Twierdzenie o niepustym barze, czyli zmechanizowana naturalna dedukcja

Sprawdzanie poprawności dowodów matematycznych często wymaga sporej wiedzy i ogromu nużącej pracy. O ile dochodzenie do zrozumienia istoty dowodu, czyli dlaczego dane twierdzenie matematyczne zachodzi, może sprawiać Czytelnikowi dużo satysfakcji, o tyle weryfikowanie wszystkich szczegółów dowodu jest zajęciem dość niewdzięcznym. Z tego powodu od wielu już lat trwają badania nad zaprzęgnięciem komputerów do tej żmudnej części pracy. Aktualnie nie istnieją programy, które potrafią rozumieć dowody matematyczne napisane po polsku lub po angielsku (i ogólnie, w żadnym języku naturalnym). Istnieją za to systemy z własnym językiem opisu dowodu, mniej lub bardziej oddalonym od języka naturalnego. Jednym z takich systemów jest Coq, rozwijany głównie we Francji, w instytucie naukowym INRIA. Jest to interaktywny system pozwalający na poszukiwanie dowodów twierdzeń za pomocą zwięzłych komend, nazywanych *taktykami*.

Naturalna dedukcja to opracowany w latach 30. XX wieku przez polskiego logika Stanisława Jaśkowskiego i (niezależnie) przez niemieckiego matematyka Gerharda Gentzena, zestaw formalnych zasad, które dla każdego spójnika logicznego (takiego jak np. alternatywa, implikacja, kwantyfikatory) mówią, jak w dowodzie wykorzystać zdanie zbudowane przy użyciu tego spójnika oraz jak udowodnić takie zdanie.

Każda taktyka odpowiada z grubsza zastosowaniu pewnej reguły dowodzenia zwanej *naturalną dedukcją*. Coq nie służy do automatycznego dowodzenia twierdzeń, do działania potrzebuje ludzkich podpowiedzi. Każdą podpowiedzianą taktykę stosuje bezbłędnie, uzyskując wszystkie możliwe wnioski i sprawdzając, co jeszcze należy udowodnić, żeby dowód analizowanego twierdzenia był kompletny. W niniejszym artykule zaprezentujemy dowód w Coqu, wykonany z użyciem reguł naturalnej dedukcji, następującego twierdzenia:

W każdym niepustym barze jest taka osoba, że jeśli ona pije, to wszyscy piją.

Choć twierdzenie brzmi absurdalnie, jednak da się je udowodnić. Przyczyny tego dysonansu omówimy później, a teraz rozważmy dwa dowody.

Dowód ludzki

a Z *zasady wyłączonego środka* wiadomo, że w barze albo istnieje osoba, która nie pije, albo nie istnieje taka osoba.
b Rozpatrzmy dwa przypadki.
c Jeśli istnieje osoba, która nie pije, to
d spełnia ona implikację „jeśli ona pije, to wszyscy piją”,
e gdyż niespełniony jest poprzednik implikacji.
f W przeciwnym przypadku wiemy, że nie ma osoby, która nie pije,
g a zatem wszyscy piją,
h czyli konkluzja naszej implikacji jest spełniona.
i A ponieważ bar nie jest pusty,
j więc istnieje osoba, dla której ta implikacja jest spełniona.

Dowód wsparty komputerem

```
1  From Coq Require Import Classical.
2  Parameter Bar: Type.
3  Parameter Pije: Bar → Prop.
4  Parameter gosc: Bar.
5  Theorem SuperGosc: exists x, (Pije x →
   forall y, Pije y).
6  Proof.
7   assert ((exists x, ∼Pije x) ∨
   ∼(exists y, ∼Pije y)) as H by apply classic.
8   destruct H.
9   + destruct H.
10    exists x.
11    intro.
12    contradiction.
13  + exists gosc.
14    intro.
15    apply not_ex_not_all.
16    assumption.
17 Qed.
```

Na początku dowodu w Coqu twierdzenie zostaje sformalizowane. Po pierwszej technicznej linii, wczytującej zestaw twierdzeń logiki klasycznej, deklarujemy typ `Bar` (linia 2) rozumiany tak, że `x:Bar` oznacza, że osoba `x` jest w barze, i wprowadzamy predykat `Pije`, który rozumiemy tak, że `Pije x` oznacza, że osoba `x` pije. Niepustość baru zapewnia nam stała `gosc` (linia 4), oznaczająca pewną osobę, która siedzi w barze.

