Constraint Programming“, která podává obecný úvod do klasického plánování a různé formalismy, přehled stávajících technik a speciálně „plánovací grafy“ a algoritmy nad nimi (GraphPlan algorithm). Závěr kapitoly je věnován technikám řešení omezení. Tato kapitola je velice pěkným přehledem.

Třetí kapitolu tvoří hlavně původní autorovy výsledky s vlastními přínosy k řešení plánování s pomocí plánovacích grafů. Různé dílčí kroky jsou s použitím metod z Programování s omezujícími podmínkami nejdřív analyzovány teoreticky a následně i experimentálně (různé varianty a taky heuristiky). Experimenty byly vykonány na čtyřech typech úloh (zobecněné Hanojské věže, jeřáby, letadla a ještě problém zákaznického software v programovatelných obvodech) a prokázaly významné závislosti na zvolených heuristikách. Poznamenejme, že výsledky nebyly srovnány se stávajícími řešeními z plánovací komunity, protože cílem práce bylo prokázat, že techniky z Programování s omezujícími podmínkami (např. „hranové konzistence (tři varianty)“) výrazně zlepšila Graph plan algoritmus (vlastní implementace známého řešení). Následně autor zavedl a otestoval novou metodu (globální) projekční konzistence, která ještě zlepšila výsledky, hlavně pro strukturálně těžké problémy. Na závěr autor specifikoval třídu problémů, pro které je časová složitost řešení polynomiální.

Čtvrtá kapitola obsahuje původní výsledky autora, kde předešlé heuristiky jsou použity na řešení (těžkých) problémů splnitelnosti (převedených do grafů). Tady už řešení jsou srovnána i se stávajícími produkty a alespoň pro odhalení nesplnitelnosti jsou výrazně lepší (to navozuje možnost použít autorovu a jinou metodu paralelně – co si o tom myslí, by mně zajímalo).

Autor předložil originální řešení problémů plánování pomocí plánovacího grafu a splnitelnosti: tři varianty řešení pomocí hranové konzistence (metoda přenesená z Programování s omezujícími podmínkami) a novou metodu projekční konzistence. Výsledky byly publikovány ve vícero pracích (samostatných i se spoluautory), nejvýznamnější 1xIEEE, 3xAAAI, 4xLNCS a dalších 7 prací na významných zahraničních akcích komunity (už má 6 záznamů v DBLP).

Software jsem si nechal předvést a s autorovou pomocí jsem si mohl vychutnat několik experimentů. Autor dosáhl nové netriviální výsledky v oblasti teorie. Nicméně já si více cením výsledků implementačních a experimentálních. Autor skutečně naprogramoval spoustu software tak, že se s tím dá experimentovat (po konstrukci vhodného uživatelského rozhraní by si s tímto software určitě spousta lidí ráda experimentovala). Zmíním alespoň některé grafové algoritmy autora, které pro NP těžké problémy dovedl najít heuristiky na přibližné řešení, které činí řešení problému reálné.

Celá práce je krásným přínosem k synergii teoretické a experimentální informatiky a ukazuje, že i v informatice se může uplatnit cyklus poznávacích metodologií střídajících teorií a experiment, jak je tomu běžně třeba ve fyzice.

Nemohu úplně souhlasit s tím, že řešení přispívají k racionálnímu uvažování (typicky lidskému postupu při řešení), viz úvahy na str. 2. Pořád jsme od toho daleko a pořád výkon počítačů nahrazuje to, co z lidského způsobu řešení problémů neumíme ani modelovat.

Zajímalo by mne, jak se vyvíjí složitost pro jednotlivé úlohy (autorovy diagramy míchající výkony na všech typech problémů nejsou zrovna přehledné). Dalo by se alespoň orientačně uvést, jaké výsledky na daných problémech dosahují špičkové obecné (ne specifické pro daný problém) plánovací algoritmy (ne nutně řešící převod problému na grafový plán) a jaké jsou možnosti vylepšení těchto plánovačů z hlediska výsledků práce (nebylo by vhodné problém zadat už ve formátu usnadňujícím tvorbu grafu a ne v pddl?). Jak vypadají perspektivy, když známé řešení se musí kvůli malé změně (zavřela se brána na letišti, vypadl kamion,...) přepočítat, musí se problém řešit od začátku, nebo lze už část řešení použít? Taky by mne zajímalo, kdyby se uchazeč vyjádřil k tomu, nakolik grafové plánování je vhodné pro odhalování konfliktů v reálném světě (čas a 3D prostor – kde na jednom místě a v jednom čase může být jen jeden objekt) a jaké metody by byly vhodné pro virtuální světy (plánování paralelních/distribovaných procesů, grid computing, ...).

Na závěr konstatuji, že autor jednoznačně prokázal, že zvládl metody vědecké práce a dosáhl vícero původních vědeckých výsledků. Jednoznačně doporučuji, aby byl připuštěn k obhajobě a po jejím úspěšném průběhu aby mu byl udělen titul PhD.

V Praze 2.8.2008



l

c.