A jednak się da (III),

czyli saga kryptologiczna w odcinkach.

Tym razem: o dowodach z wiedzą zerową.

Tomasz KAZANA

Bywa tak, że chcemy o czymś przekonać niedowiarków, jednak w taki sposób, aby uwierzyli, ale też aby za dużo się nie dowiedzieli. Fabularna, nieinformatyczna ilustracja, którą lubię przywoływać, gdy próbuję wyrazić, o co mi chodzi, jest następująca. Wyobraźmy sobie, że potrafię zliczyć liczbę liści na drzewie, jeśli tylko spojrzę na nie przez 5 sekund. Więcej: twierdzę, że umiem to publicznie udowodnić w przeciągu 5 minut, i to tak, że wszyscy mi uwierzą, ale nikt się nie zorientuje, jak ja to robię!

Jak mógłby wyglądać *protokół*, który realizuje powyższe założenia? Popatrzmy:

1. Proszę publiczność o wskazanie dowolnego drzewa.
2. Patrzę na to drzewo przez 5 sekund i mówię: „to drzewo ma M liści".
3. Proszę publiczność o zawiązanie mi oczu.
4. Proszę publiczność o podejście do analizowanego drzewa i zerwanie z niego dowolnych K liści (oczywiście K publiczność ustala samodzielnie, nie informując mnie o swoim wyborze).
5. Proszę publiczność o rozwiązanie mi oczu.
6. Ponownie patrzę na drzewo przez 5 sekund i mówię: „to drzewo ma X liści".

I teraz tak: jeśli $X \neq M - K$, to z pewnością nikogo nie przekonałem. Jednak jeśli $X = M - K$, to eksperyment daje do myślenia. Oczywiście, aby zmniejszyć szansę zdarzenia, że po prostu miałem szczęście, protokół weryfikacyjny można kilka razy powtórzyć, dla różnych drzew i różnych wyborów zmiennej K. Jeśli więc pozytywnie przejdę na przykład 10 testów pod rząd, to sądzę, że już wszyscy mi uwierzą, że naprawdę umiem błyskawicznie zliczać liście na drzewie. Mimo to: nikt nie ma nawet cienia wskazówki, jak ja to robię!

Prezentowana anegdota jest elementem informatycznego folkloru. Autor tego tekstu po raz pierwszy usłyszał ją od profesora Damiana Niwińskiego.

Właśnie takie przekonywanie jak wyżej nazywamy fachowo *dowodzeniem z wiedzą zerową* (*zero-knowledge proofs*), a bardziej kolokwialnie: przekomarzaniem się w stylu „wiem, ale nie powiem". Okazuje się, że zaskakująco wiele stwierdzeń matematycznych da się udowodnić w taki właśnie sposób. W szczególności: gdybyśmy na przykład potrafili rozstrzygnąć pozytywnie hipotezę Riemanna, to możemy zaproponować światu pewną zabawę, w wyniku której wszyscy uwierzą, że faktycznie hipoteza Riemanna jest prawdziwa, ale nikt nawet powierzchownie nie liźnie smaku naszego dowodu. Jak to zrobić – obiecuję naszkicować na końcu, a na razie zajmijmy się prostszym przykładem.

Izomorfizm grafów

Szerzej o problemie izomorfizmu grafów pisał Łukasz Kowalik w Δ^{12}_{18}.

Opiszemy protokół dowodu z wiedzą zerową dla problemu izomorfizmu grafów. Przypomnijmy: problem polega na rozstrzygnięciu, czy dwa dane grafy G_1 oraz G_2 są izomorficzne, czy też nie. Załóżmy więc, że znamy pewien izomorfizm π między G_1 a G_2. Naszym celem jest przekonać *publiczność* (fachowo: *weryfikatora*), że faktycznie znamy π, ale bez ujawniania, jak to π w rzeczywistości wygląda.

Załóżmy, że zarówno zbiór wierzchołków G_1, jak i G_2 to $\{1, 2, \ldots, n\}$. Permutacja $\pi : \{1, 2, \ldots, n\} \to \{1, 2, \ldots, n\}$ jest izomorfizmem między G_1 a G_2, gdy dla każdej pary i, j spełniony jest warunek: w G_1 istnieje krawędź między wierzchołkami i a j wtedy i tylko wtedy, gdy w G_2 istnieje krawędź między wierzchołkami $\pi(i)$ a $\pi(j)$.

Do dzieła:

1. Postulujemy publicznie, że grafy G_1 oraz G_2 są izomorficzne.
2. Publikujemy dowolny graf H, który jest izomorficzny z G_1 (czyli również z G_2).
3. Weryfikator wybiera liczbę $k \in \{1, 2\}$.
4. Publikujemy izomorfizm π' między H a G_k.
5. Weryfikator sprawdza, czy π' jest rzeczywiście poprawnym izomorfizmem między H a G_k.
6. Kroki 2–5 powtarzamy N razy.