Po *zrozumieniu* treści twierdzenia (linia 5) Coq przechodzi w tryb dowodu, w którym oczekuje komend od użytkownika – instrukcji, jaką regułę wnioskowania ma zastosować. Po każdej komendzie Coq wyświetla aktualną sytuację dowodową, czyli co jeszcze należy pokazać i przy jakich lokalnych założeniach, aby ukończyć dowód twierdzenia.

Pierwsza taktyka naszego dowodu w Coqu (linia 7) powoduje wzmocnienie lokalnych założeń o alternatywę wynikającą z prawa wyłączonego środka. To tak samo, jak w dowodzie ludzkim w punkcie a. Prawo wyłączonego środka w Coqu nosi nazwę `classic`, a przywołane założenie otrzymuje nazwę `H`. Po tej linii sytuacja dowodowa wygląda następująco[a]:

```
H : (exists x : Bar, ~Pije x) ∨
 ~(exists y : Bar, ~Pije y)
============================
 exists x : Bar, Pije x → forall y : Bar, Pije y
```

Przy założeniach widocznych nad podwójną linią (aktualnie jest tylko jedno założenie – o nazwie `H`) mamy udowodnić formułę znajdującą się poniżej linii.

Taktyka **destruct** `H` odpowiada regule użycia głównego spójnika założenia `H`. W przypadku linii 8 założenie `H` to alternatywa. Jej reguła użycia odpowiada dowodowi przez rozbicie na przypadki (patrz punkt b z dowodu ludzkiego), – rozdzielenie dowodu na dwa wątki, których początki zaznaczone są znakiem `+` (w dowodzie ludzkim wątki te zaczynają się w punktach c i f). Odpowiadająca pierwszemu wątkowi sytuacja dowodowa (po `+` w linii 9) jest następująca:

```
H : exists x : Bar, ~Pije x
============================
 exists x : Bar, Pije x → forall y : Bar, Pije y
```

Komenda z linii 8 spowodowała zastąpienie poprzedniego założenia `H` przez założenie odpowiednie dla wątku – czyli w tym przypadku pierwszy człon alternatywy. Kolejna komenda **destruct** `H` tym razem odpowiada użyciu formuły z kwantyfikatorem egzystencjalnym. Polega ono na wprowadzeniu do dowodu nowej zmiennej `x`, zwanej „świadkiem” (istnienia osoby niepijącej) oraz jego własności `~Pije x`. Tak jak poprzednio, użyte założenie `H` znika, a nazwa ta zostaje wykorzystana ponownie.

Zauważmy, że ponieważ wprowadzono do kontekstu dowodowego nazwę `x`, zmienna związana w formule poniżej podwójnej linii została przemianowana na `x0`.

```
x : Bar
H : ~Pije x
============================
 exists x0 : Bar, Pije x0 → forall y : Bar, Pije y
```

Powyższy krok dowodowy nie ma bezpośredniego odzwierciedlenia w dowodzie ludzkim. Odpowiada mu to, że do „osoby”, wymienionej w punkcie c, można odwoływać się w dalszej części tego wątku dowodu (punkty d i e). Zwykle w dowodach w języku naturalnym nie odróżnia się zdania egzystencjalnego jako całości od istnienia świadka (co wynika z tego zdania) i jego własności, które mogą być wykorzystane w dalszej części dowodu.

