Nr rejestracyjny: 2017/27/B/ST6/00299; Kierownik projektu: dr Jan Otop

Specyfikacje ilościowe: uczenie się, algorytmy i zastosowania

W dzisiejszych czasach systemy informatyczne stanową nieodłączną część życia. Mimo dużych postępów w dziedzinach inżynierii oprogramowania i testowania, wciąż napotyka się na różnego rodzaju błędy, zarówno w sprzęcie, jak i w oprogramowaniu. Aby wyeliminować błędy zaproponowano formalną weryfikację, w której dowodzi się poprawności programów i układów cyfrowych. Formalna weryfikacja jest intensywnie badana teoretycznie oraz intensywnie rozwijana w przemyśle: Intel stosuje ją przy tworzeniu procesorów; Microsoft i Facebook stosują ją przy tworzeniu oprogramowania; Bosch stosuje ją przy projektowaniu tempomatów w samochodach. Wiodącą techniką w formalnej weryfikacji jest weryfikacja modeli, w ramach której sprzęt lub oprogramowanie abstrahuje się do uproszczonych struktur, nazywanych *modelami*, celem dokonania weryfikacji ich poprawności.

Weryfikacja modeli daje odpowiedź jakościową: „tak" albo „nie". Pozwala to odpowiadać na pytania w stylu „czy serwer odpowie na każde żądanie". Jednak odpowiedź pozytywna na takie pytanie jest często niewystarczająca, gdyż nie określa, jak szybko serwer odpowie. Dlatego ważne są również aspekty ilościowe, na przykład pytanie „jaki jest średni czas pomiędzy żądaniem a odpowiedzią serwera". Aby móc wnioskować o aspektach ilościowych systemów, stosuje się ilościową weryfikację modeli, w której można zadawać wartości liczbowe oraz otrzymywać liczby jako odpowiedzi. Takie odpowiedzi niosą ze sobą więcej informacji, które lepiej oddają poprawność systemu. Pomimo swoich zalet, ilościowa weryfikacja modeli nie jest używana w praktyce z powodu licznych trudności. Opisywanie własności ilościowych jest trudne, problem weryfikacji własności ilościowych jest obliczeniowo trudny, a interpretacja ilościowych odpowiedzi jest trudna. Celem tego projektu jest zniwelowanie powyższych trudności.

W tym projekcie badawczym postaramy się usprawnić proces ilościowej weryfikacji modeli skupiając się na trzech etapach weryfikacji: *specyfikacji* własności ilościowych, algorytmach *weryfikacji* modeli względem własności ilościowych, oraz *interpretacji* odpowiedzi ilościowych zwracanych w procesie weryfikacji.

Aby uczynić proces tworzenia specyfikacji łatwiejszym dla użytkowników, będziemy badali obliczeniowe uczenie się specyfikacji oraz języki zadawania specyfikacji. Będziemy badać dwa rodzaje problemów: uczenie *offline*, w którym użytkownik jednorazowo podaje pewną listę przykładów i na ich podstawie generowana jest specyfikacja, oraz uczenie *online*, w czasie którego algorytm może zadawać użytkownikowi dodatkowe pytania dotyczące słów, a także pytać, czy dana specyfikacja jest taka, jak żądano. Podobne techniki uczenia się z powodzeniem sprawdzają się w klasycznej weryfikacji.

Zakładając, że mamy zadaną specyfikację ilościową, kolejnym krokiem w weryfikacji jest obliczenie jej wartości na danym modelu systemu. Modele często bywają bardzo duże, dlatego ważne jest znajdowanie efektywnych algorytmów do weryfikacji, a w szczególności takich, które korzystają z technik pozwalających operować na mniejszy modelach. W tym projekcie podejmiemy próbę stworzenia takich algorytmów dla weryfikacji ilościowej.

Podobnie jak w oprogramowaniu, błędy mogą się również zdarzać w specyfikacjach. Może to prowadzić do sytuacji, gdy w wyniku weryfikacji otrzymujemy informację, że nasz system zachowuje się poprawnie, podczas gdy tak nie jest. Planujemy stworzyć narzędzia wspomagające wykrywanie tego typu przypadków. Celem takich narzędzi byłoby wskazywanie użytkownikowi, że jego specyfikacja, na przykład, jest spełniona podejrzanie łatwo lub że wynik procesu weryfikacji nie zależy od dużej części modelu.