Powyższy protokół ma trzy kluczowe cechy:

(a) Jeśli naprawdę znamy postulowany izomorfizm, to jesteśmy zawsze w stanie wykonać poprawnie cały protokół;

Dowody z wiedzą zerową nie muszą być stuprocentowo pewne. Formalna definicja (której w tym tekście nie podajemy) dopuszcza bardzo małe (*negligable*, pomijalne) prawdopodobieństwo, że weryfikator zostanie oszukany.

(b) Szansa, że oszukamy weryfikatora (to znaczy: przekonamy go, że znamy izomorfizm między G_1 a G_2, mimo że wcale go nie znamy) wynosi co najwyżej $\frac{1}{2^N}$;
(c) Przekonany weryfikator nie ma bladego pojęcia, jak może wyglądać izomorfizm między G_1 a G_2.

Naszkicujmy zatem uzasadnienia stwierdzeń (b) oraz (c) (pozwolimy sobie napisać, że (a) jest trywialne):

Ad (b) Zauważmy, że jeśli odnosimy sukces z prawdopodobieństwem powyżej $\frac{1}{2^N}$, to przynajmniej w jednej z N iteracji protokołu, w kroku 4. byliśmy przygotowani na opublikowanie izomorfizmu zarówno między H a G_1 (ozn. π_1), jak i między H a G_2 (ozn. π_2). Istotnie – gdyby tak nie było, to w każdej iteracji mielibyśmy szansę co najwyżej $\frac{1}{2}$ na akceptację przez weryfikatora, a więc ogólnie – co najwyżej tylko $\frac{1}{2^N}$. Jednak znajomość zarówno π_1 jak i π_2 oznacza również znajomość $\pi_2 \circ \pi_1^{-1}$, a to jest przecież izomorfizm między G_1 a G_2.

Ad (c) Zastanówmy się, czego – po wykonaniu całego protokołu – *dowiedział* się weryfikator. Jak widać, poznał N różnych trójek postaci (H_i, e_i, π'_i) takich, że π'_i jest izomorfizmem między H_i a G_{e_i}. Czy taka wiedza jest w jakikolwiek sposób ekskluzywna? Otóż łatwo zauważyć, że absolutnie nie – przecież takie trójki można sobie samemu (w dowolnych ilościach) produkować, znając wyłącznie G_1 oraz G_2, a te są przecież publiczne od samego początku!

Cykl Hamiltona w grafie

Cykl Hamiltona to takie uszeregowanie wierzchołków grafu, że między każdą parą kolejnych (oraz między ostatnim i pierwszym) wierzchołków w tym uszeregowaniu istnieje w tym grafie krawędź.

Czas na nieco bardziej skomplikowany przykład – protokół dowodu z wiedzą zerową dla problemu istnienia cyklu Hamiltona w grafie. W tym protokole będziemy używać idei *zobowiązań* (Commit), opisanych przez Łukasza Rajkowskiego w poprzednim odcinku AJSD, w numerze Δ^{11}_{18}.

Jak poprzednio, załóżmy, że znamy cykl Hamiltona $c = (a_1, \ldots, a_n)$ pewnego grafu G. Teraz:

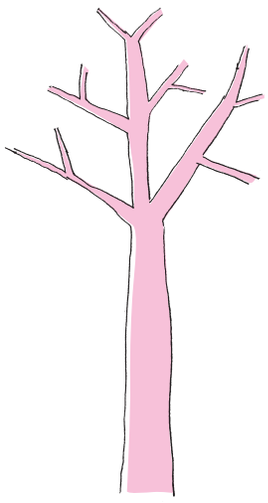
1. Postulujemy publicznie, że w G istnieje cykl Hamiltona. Będziemy próbować to udowodnić, ale nie zdradzając, jak faktycznie wygląda c.
2. Losujemy dowolną permutację π oraz obliczamy graf $\pi(G)$ (czyli izomorficzny do G, ale z wierzchołkami spermutowanymi według π).
3. Upubliczniamy następujące zobowiązania:
   - Commit(π)
   - Commit(e_1), Commit(e_2), …, Commit(e_m), gdzie $\{e_1, \ldots, e_m\}$ stanowią opisy wszystkich krawędzi grafu $\pi(G)$ (to znaczy: każde e_i jest parą pewnych wierzchołków grafu $\pi(G)$).
4. Weryfikator wybiera liczbę $k \in \{1, 2\}$.
5. Jeśli $k = 1$, to otwieramy wszystkie (zarówno permutacji, jak i krawędzi) zobowiązania z kroku 3.
6. Jeśli $k = 2$, to otwieramy zobowiązania tych (i tylko tych) krawędzi, które tworzą cykl Hamiltona w $\pi(G)$ (NIE otwieramy zobowiązania permutacji!).
7. Jeśli $k = 1$, to weryfikator sprawdza, czy otwarte krawędzie $\{e_1, \ldots, e_m\}$ są faktycznie krawędziami grafu $\pi(G)$.
8. Jeśli $k = 2$, to weryfikator sprawdza, czy faktycznie otwarte krawędzie tworzą jeden zamknięty cykl długości n.
9. Kroki 2–8 powtarzamy N razy.

