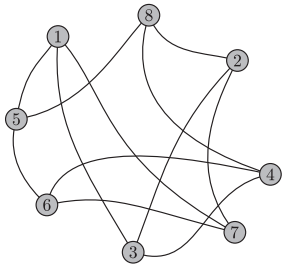
Uwierzytelnianie cyfrowe za pomocą grafów – dowody z wiedzą zerową

Krzysztof KULEWSKI

Artykuł ten jest częścią referatu wygłoszonego na I Kongresie Młodych Matematyków Polskich, który odbył się w Warszawie w dniach 17–19 września 2004 roku.

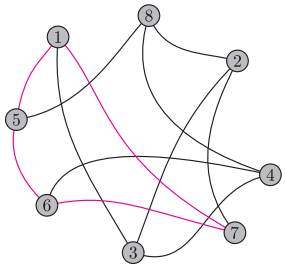
Dowody z wiedzą zerową – w wielkim uproszczeniu jest to sposób przekonania kogoś o naszej znajomości pewnego faktu bez ujawniania mu tego faktu.



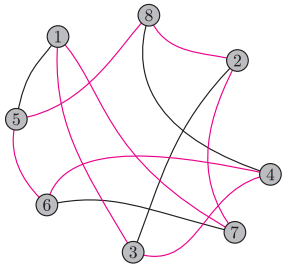
Rys. 1. Przykładowy graf G.

1	2	3	4	5	6	7	8
3	3	1	3	1	4	1	2
5	7	2	6	6	5	2	4
7	8	4	8	8	7	6	5

Reprezentacja grafu G.



Rys. 2. Cykl w grafie G. Reprezentacja cyklu w grafie G: 1 5 6 7.



Rys. 3. Cykl Hamiltona w grafie G. Reprezentacja cyklu Hamiltona w grafie G: 1 3 4 6 5 8 2 7.

Cykl Eulera to taki, w którym każda z krawędzi występuje dokładnie jeden raz.

Powszechnie znane są możliwości stosowania, na przykład, dużych liczb pierwszych w uwierzytelnianiu cyfrowym – tak między innymi działa algorytm RSA. Ale wymyślono także inne metody realizacji tego zagadnienia – niekiedy bardzo zaskakujące i nieoczekiwane. Jedną z nich są dowody z wiedzą zerową.

Ale zacznijmy od początku. Algorytm, o którym będziemy mówili, wykorzystuje do swojego działania grafy („wierzchołki połączone krawędziami"). Graf to zbiór wierzchołków V oraz zbiór krawędzi E – połączeń pomiędzy wierzchołkami.

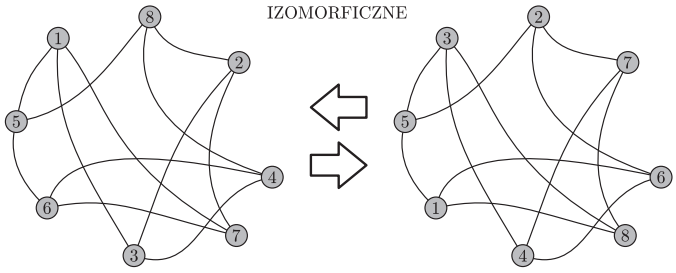
Na rysunku 1. znajduje się przykładowy graf G. Zawiera on 8 wierzchołków i 12 krawędzi.

Grafy możemy reprezentować np. przez macierz – czyli po prostu tabelkę. Pierwszy wiersz to etykiety kolejnych wierzchołków. Jeżeli pod kreską w kolumnie i znajduje się liczba j, oznacza to, że wierzchołki i oraz j połączone są krawędzią. Dobrze jest, aby zarówno w pierwszym wierszu, jak i w kolumnach, obowiązywał porządek rosnący.

Cyklem w grafie nazywamy taki ciąg jego krawędzi, który przeprowadza nas przez kilka wierzchołków i wraca do punktu wyjścia. Jeśli w naszym grafie G pójdziemy ścieżką, zaczynając od wierzchołka 1, potem 5, 6, 7 i na końcu znów wracając do wierzchołka 1, to będzie to cykl, tak jak na rysunku 2.

Cyklem Hamiltona w grafie nazywamy cykl, który przechodzi przez każdy wierzchołek dokładnie jeden raz. Na rysunku 3. znajduje się cykl Hamiltona w naszym grafie G. Dla wygody, cykl Hamiltona rozpoczynamy od wierzchołka oznaczonego etykietą 1.

