ROCZNIKI POLSKIEGO TOWARZYSTWA MATEMATYCZNEGO
Seria II: WIADOMOŚCI MATEMATYCZNE XLI (2005)

Witold Kraśkiewicz (Toruń)

Madhu Sudan i teoria kodowania

Począwszy od 1982 roku medalowi Fieldsa towarzyszy nagroda im. Rolfa Nevanlinny, przyznawana młodym naukowcom za rezultaty w badaniach nad matematycznymi aspektami informatyki. W roku 2002 na kongresie w Pekinie nagrodę tę odebrał Madhu Sudan, matematyk i informatyk pochodzący z Indii, ale pracujący w USA. Przyznano mu ją za prace dotyczące trzech zagadnień informatyki teoretycznej: dowodów sprawdzalnych probabilistycznie, nieaproksymowalności pewnych zagadnień optymalizacji i teorii kodów korygujących błędy.

Pierwsze dwa zagadnienia wiążą się z fundamentalnym, rozsławionym przez Clay Mathematics Institute jako jeden z siedmiu tzw. problemów milenijnych pytaniem, czy klasa P problemów rozwiązywalnych w czasie wielomianowym jest identyczna z klasą NP problemów, dla których istnieją certyfikaty sprawdzalne w czasie wielomianowym[1]. Zwięzłe omówienie dorobku Madhu Sudana dotyczącego tych zagadnień można znaleźć w [7]. Tutaj zajmiemy się trzecim z wymienionych zagadnień.

W modelu cyfrowym informacją jest wybór jednego ze skończonej liczby możliwych elementów: np. odpowiedź tak lub nie, jeden z 256 kolorów opisujących plamkę na ekranie itd. Tego typu informacje chcemy przesyłać (lub zapisywać), korzystając z pewnego urządzenia fizycznego zwanego kanałem. Załóżmy, że urządzenie to może wysyłać ciągi sygnałów należących do pewnego skończonego zbioru $\mathbb{F}$ zwanego alfabetem. Elementy zbiory $\mathbb{F}$ będziemy nazywać literami. Bardzo często zbiór ten składa się z dwóch zaledwie liter oznaczanych symbolami 0 i 1, ale jak zobaczymy dalej, przyjęcie takiego ograniczenia jest zbyt restrykcyjne. Niech q oznacza liczbę liter w alfabecie.

Aby użyć takiego kanału do przesyłania informacji, musimy umówić się, w jaki sposób interesującą nas informację przedstawić w postaci ciągu liter alfabetu $\mathbb{F}$. Przyporządkowanie to nazywamy kodowaniem. Przykładem

[1] Por. L. Pacholski, *Pierwszy problem milenijny: czy $NP = P$?*, Wiadomości Matematyczne 40 (2004), str.1–21. (Przyp. Redakcji)

takiego kodowania jest alfabet Morse'a, który każdej literze alfabetu angielskiego przyporządkowuje ciąg złożony z sygnałów krótkich i długich zakończony przerwą międzyliterową (a więc $q = 3$). Innym kodowaniem jest kod ASCII, który każdemu znakowi klawiatury komputerowej przypisuje ciąg zero-jedynkowy długości 7. W dalszym ciągu będziemy zakładali, że wszystkie ciągi liter alfabetu $\mathbb{F}$ reprezentujące informację mają tę samą długość n.

Po przejściu przez kanał otrzymujemy nadane słowo i odwracając proces kodowania, czyli dekodując otrzymane słowo, odzyskujemy wyjściową informację. Niestety, wszystkie rzeczywiste urządzenia fizyczne są w większym lub mniejszym stopniu zawodne. Skutkiem tego wysyłając kanałem literę $a \in \mathbb{F}$, możemy w pewnych sytuacjach otrzymać literę $a' \neq a$. Mamy wówczas do czynienia z błędem transmisji. Jeżeli każde słowo długości n nad alfabetem $\mathbb{F}$ reprezentuje pewną informację, to w wyniku dekodowania odbiorca otrzymuje inną informację niż wysłana przez nadawcę.