Telegraficzny skrót wiedzy o zobowiązaniach: po opublikowaniu Commit(x) praktycznie nie jest możliwe odczytanie x. *Otworzyć* zobowiązanie (ujawniając x) może tylko jego autor, przy czym – nie jest w stanie oszukać, czyli wmówić, że w zobowiązaniu znajdował się jakiś $x' \neq x$.

Intuicja: jeśli $k = 1$, to weryfikator sprawdza, czy nie oszukujemy przy konstruowaniu $\pi(G)$; jeśli $k = 2$, to weryfikator sprawdza, czy w zobowiązanym grafie faktycznie jest cykl Hamiltona; żadna z tych informacji pojedynczo nie pozwala na odtworzenie cyklu Hamiltona w G, jednak aby przygotować zobowiązania z kroku 3 tak, aby mieć pewność przejścia obu testów w krokach 7–8, TRZEBA znać cykl Hamiltona w G.

Prezentowane przykłady są poprawne i działają w rozsądnej asymptotycznej złożoności czasowej, jednak daleko im jeszcze do uznania ich za zupełnie praktyczne. Stąd istnieje ważny nurt badań kryptologicznych poszukujących innych protokołów zero-knowledge. Pewien ważny postęp miał miejsce w roku 2007, gdy Jens Groth i Amit Sahai zaproponowali bardzo szybkie (po prostu praktyczne) protokoły oparte o teorię grup dwuliniowych, działające dla dość szerokiej, ale niestety istotnie mniejszej od NP, klasy problemów. Mają one jeszcze inną pożądaną cechę: nie wymagają wielorundowej interakcji między weryfikatorem a osobą dowodzącą, jak w naszych przykładach.

Twierdzimy, że ten protokół również ma trzy cechy, o których mówiliśmy w poprzednim przykładzie. To znaczy: jeśli rzeczywiście znamy cykl Hamiltona w G, to wykonamy protokół bez problemu (trywialna obserwacja). Dalej: szansa na oszukanie weryfikatora wynosi ponownie co najwyżej $\frac{1}{2^N}$, a schemat dowodu jest dokładnie taki sam jak poprzednio. Wystarczy tylko zauważyć, że jeśli w którejkolwiek iteracji jesteśmy pewni akceptacji weryfikatora (niezależnie od wyboru k), to znaczy, że zobowiązaliśmy się do permutacji π takiej, że znamy cykl Hamiltona w $\pi(G)$. To jednak oznacza, że znamy również cykl Hamiltona w G. Pozostaje tylko upewnić się, że nabyta (w wyniku realizacji protokołu) przez weryfikatora wiedza nic mądrego mu nie mówi. I tak jest w istocie: jeśli



$k = 1$, to weryfikator uczy się tylko pewnego izomorfizmu G, a jeśli $k = 2$, to weryfikator poznaje losową permutację liczb $\{1, \ldots, n\}$, zupełnie niezależną od G (bo w tym przypadku nie zna π!).

Co jeszcze?

Pokazaliśmy dwa przykłady protokołów dowodów z wiedzą zerową. Nasuwa się pytanie: jakie inne (i jak bardzo skomplikowane) stwierdzenia możemy podobnie dowodzić? Okazuje się (być może zaskakująco), że… *niemal wszystkie*. Aby się o tym przekonać, potrzebna jest pewna wiedza z teorii złożoności obliczeniowej, a konkretnie: informacja, że problem istnienia cyklu Hamiltona w grafie jest problemem NP-zupełnym. Nie jest ambicją tego artykułu, aby dokładnie wytłumaczyć to pojęcie (patrz na przykład Δ^{11}_{13}, Δ^{1}_{17} czy Δ^{11}_{17}), więc siłą rzeczy musimy w tym momencie trochę rozluźnić rygor ścisłej precyzji na rzecz intuicji.

(„Niemal wszystkie” w akapicie wyżej należy rozumieć jako „niemal wszystkie występujące w praktyce”. Dla Prawdziwych Teoretyków klasa NP to wręcz „niemal nic”.)