Ostatnim pojęciem, którego będziemy potrzebowali, jest izomorficzność. Mówimy, że dwa grafy są izomorficzne, gdy jeden powstaje z drugiego przez pozamienianie etykiet wierzchołków. Na rysunku 4. znajduje się nasz graf G oraz izomorficzny z nim graf H.



Rys. 4. Izomorficzne grafy G i H.

W grafie H mamy więc cykl Hamiltona: 3 4 6 1 5 2 7 8, co po uporządkowaniu daje 1 5 2 7 8 3 4 6.

Pojawiają się dwa problemy. Jak znaleźć cykl Hamiltona w dowolnym grafie oraz jak stwierdzić, czy dwa grafy są izomorficzne?

W informatyce wyróżniono kilka klas problemów obliczeniowych. Do klasy P (ang. *polynomial* – wielomianowy) zaliczono problemy obliczeniowo łatwe – czyli takie, dla których znane jest rozwiązanie działające w czasie wielomianowym. Przykładem może być zagadnienie znajdowania najkrótszej ścieżki w grafie bądź też znajdowanie cyklu Eulera. Problemy NP (ang. *nondeterministic polynomial* – niedeterministycznie wielomianowy) to takie, których rozwiązania są weryfikowalne w czasie wielomianowym, rozwiązywalne zaś algorytmem niedeterministycznym w czasie wielomianowym (np. na niedeterministycznej maszynie Turinga).

W szczególności łatwo zauważyć, że problemy klasy P są NP, ponieważ ich rozwiązania można sprawdzić w czasie wielomianowym.

Istnieje dość liczna grupa problemów, dla których pokazano, że należą do klasy NP, a nie udało się pokazać, że należą do klasy P. W problemach NP

Jeżeli weźmiemy permutację wierzchołków, przeprowadzającą graf G na graf H:

$$\begin{pmatrix} 1 & 2 & 3 & 5 & 6 & 7 & 8 \\ 3 & 7 & 4 & 5 & 1 & 8 & 2 \end{pmatrix}$$

to graf H będzie postaci:

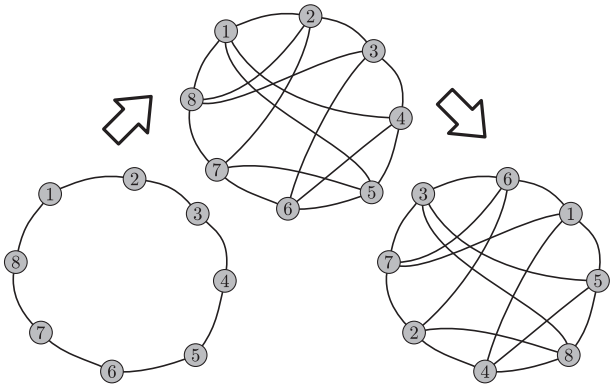
3	7	4	6	5	1	8	2
4	2	3	4	1	6	7	5
8	8	6	1	2	5	3	7
5	4	7	2	3	8	1	6

czyli po uporządkowaniu:

1	2	3	4	5	6	7	8
5	5	4	3	1	1	2	1
6	6	5	6	2	2	4	3
8	7	8	7	3	4	8	7

W grafie H mamy więc cykl Hamiltona: 3 4 6 1 5 2 7 8, co po uporządkowaniu daje: 1 5 2 7 8 3 4 6.

Jak wygenerować graf wraz z cyklem Hamiltona? Bardzo prosto. Najpierw łączymy kolejne wierzchołki tak, żeby mieć cykl Hamiltona, a następnie losowo dodajemy dużą liczbę krawędzi. Na koniec permutujemy graf. Schemat ten jest pokazany na rysunku 5.



Rys. 5. Schemat tworzenia grafu do uwierzytelniania.

Liczba rund	Szansa oszustwa
10	$9,76562 \cdot 10^{-4}$
30	$9,31323 \cdot 10^{-10}$
100	$7,88861 \cdot 10^{-31}$

Jeśli oszust byłby w stanie wykonywać takie sto rund w czasie 10^{-9} sekundy każda, to udałoby mu się podszyć pod klienta średnio po $2 \cdot 10^{13}$ lat.

Literatura:

[1] Douglas Robert Stinson *Cryptography. Theory and Practice*, CRC Press LLC, Washington DC. 1995,

[2] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest *Wprowadzenie do algorytmów*, Wydawnictwa Naukowo-Techniczne, Warszawa 1998.