Rozwiązaniem okazuje się tak zwane kodowanie nadmiarowe: do reprezentowania informacji używamy nie wszystkich słów długości n nad alfabetem, a jedynie pewnych, najczęściej mających pewną łatwo sprawdzalną własność. Zbiór ciągów, które w ustalonym kodowaniu reprezentują informacje nazywamy kodem i będziemy go oznaczać symbolem C, a jego elementy nazywamy słowami kodowymi.

Pierwszym rozwiązaniem praktycznym stosowanym od początków techniki cyfrowej był tak zwany kod kontroli parzystości złożony z parzystych słów binarnych określonej długości, czyli takich słów, w których liczba jedynek jest parzysta. Jeżeli założyć, że błędy w trakcie transmisji pojawiają się w słowie w sposób losowy i prawdopodobieństwo p przekłamania pojedynczej litery jest małe, to najbardziej prawdopodobny błąd będzie polegał na przekłamaniu pojedynczej litery w słowie i zostanie wykryty przy próbie dekodowania. Możemy wówczas na przykład zażądać powtórnej transmisji.

Okazuje się, że korzystając z odpowiednich kodów możemy nie tylko wykryć pewne błędy, ale też je naprawić. Załóżmy, że każde dwa słowa kodowe kodu C różnią się na co najmniej d miejscach. Najmniejszą liczbę d o tej własności nazywamy minimalną odległością kodu C. Zmodyfikujmy procedurę dekodowania w sposób następujący: jeżeli słowo otrzymane należy do kodu C, to będziemy zakładać, że jest to słowo wysłane. W przeciwnym wypadku szukamy słowa kodowego, które różni się od słowa otrzymanego na co najwyżej $e = \lfloor \frac{d-1}{2} \rfloor$ miejscach. Jeżeli jest takie słowo (zauważmy, że jest co najwyżej jedno takie słowo), to uznajemy, że właśnie ono zostało nadane. W przeciwnym wypadku deklarujemy błąd (lub jeśli zmuszają nas do tego okoliczności, dekodujemy jakkolwiek). Jeżeli liczba pojedynczych błędów w trakcie transmisji słowa kodowego nie przekroczyła e, to powyższa procedura pozwala naprawić błędy transmisji. Mówimy więc, że kod C koryguje e błędów.

Możliwość korygowania błędów okupiona jest pewnym spowolnieniem przesyłania informacji. Załóżmy dla prostoty, że informacja którą będziemy przekazywać, jest zapisana przy pomocy słów k-literowych nad tym samym alfabetem $\mathbb{F}$, którego używamy do przekazywania informacji przez kanał, przy czym każdy z q^k ciągów jest równie prawdopodobnym komunikatem. Kodowanie jest wówczas pewną funkcją różnowartościową $c : \mathbb{F}^k \longrightarrow \mathbb{F}^n$, a kod C jest obrazem funkcji c. Jeżeli chcemy, aby każde dwa słowa kodowe różniły się na co najmniej d miejscach, to jak łatwo zauważyć musi zachodzić $n - d + 1 \geq k$. Liczba $R = \frac{k}{n}$ jest miarą spowolnienia transmisji danych, a liczba $\delta = \frac{d}{n}$ jest miarą zdolności korekcyjnych kodu.

Osobą, która prawdopodobnie jako pierwsza zwróciła uwagę na możliwość korekcji błędów, był Richard Hamming. Powodowany względami praktycznymi, skonstruował interesujące przykłady kodów, w tym nazywany dzisiaj jego nazwiskiem kod o parametrach $q = 2$, $n = 7$, $k = 4$. Jednak za początek teorii należy chyba uznać fundamentalną pracę Claude'a Shannona [3] z 1948 roku. Udowodnił on, że o ile zgodzimy się na prędkość transmisji mniejszą od pewnej wartości krytycznej zależnej jedynie od prawdopodobieństwa p, to wybierając odpowiedni sposób kodowania, można prawdopodobieństwo błędnego zdekodowania uczynić dowolnie małym. Niestety, dowód Shannona był nieefektywny: nie dawał żadnej wskazówki, jak w praktyce znaleźć odpowiednie kody.