Otóż fakt, że problem cyklu Hamiltona (CH) jest NP-zupełny, oznacza, że dzięki temu, że pokazaliśmy protokół dowodu z wiedzą zerową dla CH, wiemy jak konstruować protokoły z wiedzą zerową dla **dowolnego innego** problemu z klasy NP! W jaki sposób? Wystarczy zastosować odpowiednią efektywną redukcję do problemu CH (która musi zawsze istnieć) i dalej stosować protokół opisany wyżej.

Czytelnik Spostrzegawczy zauważy, że jedną rzecz na temat naszego dowodu jednak zdradzamy – mianowicie górne oszacowanie jego długości (wynoszące k).

W szczególności: załóżmy, że udowodniliśmy hipotezę Riemanna. Rozważmy teraz język:

$$L = \{\text{zdanie prawdziwe } \phi \mid \phi \text{ ma dowód długości } \leqslant k\}.$$

Niewątpliwie $L \in \mathrm{NP}$ (dlaczego?), więc dla każdego $\phi' \in L$ istnieje dowód z wiedzą zerową, w szczególności: dla $\phi' =$ „Hipoteza Riemanna jest prawdziwa”.

A po ludzku: okazuje się, że istnieje graf G^k_{Riemann} (rozmiaru wielomianowego od k) taki, że jeśli hipoteza Riemanna ma dowód długości $\leqslant k$, to w tym grafie jest cykl Hamiltona, a jeśli nie ma takiego dowodu – to i cyklu Hamiltona w G^k_{Riemann} nie znajdziemy. Więcej: znalezienie grafu G^k_{Riemann} jest efektywnie obliczalną funkcją zdania ϕ' (zapisanego w jakiejś formalnej logice). Jeśli więc rzeczywiście znajdziemy dowód dla hipotezy Riemanna długości k i mamy nieodpartą pokusę pohandryczyć się ze światem, to możemy zawsze przedstawić wyłącznie dowód istnienia cyklu Hamiltona w grafie G^k_{Riemann}. Dowód z wiedzą zerową, rzecz jasna!

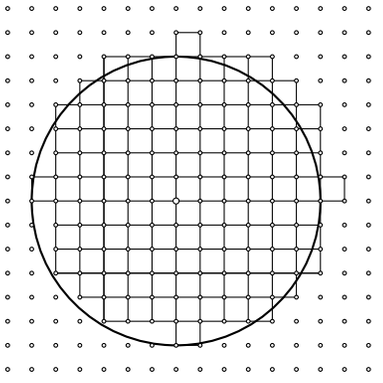
Czytelnik Niezłośliwy jest zapewne w stanie uwierzyć, że dowody z wiedzą zerową są ważnym narzędziem całej kryptologii, a nie tylko jej złośliwej części. Są chociażby podstawą anonimowych kryptowalut (np. Zerocash). Inne ciekawe ich zastosowanie przedstawimy w odcinku VI naszej sagi.

Kraty

*Jarosław GÓRNICKI**

* Wydział Matematyki i Fizyki Stosowanej, Politechnika Rzeszowska

Na płaszczyźnie euklidesowej $\mathbb{R}^2$ zbiór $\mathbb{Z}^2 = \{(m, n) : m, n \in \mathbb{Z}\}$ nazywamy *kratą*, a jego elementy *punktami kratowymi*.



Rys. 1. Liczba $L(r)$ jest równa powierzchni pokrytej przez kwadraty jednostkowe, których dolny lewy wierzchołek leży wewnątrz lub na brzegu koła

Carl Gauss zauważył, że jeśli liczba punktów kratowych w kole $x^2 + y^2 \leqslant r^2$ wynosi $L(r)$, to $\dfrac{L(r)}{r^2} \to \pi$, gdy $r \to \infty$ (rys. 1). Empirycznie wyznaczył $L(100) = 31\,417$, więc $\pi \approx 3{,}1417$. Precyzyjnie, Gauss pokazał, że $L(r) = \pi r^2 + E(r)$, gdzie błąd $|E(r)| \leqslant 2\sqrt{2}\pi r$. Do dziś nie wiemy, jakie jest najlepsze oszacowanie tego błędu.

Szybko okazało się, że kraty to ciekawy obiekt badań matematycznych. Na przykład, na płaszczyźnie łatwo wykreślić prostą, która nie przecina zbioru $\mathbb{Z}^2$. Jeśli na prostej znajdują się dwa punkty kratowe, to jest ich na tej prostej nieskończenie wiele i są one rozmieszczone w równych odstępach. Istnieją też proste, które zawierają dokładnie jeden punkt kratowy. Gdyby prosta przechodząca przez punkty $(0, 0)$ i $(1, \sqrt{2})$ zawierała punkt kratowy $(m, n) \neq (0, 0)$, to z twierdzenia Talesa uzyskalibyśmy, że $\sqrt{2} = \dfrac{n}{m}$, a to jest niemożliwe, bo $\sqrt{2}$ jest liczbą niewymierną.