ՀՏԴ 004.02

Կոդավորման կիրառություն

Սուրեն Էքսուզյան

DOI: https://doi.org/10.58726/27382923-ne2023.1-59

Հանգուցային բառեր. *հաղորդագրության կոդ, կոդավորման փոխմիարժեքություն, ենթաբազմության կոդ, դիսկրետ օպտիմալացման խնդիր, բազմության տրոհում, ուսապարկի խնդիր, տարբերակների դիտարկման ալգորիթմ*

Ներածություն

Կոդավորումը կարևոր դեր ունի մաթեմատիկայում: Այն հնարավորություն է տալիս որևէ բնույթի օբյեկտների ուսումնասիրությունը բերել այլ բնույթի օբյեկտների ուսումնասիրության:

Մաթեմատիկայում կոդավորման կարևոր օրինակ է երկրաչափական օբյեկտների կոդավորումը անալիտիկ արտահայտությունների միջոցով, թվերի ներկայացումը հաշվարկման տարբեր համակարգերում: Կոդավորումը կարևոր կիրառություն ունի կառավարման համակարգերում:

Կոդավորումը հնարավորություն է տալիս նաև ապահովելու տեղեկատվության գաղտնիություն, տեղեկատվության առաքման և ստացման տեխնիկական հարմարավետություն: Հարմար կոդավորման միջոցով հաճախ հնարավոր է լինում հեշտությամբ լուծել տարբեր բնագավառների խնդիրներ:

Կոդավորումը կարևոր դեր ունի կոմբինատոր խնդիրներ լուծելիս: Այն թույլ է տալիս որևէ բազմության տարրերի քանակի հաշվման խնդիրը բերել կոդերի քանակի հաշվման խնդրի: Դրա համար անհրաժեշտ է բազմության տարրերին համապատասխանեցնել կոդավորման որևէ այբուբենի կոդեր այնպես, որ տարրերի և կոդերի բազմությունների միջև ստեղծվի 1-1 համապատասխանություն: Տարրերի վերջավոր բազմության դեպքում դա նշանակում է, որ տվյալ բազմության տարրերի քանակը հավասար է կոդերի քանակին: Հետևաբար, այդ բազմության տարրերի քանակի հաշվման փոխարեն կարելի է հաշվել կոդերի քանակը, որը հաճախ կարելի է իրականացնել ավելի հեշտությամբ:

Բազմաթիվ խնդիրներում տրված բազմության տարրերի դիտարկումը կարելի է իրականացնել տարրերին համապատասխան կոդերի դիտարկմամբ:

Հիմնական գաղափարներ

Դիտարկենք կամայական U={a_1,a_2,…,a_m} այբուբեն: Այբուբենի բոլոր բառերի բազմությունը նշանակենք S(U): Յուրաքանչյուր S′$\subseteq$S(U) ենթաբազմություն կանվանենք հաղորդագրությունների բազմություն, իսկ նրա յուրաքանչյուր բառ ընդունենք որպես հաղորդագրություն:

Դիտարկենք կամայական V={b_1,b_2,…, b_k} այբուբեն: V այբուբենի բոլոր բառերի բազմությունը նշանակենք S(V): Դիտարկենք F արտապատկերում, որը յուրաքանչյուր A∈S′ բառի համապատասխանեցնում է B=F(A) բառը, որտեղ B∈S(V): V այբուբենը կանվանենք կոդավորման այբուբեն, B բառը՝ A հաղորդագրության կոդ, A հաղորդագրությունից կոդի անցումը՝ կոդավորում: Կոդավորման տեսության մեջ F արտապատկերումը տրվում է որոշակի ալգորիթմով: Այն կարելի է տալ նաև կոդավորման սխեմայով:

Ապա կոդավորումը հաղորդագրության կոդից հաղորդագրության վերականգնման գործընթաց է: Հաղորդագրությունների համար ապակոդավորումը հնարավոր է, եթե գոյություն ունի F^{-1} հակադարձ արտապատկերում:

Շատ կարևոր է, որ կոդավորումը և ապակոդավորումը լինեն միարժեք (կոդավորման փոխմիարժեքություն): Սովորաբար որպես կոդավորում ընտրում են միարժեք կոդավորում: Կարևոր է, որ համապատասխան ապակոդավորումը ևս լինի միարժեք: Մի շարք կոդավորումների համար ապացուցված են կոդավորման փոխմիարժեքության հայտանիշներ [2, 174-180]:

Բազմության ենթաբազմությունների կոդավորում

Դիտարկենք կամայական B={b_1,b_2,…, b_n} բազմություն: B բազմության ենթաբազմություններին համապատասխանեցնենք կոդեր (բառեր) {0,1} այբուբենից այնպես, որ յուրաքանչյուր ենթաբազմությանը համապատասխանի միակ կոդ, և յուրաքանչյուր կոդի համապատասխանի միակ ենթաբազմություն: Այլ կերպ ասած՝ ստեղծենք 1-1 համապատասխանություն B բազմության ենթաբազմությունների բազմության և կոդերի բազմության միջև:

Յուրաքանչյուր B′$\subseteq$B ենթաբազմությանը համապատասխանեցնենք $\beta_1\beta_2\ldots\ \beta_n$ կոդ {0, 1} այբուբենից, որտեղ β_i (i=1,2,…,n) տարրերը որոշվում են հետևյալ կերպ՝

$$\beta_i = \begin{cases} 1, \text{եթե } b_i \in B' \\ 0, \text{հակառակ դեպքում} \end{cases} \qquad (1)$$

Ակնհայտ է, որ նշված ձևով յուրաքանչյուր $B' \subseteq B$ ենթաբազմությանը միակ ձևով համապատասխանում է {0,1} այբուբենից n երկարությամբ $\beta_1\beta_2\ldots\beta_n$ կոդ: Ճիշտ է նաև հակառակը: {0,1} այբուբենի n երկարությամբ յուրաքանչյուր $\beta_1\beta_2\ldots\beta_n$ կոդի միակ ձևով համապատասխանում է B բազմության ենթաբազմություն: Նշված եղանակով տարբեր ենթաբազմությունների կհամապատասխանեն տարբեր կոդեր [1, 24]:

$\beta_1\beta_2\ldots\beta_n$ կոդից ենթաբազմության ստացումը կատարվում է հետևյալ կերպ. ստուգվում է β_i (i=1,2,…,n) արժեքը: Եթե β_i=1, ապա $b_i \in B$ տարրը ընտրվում է որպես կոդին համապատասխան ենթաբազմության տարր: Եթե β_i=0, ապա $b_i \in B$ տարրը չի ընտրվում որպես կոդին համապատասխան ենթաբազմության տարր:

Բազմության ենթաբազմությունների դիտարկման ալգորիթմ

Քանի որ բազմության ենթաբազմությունների բազմության և այդ ենթաբազմությունների նշված ձևով կոդավորմամբ ստացվող կոդերի բազմության միջև ստեղծված է 1-1 համապատասխանություն, ապա որոշակի կարգով այդ բոլոր կոդերի դիտարկումը կարելի է կիրառել բազմության բոլոր ենթաբազմությունների դիտարկման համար:

n տարր պարունակող բազմության ենթաբազմությունների քանակն է 2^n: Այդ ենթաբազմություններին համապատասխանող կոդերը հաշվարկման 10-ական համակարգի 0-ից մինչև $2^n - 1$ թվերի n-նիշ 2-ական կոդերն են:

Այդ կոդերը հաջորդաբար դիտարկելու համար բավական է դիտարկել 0 թվի n-նիշ 2-ական ներկայացումը և հաջորդ յուրաքանչյուր քայլում ընթացիկ կոդին 2-ական հաշվարկման համակարգում գումարել 1 նիշը: Բոլոր կոդերը դիտարկելու համար անհրաժեշտ է 1 նիշի 2-ական գումարում կատարել $2^n - 1$ անգամ: Ելակետային կոդին կհամապատասխանի դատարկ բազմությունը: Կոդերի դիտարկման ընթացքում յուրաքանչյուր կոդին անհրաժեշտ է համապատասխանեցնել դրան համապատասխան ենթաբազմությունը:

Արդյունքում կունենանք բազմության ենթաբազմությունների դիտարկման հետևյալ ալգորիթմը.