Względy praktyczne stawiają więc następujący problem kombinatoryczny: dla zadanych q, k oraz n znaleźć kod $C \subset \mathbb{F}^n$ o q^k słowach kodowych z możliwie największą minimalną odległością d. To i pokrewne jemu zagadnienie znalezienia ograniczeń, jakim podlegają podstawowe parametry n, k i d opisujące kod, na wiele lat wyznaczyły główny kierunek rozwoju teorii kodów korygujących błędy. Skonstruowano wiele rodzin kodów, sięgając nieraz do bardzo zaawansowanych teorii matematycznych.

Tutaj poprzestaniemy na przedstawieniu kodu skonstruowanego przez Irvinga Reeda i Gustave'a Solomona w roku 1960. Będziemy zakładać, że alfabet $\mathbb{F}$ ma strukturę ciała. Dla ciągu $a = (a_0, a_1, \ldots, a_{k-1})$ długości k elementów z $\mathbb{F}$ symbolem f_a będziemy oznaczać wielomian $f_a = a_0 + a_1 X + \ldots + a_{k-1} X^{k-1}$. Niech $n \leq q$ i wybierzmy n różnych elementów $x_1, x_2, \ldots, x_n$ z ciała $\mathbb{F}$. Zdefiniujmy odwzorowanie kodujące $c : \mathbb{F}^k \to \mathbb{F}^n$ za pomocą ciągu wartości wielomianu f_a:

$$a \;\mapsto\; c(a) = (f_a(x_1),\, f_a(x_2), \ldots,\, f_a(x_n)). \tag{1}$$

Obraz C tego odwzorowania jest podprzestrzenią liniową w $\mathbb{F}^n$. Ponieważ wielomian stopnia co najwyżej $k - 1$ może mieć co najwyżej $k - 1$ pierwiastków, więc każde dwa słowa z C różnią się na co najmniej $n - k + 1$ miejscach. Dla $k < n$ otrzymujemy w ten sposób kod o parametrach n, k i $d = n - k + 1$ zwany kodem Reeda–Solomona. Używając tego kodu, możemy korygować więc nie więcej niż $e = \lfloor \frac{n-k+1}{2} \rfloor$ błędów. W rzeczywistości

powyższa konstrukcja jest uogólnieniem oryginalnej konstrukcji Reeda i Solomona.

Dzisiaj kody Reeda–Solomona są częścią wielu standardów technicznych. Korzysta się z nich na przykład przy zapisie informacji na dyskach optycznych i magnetycznych i w telekomunikacji satelitarnej. Co zadecydowało o wyborze tych kodów? Aby w praktyce zastosować dany kod, nie wystarczają duża szybkość transmisji i dobre własności korekcyjne (a więc duże wartości $\frac{k}{n}$ oraz $\frac{d}{n}$). Potrzeba jeszcze efektywnego i prostego w implementacji algorytmu dekodowania. W przypadku kodów Reeda–Solomona algorytmem takim okazał się algorytm oparty na pomyśle Elwyna Berlekampa, a w szczególności wariant tego algorytmu opatentowany w roku 1986 przez Lloyda Welcha i Berlekampa [6]. Algorytm ten w istotny sposób wykorzystuje nieopisane tu szczegóły oryginalnej konstrukcji kodu i nie nadaje się do dekodowania uogólnionych kodów Reeda–Solomona.

W roku 1997 Madhu Sudan opublikował w pracy [4] zupełnie nowy algorytm dekodowania kodów Reeda–Solomona. Aby przedstawić jego ideę, przyjrzyjmy się danym, którymi dysponuje osoba przystępująca do dekodowania. Zna ona otrzymany ciąg $(y_1, \ldots, y_n)$ oraz wszystkie szczegóły procesu kodowania, więc także ciąg $(x_1, \ldots, x_n)$. Poszukuje natomiast wielomianu $f \in \mathbb{F}[X]$, którego stopień jest ograniczony z góry przez $s = k - 1$ i którego wykres przechodzi przez maksymalnie wiele spośród punktów $(x_i, y_i) \in \mathbb{F} \times \mathbb{F}$, $i = 1, \ldots, n$. Gdyby odrzucić ograniczenie na stopień, można by z łatwością znaleźć wielomian o wykresie przechodzącym przez wszystkie punkty (x_j, y_i). Niestety, nie powie on nam nic o poszukiwanym wielomianie z wyjątkiem mało interesującego przypadku, gdy w czasie transmisji nie popełniono żadnych błędów.

