Sławomir Lasota
Wydział Matematyki, Informatyki i Mechaniki
Uniwersytet Warszawski
e-mail: sl@mimuw.edu.pl

Warszawa, 27 marca 2015

Opinia o rozprawie doktorskiej Pana Piotra Witkowskiego

Zawartość rozprawy

Rozprawa poświęcona jest rozstrzygalności i złożoności obliczeniowej różnych rozszerzeń logiki $\mathcal{C}^2$, która z kolei jest rozszerzeniem logiki pierwszego rzędu z dwoma zmiennymi (FO2) o kwantyfikatory zliczające. W wielkim uproszczeniu, rozprawa koncentruje się na logice $\mathcal{C}^2$ + Datalog, czyli na połączeniu dwóch formalizmów: logiki $\mathcal{C}^2$, i programów (zapytań) w Datalogu. Faktyczna sytuacja jest jednak dużo bardziej skomplikowana.

Po pierwsze, autor rozprawy rozważa trzy logiki (w kolejności od najsilniejszej do najsłabszej): $\mathcal{C}^2$, FO2, oraz uniwersalny fragment FO z dwoma zmiennymi $\forall\forall$. Po drugie, w rozprawie rozważa się tylko tzw. monadyczny Datalog, w którym wszystkie predykaty definiowane (ang. *intensional*) są unarne. Po trzecie, w semantyce logiki $\mathcal{C}^2$ + Datalog niejawnie zakłada się dodatkowe warunki semantyczne, oznaczane w pracy symbolami „FUN" (ang. *functionality*), „BIR" (ang. *bounded intersection freeness*) i „BSR" (ang. *bounded non-sharing*). Po czwarte, w rozprawie rozważa się przypadek nieco ogólniejszy, gdzie zamiast pojedynczego programu jest wiele osobnych programów w Datalogu, a trzy powyższe warunki semantyczne odnoszą są do każdego z programów osobno. Po piąte, interakcja pomiędzy formułą $\mathcal{C}^2$ a programem w Datalogu jest ograniczona (a dokładniej, ogranicza się dozwolone wystąpienia symboli definiowanych przez Datalog): w pracy rozważane są trzy (kojelno coraz słabsze) ograniczenia, oznaczane symbolami q, r i $r2$. W szczególności, ograniczenie $r2$ jest słabsze niż r ponieważ dotyczy wszystkich programów w Datalogu z wyjątkiem pewnych wskazanych dwóch programów. Przy czym pierwsze dwa z warunków FUN, BIR i BSR są wyrażalne w logice $\mathcal{C}^2 + r$ definując jej fragmenty, ale już żaden z trzech warunków nie jest wyrażalny w FO2 $+r2$. Po szóste, kombinację logik $\mathcal{C}^2$ + Datalog dodatkowo rozszerza się w rozprawie o nierówności liniowe dotyczące liczby elementów

w strukturze spełniających poszczególne predykaty. Krajobraz definicyjny jest zatem dość skomplikowany, a liczba różnych kombinacji znaczna.

Wymienione powyżej rozszerzenia i modyfikacje logiki $\mathcal{C}^2$ + Datalog mają na celu uzyskanie rozstrzygalności problemu spełnialności, a dodatkowo przyzwoitej złożoności obliczeniowej. Podstawową motywacją autora jest zastosowanie procedury decyzyjnej w tzw. ograniczonej weryfikacji programów, które operują na strukturach wskaźnikowych; weryfikacja ograniczona polega na sprawdzeniu, czy istnieje kontrprzykład (czyli błędne wykonanie programu) ograniczonej długości. Logika $\mathcal{C}^2$, lub jej fragment, używana jest do wyrażenia semantyki weryfikowanego programu, a programy w Datalogu do opisania początkowej struktury wskaźnikowej.

Głównymi wynikami rozprawy, omówionymi w rozdziałach 3 i 5, są górne granice dla złożoności problemu spełnialności dla dwóch (nieporównywalnych co do siły wyrazu) logik,

$$\mathcal{C}^2 + \text{Datalog} + r + \{\text{BSR}\} \qquad \text{oraz} \qquad \mathcal{C}^2 + \text{Datalog} + r2 + \{\text{BSR}, \text{BIR}\};$$

w obydwu przypadkach problem spełnialności jest rozstrzygalny w niedeterministycznym czasie wykładniczym. W pierwszym przypadku dowód polega na efektywnym przekształceniu do logiki $\mathcal{C}^2$, której spełnialność jest w NEXPTIME. W drugim przypadku dowód jest znacznie bardziej skomplikowany. Pierwszy krok to przetłumaczenie logiki do rozszerzenia $\mathcal{C}^2$, w którym zakłada się, że pewne dwa wyróżnione symbole binarne są interpretowane w strukturze jako lasy skierowane. Logika na nazywana jest w pracy „$\mathcal{C}^2$ z drzewami" (ang. $\mathcal{C}^2$ *with trees*), podczas gdy bardziej adekwatna byłoby prawdopodobnie nazwa „$\mathcal{C}^2$ z lasami". W drugim kroku autor rozprawy wykazuje, że problem spełnialności dla logiki $\mathcal{C}^2$ z drzewami jest w NEXPTIME. Metoda dowodowa stanowi rozwinięcie dowodu złożoności NEXPTIME dla spełnialności logiki $\mathcal{C}^2$, zaproponowanej przez Iana Pratt-Hartmanna w pracach [77, 78]. Jako wniosek otrzymujemy złożoność NEXPTIME dla wszystkich wariantów logiki $\mathcal{C}^2$ + Datalog w których zakłada się warunek semantyczny BSR (ang. *bounded non-sharing*).

