Управляемые системы, ИМ, ИК СО АН СССР 1969 г. выпуск 2.
УДК.519.95.

АЛГОРИТМ РЕШЕНИЯ ТРАНСПОРТНОЙ ЗАДАЧИ ПО ВРЕМЕНИ ДЛЯ ДВУХПОЛЮСНОЙ СЕТИ

В.А.Евстигнеев, В.П.Ясакова

В работе [1] была рассмотрена задача о нахождении минимального времени перевозки грузов по двухполюсной транспортной сети. Задача решалась в предположении, что все перевозки осуществляются по дугам сети дискретными порциями, не превышающими по величине пропускной способности дуг.

Было доказано, что при сделанных предположениях минимальное время перевозки груза P определяется по формуле

$$T = \left[\frac{P - 1 + \sum\limits_{u \in G} \varphi(u) t(u)}{\Phi} \right], \qquad /1/$$

если только

$$P \geq B(G),$$

где $B(G) = \sum\limits_{\mu} \varphi_{\mu}(T - t_{\mu} + 1)$ является постоянной, зависящей от строения сети. Здесь $\varphi(u)$ - оптимальный поток по сети G , под которым мы подразумеваем наибольший поток, минимизирующий функционал $\sum\limits_{u \in G} \varphi(u) t(u)$

В случае, когда $P < B(G)$, время определяется по формуле

$$T = \left[\frac{P - 1 + \sum\limits_{u \in G'} \overline{\varphi}(u) t(u)}{\overline{\Phi}} \right], \qquad /2/$$

где G' - подсеть сети G $(G' \subset G)$, для которой справедливо неравенство

$$B(G') \leq P < B(G'')$$

$(G' \subset G'' \subseteq G)$ и $\overline{\varphi}(u)$ - поток, оптимальный в указанном выше смысле по сети G' .

Цель настоящей работы - описать алгоритм для численного решения данной задачи.

Алгоритм состоит в последовательном построении сетей $G_0 \subset G_1 \subset \ldots \subset G_k$, удовлетворяющих условиям:

а/ $\Phi_0 \leq \Phi_1 \leq \ldots \leq \Phi_k \leq \Phi$;

б/ $B(G_0) \leq B(G_1) \leq \ldots \leq B(G)$;

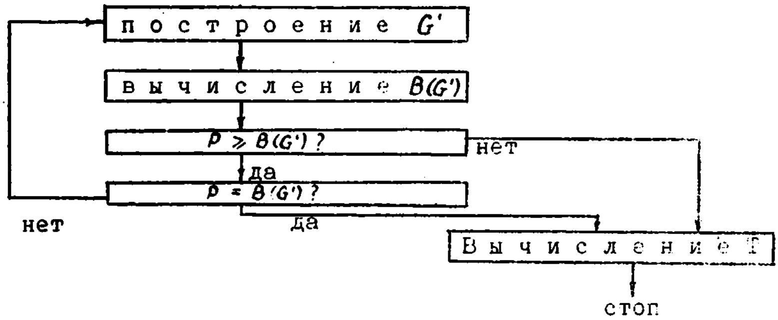
в/ $B(G_k) \leq P$.

Построение прекращается, как только для некоторого k будем иметь:

а/ $B(G_k) \leq P < B(G_{k+1})$, если $G_k \subset G$;

б/ $B(G_k) \leq P$, если $G_k = G$.

Ниже приводится блок-схема алгоритма



Рассмотрим подробнее работу алгоритма по шагам.