Istota pomysłu Sudana polega na zastąpieniu wielomianu jednej zmiennej wielomianem dwóch zmiennych $F = \sum_{i,j} F_{i,j} X^i Y^j \in \mathbb{F}[X, Y]$, takim że krzywa o równaniu $F(X, Y) = 0$ przechodzi przez wszystkie punkty (x_i, y_i), $i = 1, \ldots, n$. Wszystkie takie wielomiany można znaleźć, rozwiązując liniowy układ równań względem nieznanych współczynników $F_{i,j}$. Okazuje się, że przy dowolnym s i przy $t > \sqrt{2ns}$ można tak dobrać postać wielomianu F (narzucając warunki typu $F_{i,j} = 0$ dla pewnych i, j), aby każda krzywa $F(X, Y) = 0$ tej postaci przechodząca przez wszystkie punkty (x_i, y_i), $i = 1, \ldots, n$, spełniała następujący warunek: jeśli dla pewnego wielomianu f stopnia co najwyżej s jego wykres przechodzi przez co najmniej t spośród punktów (x_i, y_j), to wielomian F jest podzielny przez wielomian $Y - f(X)$. Wynika stąd następujący algorytm:

1. znajdujemy dowolny wielomian F żądanej postaci,
2. rozkładamy wielomian F na czynniki nierozkładalne nad ciałem $\mathbb{F}$,
3. dla każdego czynnika nierozkładalnego postaci $Y - f(X)$ sprawdzamy, dla ilu wskaźników i zachodzi $y_i = f(x_i)$ i jeśli liczba ta przekracza t, dopisujemy wielomian f do listy wyników.

Jak widać, algorytm jest czysto algebraiczny, a jego czas pracy zależy wielomianowo od parametrów n, s i q.

Biorąc $d = k-1$ i $t = \frac{n+k-1}{2}$, otrzymujemy algorytm dekodowania kodów Reeda–Solomona, ale wybierając mniejsze wartości parametru t, otrzymujemy znacznie więcej, niż oczekuje się zwykle w teorii kodowania. Nawet jeśli liczba błędów przewyższa liczbę $e = \lfloor \frac{n-k+1}{2} \rfloor$, to potrafimy podać listę najbardziej prawdopodobnych słów nadanych. Jeżeli lista ta nie jest zbyt długa, otrzymujemy użyteczną informację.

Bardziej geometryczne podejście do krzywej $F(X, Y) = 0$ (rozważenie jej krotności w punktach (x_i, y_i)) zaowocowało w [2] analogicznym algorytmem przy zmniejszeniu wartości parametru t do $t > \sqrt{sn}$.

Idea, by w przypadku liczby błędów przekraczającej połowę minimalnej odległości kodu dekodować słowo poprzez podanie listy najbliższych mu słów kodowych (ang. *list decoding*), nie jest nowa. Została sformułowana przez Petera Eliasa już w latach pięćdziesiątych, ale algorytm Sudana jest pierwszym efektywnym algorytmem tego typu. Wkrótce potem zarówno w pracach Sudana i współpracowników jak i w pracach innych autorów pojawiły się analogiczne algorytmy dla innych rodzin kodów, w tym dla kodów arytmetycznych opartych na chińskim twierdzeniu o resztach.

O ile mi wiadomo, algorytmy dekodowania poprzez listę na razie nie zostały bezpośrednio zastosowane w praktyce, ale znalazły zastosowanie w teorii złożoności. Nie jest to pierwsze użycie teorii kodowania do badania złożoności obliczeń. Znane jest zastosowanie pewnych nierówności dla kodów korygujących błędy do oszacowania liczby operacji niezbędnych do wymnożenia dwóch macierzy binarnych, przy czym nie chodzi tu o powszechnie znany, „definicyjny" sposób mnożenia wiersza przez kolumnę, ale o jakikolwiek znany lub nieopisany jeszcze algorytm realizujący mnożenie.