Ponadto, rozdział 6 rozprawy zawiera kilka dowodów dolnych granic. Po pierwsze, wszystkie kombinacje i warianty logiki $\mathcal{C}^2$+Datalog sa NEXPTIME-trudne. Po drugie, problem spełnialności dla logik

$$\forall\forall + \text{Datalog} + q + \{\text{FUN}\} \qquad \text{oraz} \qquad \forall\forall + \text{Datalog} + q + \{\text{FUN}, \text{BIR}\}$$

jest przynajmniej tak trudny jak problem osiągalności w sieciach Petriego (czyli usunięcie warunku semantycznego BSR prowadzi prawdopodobnie do

bardzo wysokiej złożoności). Po trzecie, uniwersalny fragment FO z trzema zmiennymi z Datalogiem, a dokładnie

$$\forall\forall\forall + \text{Datalog} + q + X, \qquad \text{dla dowolnego } X \subseteq \{\text{FUN}, \text{BSR}, \text{BIR}\},$$

jest nierozstrzygalny.

Rozdział 4 rozprawy jest odmienny od pozostałych: poświęcony został ilustracji potencjalnych zastosowań logik z dwoma zmiennymi i z Datalogiem do weryfikacji programów wskaźnikowych. Idea zastosowania logiki z dwoma zmiennymi do wyrażenia semantyki programów, a Datalogu do opisu początkowej struktury danych programu, pochodzi z wcześniejszych prac promotora rozprawy [20,21]. Jest to interesująca instancja ograniczonej weryfikacji programów, która sprowadza się w ogólnym zarysie do sprawdzenia spełnialności formuły opisującej symbolicznie wszystkie błędne wykonania programu.

Rozprawa, zredagowana w języku angielskim, oparta jest w dużej mierze na dwóch publikacjach konferencyjnych (LPAR 2010, LICS 2013) autora rozprawy i jej promotora.

Uwagi na temat rozprawy

Po przeczytaniu rozprawy, ogólne wrażenie jest zdecydowanie pozytywne.

Dowody są w większości zdecydowanie nietrywialne; szczególnie imponujące wrażenie zrobił na mnie rozdział 5 zawierający dowód górnej granicy dla logiki „$\mathcal{C}^2$ z drzewami". Górna granica jest taka sama jak dolna granica dla logiki $\mathcal{C}^2$, czyli dodanie Datalogu nie podwyższa złożoności obliczeniowej. Podobnie jest w przypadku logiki $\mathcal{C}^2 + \text{Datalog} + r + \{\text{BSR}\}$ – wydaje się, że dobranie parametrów $(r + \{\text{BSR}\})$ dających tę optymalną złożoność wymagało sporo wysiłku i bardzo wnikliwej analizy problemu. Od strony matematycznej, rozprawę uważam za znaczący wkład w rozwój dziedziny.

Kolejną mocną stroną rozprawy jest zgrabne połączenie wyników interesujących od strony teoretycznej, z ich potencjalnymi zastosowaniami w weryfikacji. I chociaż do faktycznych zastosowań wciąż daleko, to pierwszy nietrywialny krok został zrobiony. Mam tu na myśli głównie zastosowanie spełnialności w weryfikacji ograniczonej, aczkolwiek w rozprawie pokazano też zastosowanie spełnialności w dowodzeniu poprowności programów w stylu trójek Hoare'a.

Nie mogę się jednak powstrzymać od kilku poważnych uwag negatywnych na temat rozprawy. Po pierwsze, większość matematycznej zawartości rozprawy sformułowana jest w języku bardzo powściągliwym i formalnym; szczególnie dotyczy to rozdziału 5. Czytelnik musi przedzierać się przez

gąszcz formalnych, zawiłych, drobiazgowych i nieintuicyjnych definicji, rozciągających się na przestrzeni wielu stron, bez cienia wyjaśnienia w jakim celu te pojęcia są wprowadzone, i bez żadnej ilustracji czy przykładów. Bardzo trudno jest dostrzec podstawowe idee w takim gąszczu! Mam wrażenie, że autor rozprawy doktorskiej powinien włożyć więcej trudu w zrozumiałe przedstawienie materiału matematycznego, nawet jeśli jest on tak trudny i nietrywialny jak rozdział 5. W konsekwencji, bardzo trudno jest czytelnikowi sprawdzić poprawność rozumowań zawartych w rozprawie. Ponadto, brak też wyraźnego rozgraniczenia pomiędzy metodą z prac [77, 78], a wkładem własnym autora rozprawy.