Ալգորիթմ: (Բազմության ենթաբազմությունների դիտարկման ալգորիթմ):

Մուտք: Բազմության տարրերի քանակը՝ n, {b_1,b_2,…,b_n} բազմություն:

Ելք: {b_1,b_2,…, b_n} բազմության բոլոր ենթաբազմությունները:

Քայլ 1: Որպես ենթաբազմության կոդ ընտրել n տարր պարունակող

W=$\underbrace{00\ldots00}$ կոդը: Այդ կոդին համապատասխանեցնել դատարկ բազմությունը:

Քայլ 2: Վերագրել i ← 1:

Քայլ 3: Որպես հերթական ենթաբազմություն ընտրել A=∅ բազմությունը:

Քայլ 4: W կոդի նոր արժեք համարել W կոդի ընթացիկ արժեքին 2-ական համակարգում 1 նիշը գումարելով ստացված կոդը:

Քայլ 5: Վերագրել j ← 1:

Քայլ 6: Եթե W կոդի ձախից j-րդ կարգանիշն ունի 1 արժեքը, ապա A ենթաբազմությանը ավելացնել b_j տարրը:

Քայլ 7: Վերագրել j ← j +1:

Քայլ 8: Եթե j ≤ n, ապա անցնել Քայլ 6-ին:

Քայլ 9: Արտածել A ենթաբազմությունը:

Քայլ 10: Վերագրել i ← i +1:

Քայլ 11: Եթե i $\le 2^n - 1$, ապա անցնել Քայլ 3-ին:

Քայլ 12: Ալգորիթմի աշխատանքն ավարտել:

n=4, B={b_1,b_2,b_3,b_4} տվյալների համար ալգորիթմի աշխատանքի արդյունքում ստացվել են հետևյալ ենթաբազմությունները. ∅, {b_4}, {b_3}, {b_3,b_4}, {b_2}, {b_2,b_4}, {b_2,b_3}, {b_2,b_3,b_4}, {b_1}, {b_1,b_4}, {b_1,b_3}, {b_1,b_3,b_4}, {b_1,b_2}, {b_1,b_2,b_4}, {b_1,b_2,b_3}, {b_1,b_2,b_3,b_4}:

Բազմության ենթաբազմությունների դիտարկման ալգորիթմի կիրառություն

Բազմաթիվ խնդիրներում անհրաժեշտություն է առաջանում խնդրի լուծումը փնտրել որևէ բազմության ենթաբազմությունների մեջ: Նման խնդիրներում անհրաժեշտ է դիտարկել տվյալ բազմության բոլոր ենթաբազմությունները և որոշակի հայտանիշով որպես լուծում ընտրել այդ ենթաբազմություններից լավագույնը:

Բազմության ենթաբազմությունների դիտարկման խնդիր է առաջանում ուսապարկի խնդիր, բազմության տրոհման վերաբերյալ խնդիրներ և այլ խնդիրներ լուծելիս:

Նշված ընդհանուր խնդրին են բերվում բազմաթիվ բնագավառներում առաջացող շատ կոնկրետ խնդիրներ:

Նման խնդիրների լուծման համար կարելի է օգտվել աշխատանքում բազմության ենթաբազմությունների դիտարկման համար մշակված ալգորիթմից:

Եզրակացություն

Դիսկրետ օպտիմալացման խնդիրներում խնդրի օպտիմալ լուծման որոնում կատարվում է որոշակի բազմության տարրերից: Տարրերի համեմատության համար նախապես ֆիքսվում է որոշակի որակի հայտանիշ (հայտանիշներ): Նման որոնում հաճախ կատարվում է որևէ բազմության ենթաբազմությունների բազմությունից, բազմության տարրերի տեղափոխությունների բազմությունից կամ որևէ այլ բազմությունից: Հետևաբար, անհրաժեշտություն է առաջանում բազմության տարրերի՝ ըստ այդ հայտանիշի համեմատության համար սահմանել տարրերի դիտարկման որոշակի կարգ, այլ կերպ ասած՝ մշակել բազմության տարրերի դիտարկման ալգորիթմ: Այն դեպքում, երբ խնդրի լուծումը որոնում են որևէ բազմության ենթաբազմությունների մեջ, անհրաժեշտություն է առաջանում օգտվել բազմության բոլոր ենթաբազմությունների դիտարկման ալգորիթմից:

Աշխատանքում մշակված է բազմության բոլոր ենթաբազմությունների դիտարկման ալգորիթմ: Ալգորիթմի մշակման համար դիտարկված է բազմության ենթաբազմությունների կոդավորում:

DOI: https://doi.org/10.58726/27382923-ne2023.1-59

Գրականություն

1. Տոնոյան Ռ., Դիսկրետ մաթեմատիկայի դասընթաց, Երևան, 1999, 129 էջ:
2. Яблонский С. Введение в дискретную математику, Москва, Наука, 1979, 272 с.

Применение кодирования

Сурен Эксузян

Резюме

Ключевые слова: *код сообщения, взаимнооднозначность кодирования, код подмножества, задача дискретной оптимизации, разбиение множества, задача рюкзака, алгоритм перечисления элементов*

Работа посвящена разработке алгоритма перечисления всех подмножеств множества.

Для этой цели рассматривается удобное кодирование подмножеств множества, так, что оно создает 1-1 соответствие между множества всех подмножеств множеста и множества всех кодов.

Организация рассмотрения всех таких кодов также позволяет удобным способом рассматривать все подмножества множества.

Подмножества множества кодируются в двоичной системе числения (с двоичними числами). Проблема перечисления подмножеств множества возникает при решении задачи о рюкзаке, задачи разбиения множества и других задачах.

К упомянутым общим проблемам приводятся множество конкретных проблем, возникающие во многих областях.

В работе разработан алгоритм перечисления всех подмножеств множества, которое целесообразно употреблять для решения всех упомянутых задач.

Разработанный алгоритм может быть применен для оптимального решения NP-полных задач небольших размеров. Все известные алгоритмы оптимального решения задач больших размеров работают за практически неприемлемое время (имеют очень большое время выполнения). По этой причине для решения NP-полных задач больших размеров разрабатываются приближенные алгоритмы с полиномиальной временной сложностью, с помощью которых получаются приближенные решения задач.

Coding Application

Suren Eksuzyan

Summary

Key words: *message code, coding equivalence, subset code, discrete optimization problem, set partitioning, knapsack problem, variant observation algorithm*

The work is devoted to the development of the algorithm for observing all subsets of the set.

For that purpose, coding for subsets of the set is considered, in such a way as to create a 1-1 correspondence between the set of subsets of the set and the set of codes.

A convenient arrangement of the observation of all those codes allows the observation of all subsets of the set in a convenient way.

Subsets of a set are coded in the binary number system (with binary numbers). The problem of considering subsets of a set arises when solving the knapsack problem, set partitioning problems, and other problems.

To the mentioned general problems are included a number of specific problems arising in many fields.

In the work, an algorithm for observing all subsets of the set is developed, which is suitable for solving all the mentioned problems.

The developed algorithm can be applied to the optimal solution of non-large NP-complete problems. All the known optimal solution algorithms for solving large-scale problems run in practically unacceptable time (have very long execution time). As a consequence, for the solution of large NP-complete problems, approximate algorithms with polynomial time complexity are developed, by means of which approximate solutions of the problem are obtained.

Ներկայացվել է 19.03.2023 թ.
Գրախոսվել է 17.04.2023 թ.
Ընդունվել է տպագրության 25.05.2023 թ.