Zastosowania algorytmu Sudana idą w trochę innych kierunkach. Tutaj zajmiemy się tzw. funkcjami jednokierunkowymi. Mówiąc niezbyt precyzyjnie, funkcję f nazywamy jednokierunkową, jeśli mając dany argument x łatwo jest obliczyć jego obraz $f(x)$, ale dla zadanego y z obrazu funkcji praktycznie niemożliwe jest obliczenie x, dla którego $y = f(x)$. Narzędziem pozwalającym zarówno teoretycznie badać jak i praktycznie wykorzystywać funkcje jednokierunkowe są swoiste „certyfikaty nieodwracalności" (po angielsku zwane *hard-core predicate*). Załóżmy, że f jest pewną funkcją jednokierunkową przekształcającą ciągi k-bitowe w ciągi k-bitowe. Takim certyfikatem jest funkcja b przyporządkowująca ciągowi k-bitowemu x pojedynczy bit $b(x)$ w ten sposób, że łatwo jest obliczyć $b(x)$ znając x, ale prawdopodobieństwo zgadnięcia $b(x)$ na podstawie $f(x)$ jest małe (powiedzmy mniejsze od $1/2 + \varepsilon$). Tego typu certyfikaty zostały znalezione dla pewnych kryptograficznie ważnych funkcji jednokierunkowych przez Manuela Bluma i Silvio Micaliego w początku lat osiemdziesiątych, ale zaskoczeniem było skonstruowanie takich certyfikatów dla dowolnej funkcji jednokierunkowej.

Następująca konstrukcja pochodzi z pracy [5] i jest uogólnieniem konstrukcji zaproponowanej przez Odeda Goldreicha i Leonida Levina. Załóżmy, że dany jest pewien kod C i kodowanie przyporządkowujące k-bitowemu słowu x słowo n-bitowe $c(x) \in C$. Niech dalej r będzie pewną „losową" liczbą z przedziału $1 \leq r \leq n$. Weźmy funkcję b przyporządkowującą ciągowi x bit o numerze r w słowie $c(x)$. Okazuje się, że jeśli n zależy wielomianowo od k i dla kodu C istnieje efektywny algorytm dekodowania poprzez listę, naprawiający co najmniej $(1/2+\varepsilon)n$ błędów, to odwzorowanie b ma żądane własności. Co więcej, kody takie zawsze można znaleźć.

Praktyczne zastosowanie znajdują certyfikaty nieodwracalności w konstrukcji bezpiecznych generatorów pseudolosowych. Są to algorytmy, które z krótkiego ciągu mniej lub bardziej „losowych" danych produkują znacznie dłuższy ciąg, który postronnemu obserwatorowi powinien wydawać się całkowicie nieprzewidywalny. Algorytmy takie są niezwykle istotne w kryptografii. Otóż jeżeli f i b są takimi funkcjami jak wyżej i x jest k-bitowym „zarodkiem losowości", to ciąg $z_n = b(f^n(x))$, $n = 1, 2, \ldots$, jest ciągiem pseudolosowym.

Literatura

[1] D. Coppersmith, M. Sudan, *Reconstructing curves in three (and higher) dimensional spaces from noisy data*, Proceedings of the 35th Annual ACM Symposium on Theory of Computing, San Diego, California, 9–11 June, 2003.

[2] V. Gurusvami, M. Sudan, *Improved decoding of Reed–Solomon and algebraic–geometry codes*, IEEE Trans. Information Theory **45** (1999), 1757–1767.

[3] C. E. Shannon, *A mathematical theory of communication*, Bell System Technical Journal **27** (1948), (s. 379–423, 623–656).

[4] M. Sudan, *Decoding of Reed Solomon codes beyond the error correction bound*, J. Complexity **13** (1997), 180–193.

[5] M. Sudan, *List decoding: algorithms and applications, Proceedings of the International Conference IFIP TCS* 2000, Sendai, Japan, LN in Comp. Sc. 1872, Springer (2000), 25–41.

[6] L. Welch, E. Berlekamp, *Error correction of algebraic block codes*, US Patent Number 4 633 470 (1986).

[7] A. Wigderson, *On the work of Madhu Sudan*, Notices AMS **50** (2003), 45–50.