Na tle rozdziału 5, pozytywnie wyróżnia się redakcja rozdziału 3, gdzie dowód głównego wyniku poprzedzony jest obszernym przykładem ilustrującym podstawowe idee dowodu.

Po drugie, szokujące wrażenie robi mnogość kombinacji, które można otrzymać manipulując poszczególnymi parametrami: $\forall\forall$, FO2, czy $\mathcal{C}^2$? q, r, czy r2?; które z warunków $\{\text{FUN}, \text{BSR}, \text{BIR}\}$? jeden czy więcej programów w Datalogu? Brakuje jasnego syntetycznego zestawienia, które z tych kombinacji są równoważne, które z warunków semantycznych są definiowalne w której z logik, oraz gdzie wśród tych wszystkich kombinacji przebiega granica złożoności NExpTime, a gdzie granica rozstrzygalności. Brak też wyjaśnienia, dlaczego wybrano właśnie takie warunki semantyczne. Jedyny jasno sformułowany w rozprawie wniosek jakościowy to fakt, że warunek BSR jest niezbędny dla uzyskania optymalnej złożoności. Ale dlaczego? A po co właściwie dodano do logiki nierówności liniowe? Domyślam się, że można je dodać „za darmo", ponieważ nierówności liniowe używane są w dowodzie. Wydaje się, że lepiej byłoby tylko wspomnieć o takiej możliwości, zamiast zaciemniać dodatkowo obraz przez kolejny element skomplikowanej i tak układanki.

Po trzecie, brak wyraźnego odróżnienia spełnialności od skończonej spełnialności, podczas gdy dla logik $\mathcal{C}^2 + \text{Datalog} + r + \{\text{BSR}\}$ oraz $\mathcal{C}^2 + \text{Datalog} + r2 + \{\text{BSR}, \text{BIR}\}$ są to dwa różne problemy. Słowo „spełnialność" w sformułowaniu twierdzeń 3.21 i 5.28 (jak i również w całej pracy począwszy od rozdziału 2.6) oznacza „skończoną spełnialność", o czym mówi wzmianka na stronie 22. Wzmiankę tę łatwo jednak przeoczyć, co prowadzić może do mylnego przypuszczenia, że dowody twierdzeń 3.21 i 5.28 stosują się do obydwu problemów. Zamieszanie powiększa brak konsekwencji w stosowaniu się do w.w. wzmianki, np. w sformułowaniu Lematu 3.25 występuje tak „spełnialność" jak i „skończona spełnialność". Brak też dyskusji dotyczącej (niekoniecznie skończonej) spełnialności: które z kroków w dowodach zawodzą w tym przypadku?

Część aplikacyjna rozprawy (rozdział 4), według mojego osobistego zdania, przedstawia dużą wartość (mimo świadomości ograniczeń; pamiętajmy, że metoda weryfikacji zaproponowana w rozprawie może zostać zastosowana tylko do *abstrakcji* programów, a nie do rzeczywistych programów). Być może wartość części aplikacyjnej jest nawet większa niż niewątpliwie istotny postęp w dyscyplinie poszerzania rozstrzygalnych fragmentów logik z dwoma zmiennymi, który przynosi rozprawa. Szkoda, że w rozprawie poprzestano na analizie abstrakcyjnych programów wskaźnikowych. Nie poruszono zupełnie tematyki ewentualnej integracji ze znanymi metodami uszczegóławiania abstrakcji, co wydaje się warunkiem niezbędnym dla jakichkolwiek prób praktycznych zastosowań. Żałuję, że tematyka taka nie pojawia się również w konkluzjach i planach na przyszłość.

Na koniec chciałbym wspomnieć o pewnych niedostatkach notacyjnych. W wielu miejscach w rozprawie mówi się o ciągach (np. ciągach programów w Datalogu), podczas gdy kolejność jest nieistotna i powinno mówić się o zbiorach. Inny, bardziej istotny i bardziej rozpowszechniony w tekście mankament rozprawy to konwencja nazewnicza, która używa semantycznych warunków do określenia syntaktycznego fragmentu logiki. Na przykład mówi się o formułach w logice

$$\mathcal{C}^2 + \mathrm{Datalog} + r + \{\mathrm{BSR}\},$$

podczas gdy powinno się mówić o formułach w logice

$$\mathcal{C}^2 + \mathrm{Datalog} + r,$$

a relacja spełniania powinna być sparametryzowana warunkami semantycznymi (w tym przypadku $\{\mathrm{BSR}\}$).

Pomijam w tej opinii szereg drobniejszych usterek, literówek, itp., których sporo udało mi się znaleźć w tekście, gdyż nie mają one wpływu na moją ocenę rozprawy.

Konkluzje

Negatywne uwagi wymienione powyżej nie przeważają nad ogólnie pozytywną oceną całościową. Uważam, że rozprawa doktorska spełnia bez wątpienia wszystkie zwyczajowe i formalne wymagania, i wnoszę o dopuszczenie Pana Piotra Witkowskiego do dalszych etapów przewodu doktorskiego.

Z wyrazami szacunku

Sławomir Lasota

S. Lasota