Autor jest studentem pierwszego roku Wydziału Matematyki, Informatyki i Mechaniki Uniwersytetu Warszawskiego.

wyróżniono również specjalną podgrupę, tak zwane problemy NP-zupełne, które można scharakteryzować jako najtrudniejsze problemy w NP. Problem jest problemem NP-zupełnym, jeżeli jest problemem NP oraz dodatkowo każdy inny problem NP jest do niego redukowalny. A zatem, gdyby pokazano, że któryś problem NP-zupełny jest problemem należącym do klasy P, oznaczałoby to, że wszystkie problemy NP należą również do klasy P.

Pierwszym problemem, dla którego pokazano NP-zupełność, był problem spełnialności formuł logicznych (SAT).

Najprawdopodobniej w NP istnieją także problemy pośrednie, to znaczy takie, które nie należą ani do P, ani do NP-zupełnych.

Zarówno problemy NP-zupełne, jak i problemy uważane za pośrednie, traktowane są obecnie jako problemy obliczeniowo trudne, to znaczy takie, których rozwiązanie w jakimś sensownym czasie jest najprawdopodobniej niemożliwe. Często najlepszym znanym rozwiązaniem jest sprawdzenie po prostu wszystkich możliwych kombinacji.

Problem znajdowania cyklu Hamiltona jest problemem NP-zupełnym, a problem izomorficzności dwóch grafów jest uważany za problem pośredni. Oznacza to, że dla obydwu tych problemów nie są znane algorytmy rozwiązujące je w krótkim czasie (ani nawet w ciągu lat czy wieków), jeżeli tylko mamy dostatecznie dużo wierzchołków i krawędzi. Ten właśnie problem matematyczny można wykorzystać do uwierzytelniania.

A teraz właściwa idea naszego uwierzytelniania. Przypuśćmy, że stroną, która chce udowodnić swoją tożsamość, jest klient banku. Początkowo obydwie strony ustalają graf G w taki sposób, aby klient znał cykl Hamiltona w tym grafie.

W ten sposób przygotowany graf G oraz cykl Hamiltona w nim są „kluczem" potwierdzającym tożsamość klienta. Jednak schemat uwierzytelniania wygląda nieco zaskakująco:

1. Klient wysyła do banku graf H będący losową permutacją grafu G;

2. Bank losowo odsyła klientowi liczbę 0 lub 1;

3. Klient, jeżeli otrzymał 0, udowadnia bankowi, że graf H jest permutacją grafu G, jeżeli zaś otrzymał 1, pokazuje cykl Hamiltona w grafie H;

4. Cała procedura jest powtarzana dowolną liczbę razy.

Dowód faktu, że G jest permutacją H, może być zrealizowany przez wysłanie permutacji przeprowadzającej graf H na G. Zauważmy, że samo sprawdzenie, czy dwa grafy są izomorficzne, gdy dysponujemy taką permutacją, jest bardzo proste i szybkie. Klient zna cykl Hamiltona w grafie G – zna go więc również w grafie H – bo to on permutował graf. On też jest w stanie pokazać, że graf H jest permutacją grafu G.

Przypuśćmy teraz, że oszust, który wcześniej podsłuchał transmisję, chce się podszyć pod klienta. Zna już szereg grafów $H_1, H_2, \ldots, H_n$. Do każdego z nich zna dokładnie jedną z informacji: albo permutację przeprowadzającą G na H_i, albo cykl Hamiltona w grafie H_i. Statystycznie pozna więc sam graf G, ale nie pozna w nim cyklu Hamiltona – ponieważ klient nigdy nie podaje jednocześnie obydwu informacji. Oszust potrafi więc wygenerować dla banku graf F, będący permutacją grafu G albo wygenerować graf J, w którym będzie znał cykl Hamiltona. Jednak nie jest w stanie połączyć tych dwóch faktów – to znaczy wygenerować grafu X, który byłby permutacją grafu G i w którym jednocześnie znałby cykl Hamiltona. Jednocześnie nie wie, czy bank odeśle mu 0 czy 1. Tak więc szansa na to, że mu się uda, wynosi $\frac{1}{2}$. Przy n-krotnym powtórzeniu całej procedury szansa ta wynosi $\frac{1}{2^n}$. A zatem już po 30 rundach zmaleje do mniej niż jeden do miliona. A n możemy ustalić na przykład na 100, co da nam niezawodność wystarczającą do praktycznie dowolnych celów.