ZASTOSOWANIA MATEMATYKI
APPLICATIONES MATHEMATICAE
XIII, 4 (1973)

ALGORITHM 25

J. KUCHARCZYK (Wrocław)

DISCRETE OPTIMIZATION BY THE METHOD OF MAO AND WALLINGFORD

1. Procedure declaration. The procedure *doptMW* solves the following problem:

Find the minimum of $g(x) = g_{01}(x) - g_{02}(x)$, given $g_{11}(x) - g_{12}(x) \geqslant 0$, $g_{21}(x) - g_{22}(x) \geqslant 0, \ldots, g_{m1}(x) - g_{m2}(x) \geqslant 0$, where $x = (x_1, x_2, \ldots, x_n)$, $x_j = 0, 1$, and the functions g_{ij} $(i = 0, 1, \ldots, m;\ j = 1, 2)$ are monotonically non-decreasing with respect to every variable x_j.

Data:

n – number of zero-one variables x_j,

m – number of conditions $g_{i1}(x) - g_{i2}(x) \geqslant 0$,

f – function with procedure heading: **real procedure** $f(k, x)$; **integer** k; **integer array** x; which for $k = 0$ calculates $g_{01}(x)$, for $k = -1$ calculates $g_{02}(x)$, for $k = 1, 2, \ldots, m$ calculates $g_{k1}(x)$, and for $k = m+1, m+2, \ldots, 2m$ calculates $g_{k-m,2}(x)$, given x in $x[1:n]$,

max – maximum allowable number of type **real**.

Results:

$x[1:n]$ – the solution vector,

z – optimum value of $g(x)$; if $z = max$ on exit, then there is no feasible solution to the problem.

2. Method used. The algorithm is based on [2] and was coded according to the flow diagram given there.

3. Certification. Several examples were solved and correct results obtained. The procedure *doptMW* was also compared with the procedure *doptLB* [1] for problems with appropriate objective functions. Both procedures gave identical correct results, though *doptMW* was, as expected, considerably slower in computation than *doptLB*.

```
procedure doptMW(n,m,f,x,z,max);
 value n,m;
 integer n,m;
 real z,max;
 integer array x;
 real procedure f;
 begin
  integer i;
  real zg,gr;
  integer array xx,xg[1:n];
  array g[1:m];
  z:=max;
  for i:=1 step 1 until n do
   xx[i]:=0;
  for i:=1 step 1 until m do
   if f(i,xx)<f(m+i,xx)
    then go to ntrivsol;
  z:=f(0,xx)-f(-1,xx);
  for i:=1 step 1 until n do
   x[i]:=xx[i];
ntrivsol:
  xx[n]:=1;
repeat:
  for i:=1 step 1 until n do
   xg[i]:=xx[i];
  for i:=n step -1 until 1 do
   if xx[i]=0
    then xg[i]:=1
    else go to END;
END:
```

```
   zg:=f(0,xx);
   if zg-f(-1,xg)≥z
    then go to skip;
   for i:=1 step 1 until m do
    begin
     gr:=g[i]:=f(m+i,xx);
     if f(i,xg)<gr
      then go to skip
    end i;
   for i:=1 step 1 until m do
    if f(i,xx)<g[i]
     then go to next;
   zg:=zg-f(-1,xx);
   if z>zg
    then
    begin
     z:=zg;
     for i:=1 step 1 until n do
      x[i]:=xx[i]
    end z>zg;
   go to next;
skip:
   for i:=1 step 1 until n do
    xx[i]:=xg[i];
next:
   for i:=n step -1 until 1 do
    if xx[i]=1
     then xx[i]:=0
     else
     begin
      xx[i]:=1;
```

```
        go to repeat
        end i,xx[i]=0;
    finish:
    end doptMW
```

References

[1] J. Kucharczyk, *Algorithm 23: Lawler and Bell's method of discrete optimization*, Zastosow. Matem. 13 (1973), p. 411-414.

[2] J. C. T. Mao and B. A. Wallingford, *An extension of Lawler and Bell's method of discrete optimization with examples from capital budgeting*, Manag. Sci. 15 (1968), p. B51-B60.

DEPT. OF STATISTICS AND OPERATIONS RESEARCH
INSTITUTE OF ADMINISTRATIVE SCIENCES
UNIVERSITY OF WROCŁAW

Received on 15. 9. 1972

ALGORYTM 25

J. KUCHARCZYK (Wrocław)

OPTYMIZACJA DYSKRETNA METODĄ MAO I WALLINGFORDA

STRESZCZENIE

Procedura *doptMW* rozwiązuje następujące zagadnienie:

Znaleźć minimum funkcji $g(x) = g_{01}(x) - g_{02}(x)$ przy warunkach $g_{11}(x) - g_{12}(x) \geqslant 0$, $g_{21}(x) - g_{22}(x) \geqslant 0, \ldots, g_{m1}(x) - g_{m2}(x) \geqslant 0$, gdzie $x = (x_1, x_2, \ldots, x_n)$, $x_j = 0, 1$, oraz funkcje g_{ij} $(i = 0, 1, \ldots, m;\ j = 1, 2)$ są monotonicznie niemalejące ze względu na każdą ze zmiennych x_j.

Dane:

n — liczba zmiennych zero-jedynkowych x_j,

m — liczba ograniczeń $g_{i1}(x) - g_{i2}(x) \geqslant 0$,

f — funkcja o nagłówku: **real procedure** $f(k, x)$; **integer** k; **integer array** x; która dla $k = 0$ oblicza $g_{01}(x)$, dla $k = -1$ oblicza $g_{02}(x)$, dla $k = 1, 2, \ldots, m$ oblicza $g_{k1}(x)$ i dla $k = m+1, m+2, \ldots, 2m$ oblicza $g_{k-m,2}(x)$ przy x zadanym w $x[1:n]$,

max — największa dopuszczalna liczba rzeczywista.

Wyniki:

$x[1:n]$ — wektor rozwiązania optymalnego,

z — optymalna wartość funkcji $g(x)$; jeżeli po wyjściu z procedury $z = max$, to zagadnienie nie ma rozwiązania dopuszczalnego.

Algorytm oparty jest na metodzie podanej w [2]. Sprawdzono jego poprawność na kilku przykładach, a ponadto porównano go — dla odpowiednich funkcji celu — z wynikami otrzymanymi za pomocą procedury *doptLB* [1]. Otrzymano wyniki zgodne, chociaż — jak się tego spodziewano — czas obliczeń dla procedury *doptMW* znacznie przewyższał czas obliczeń przy użyciu procedury *doptLB*. Wszystkie obliczenia wykonano na m.c. ODRA 1204.