Kolejna taktyka, **exists** `x` z linii 10 odpowiada kanonicznej regule dowodzenia formuły egzystencjalnej z naturalnej dedukcji: wskazujemy świadka `x` i przechodzimy do dowodu, że świadek żądaną własność faktycznie spełnia.

```
x : Bar
H : ~Pije x
============================
 Pije x → forall y : Bar, Pije y
```

[a]Zauważmy, że Coq dopisał `: Bar` w kilku miejscach, ponieważ widząc wyrażenia `Pije x` i `Pije y`, *domyślił się*, że `x` i `y` oznaczają gości baru. Poza tym Coq usunął kilka niepotrzebnych według niego nawiasów. Oczywiście, znaczenie formuł nie uległo zmianie.

Formuła, którą mamy teraz do udowodnienia, jest implikacją i znów używamy stosownej reguły kanonicznego dowodu z naturalnej dedukcji – za pomocą komendy **intro** w linii 11. Po tej komendzie okazuje się, że nasz zbiór lokalnych założeń jest sprzeczny.

```
x : Bar
H : ~Pije x
H0 : Pije x
============================
 forall y : Bar, Pije y
```

Coq zauważa to po wprowadzeniu komendy **contradiction** (linia 12), zamyka wtedy bieżący wątek dowodu (powyższe dwie komendy odpowiadają punktom d i e).

```
This subproof is complete, but there are some
unfocused goals.
```

Sytuacja początkowa drugiego wątku dowodu w Coqu (po znaku `+` w linii 13) jest następująca:

```
H : ~(exists y : Bar, ~Pije y)
============================
 exists x : Bar, Pije x → forall y : Bar, Pije y
```

Dowód przebiega analogicznie do dowodu ludzkiego, z tym że kolejność kroków rozumowania musi być zgodna z regułami naturalnej dedukcji: najpierw wskazujemy świadka formuły egzystencjalnej (linia 13 odpowiada punkt j), następnie przechodzimy do dowodu następnika implikacji (linia 14), a na koniec, korzystając z odpowiedniego prawa de Morgana, nazwanego w Coqu `not_ex_not_all`, zamieniamy kwantyfikowany ogólnie cel na formułę, która jest jednym z naszych założeń. To ostatnie spostrzeżenie przekazujemy Coqowi instrukcją **assumption** w linii 16, po której następuje już zamknięcie całego dowodu komendą **Qed**.

W dowodzie ludzkim, korzystając z elastyczności języka naturalnego, można sobie pozwolić na podanie uzasadnienia dla następnika implikacji (punkt h) jeszcze przed wskazaniem świadka dla formuły egzystencjalnej (punkt j).

Na koniec wyjaśnijmy, dlaczego twierdzenie intuicyjnie fałszywe daje się udowodnić. Wynika to z drobnego oszustwa opartego na niejednoznaczności języka naturalnego. Podane twierdzenie w języku polskim rozumiemy intuicyjnie tak: w każdym niepustym barze jest taka osoba, że *zawsze* jeżeli ona pije, to wszyscy piją – co jest oczywistą nieprawdą. Natomiast zdanie to rozumiane w języku matematyki, jak i dowodzona w Coqu formuła logiczna, opisuje (każdą) pojedynczą chwilę. I dlatego, jak pokazują dowody, jest prawdziwe.

Jacek CHRZĄSZCZ

Coq można pobrać ze strony `http://coq.inria.fr` (wersja wymagająca instalacji). Eksperymentalna wersja on-line jest dostępna na stronie `https://x80.org/collacoq/`. Po załadowaniu (co może potrwać nawet minutę) w prawym dolnym rogu (Packages) należy kliknąć w chmurkę coq-base, powodując załadowanie odpowiedniego pakietu. Następnie do lewego panelu można wpisać treść dowodu w Coqu i poruszać się po nim, używając strzałek w prawym panelu.