Построение последовательности сетей

Пусть уже имеется сеть G_i с потоком φ_i. Пусть $\widetilde{G}_i$ - сеть, которая получается из сети G, если положить

$$\bar{c}(ij) = c(ij) - \varphi_i(ij);$$

$$\bar{c}(ji) = \varphi_i(ij).$$

Выберем в $\widetilde{G}_i$ кратчайший путь μ_k из X_0 в X_n. Тогда из алгоритма построения оптимального потока Ху [2], [3] следует, что $G_{i+1} = G_i \cup \mu_k$. При этом возможны следующие случаи:

а/ μ_k не имеет общей дуги с каким-либо путем из G_i.
Тогда $G_{i+1} = G_i \cup \mu_k$, $\Phi_{i+1} = \Phi_i + \varphi_k$,

б/ μ_k имеет общую дугу $\bar{u}$ с путем μ_p $(1 \leqslant p < k)$, причем $\bar{u}$ является прямой для обоих путей. Этот случай сводится к случаю а/;

в/ $\bar{u}$ является обратной для μ_k. Тогда

$$1/\ G_{i+1} = \bigcup_{l=1}^{p-1} \mu_l \cup \mu_p \cdot \bigcup_{l=p+1}^{k-1} \mu_l \cup \mu'_k ,$$

где

$$\mu'_p = \mu_k^{нач} \cup \mu_p^{кон},$$

$$\mu'_k = \mu_p^{нач} \cup \mu_k^{кон},$$

если $\varphi_k = \varphi_p$;

$$2/\ G_{i+1} = \bigcup_{l=1}^{p-1} \mu_l \cup \mu'_p \bigcup_{l=p+1}^{k-1} \mu_l \cup \mu'_k \cup \mu_{k+1},$$

где

$$\mu'_p = \mu_k^{нач} \cup \mu_p^{кон},$$

$$\mu'_k = \mu_p^{нач} \cup \mu_k^{кон},$$

$$\mu_{k+1} = \mu_k,$$

если $\varphi_k > \varphi_p$;

3/ если же $\varphi_k < \varphi_p$, то ищем еще один путь μ_p, пересекающийся с μ_k.

Вычисление $B(G_i)$

Для построенной сети G_i величина $B(G_i)$ вычисляется по формуле

$$B(G_i) = \Phi_i(t_k + 1) - \alpha_i + 1,$$

где Φ_i и α_i определяются, соответственно, из равенств:

$$\Phi_i = \Phi_{i-1} + \varphi_k; \quad \alpha_i = \alpha_{i-1} + \varphi_k t_k. \qquad /3/$$

Вычисление T

Время $T(G_i)$ вычисляется по формуле

$$T = \left[(p - 1 + \alpha_i) / \Phi_i\right], \qquad /4/$$

где $[X]$ - есть целая часть от X, а Φ_i и α_i определены выше.

Описание программы

В программе имеются 4 вложенных друг в друга блока R_1, R_2, R_3, R_4.

Первый, R_1, выбирает кратчайший путь /при этом используется программа, описанная в [3]/.

Второй, R_2, вычисляет поток по выбранному пути, длину этого пути и преобразует матрицу A пропускных способностей по следующему правилу:

$$a_{ij} = \begin{cases} a_{ji}, & \text{если } (ji) \in \mu_k; \\ \infty, & \text{если } C(ij) = 0; \\ a_{ij} & \text{— в остальных случаях} \end{cases}$$

В этом же блоке производится вычисление времени T по формуле /4/.

Третий блок, R_3, вычисляет постоянную $B(G)$ по формуле /3/.

Четвертый блок является вспомогательным, с помощью его определяются границы динамических массивов.

Программа предусматривает использование лишь оперативной памяти ЭВМ.

При вычислении на ЭВМ М-20 в программу не вносится никаких изменений, задаются лишь исходные данные, для которых в памяти отводится 1060 ячеек.

Программа рассчитана на решение задач, для которых имеет место неравенство

$$n(2n + k + 13) + m(2k + 9) + 3k + 12 \leq 1060,$$

где $n+1, m, k$ - число соответственно вершин, ребер и допустимых путей.

Величину k можно приближенно оценить с помощью неравенства

$$k \leq \frac{\sum_i c_i}{\min_j c_j}$$

где $i \in \Gamma_{x_0}$, $j = 1, \ldots, m$.

Задание исходных данных

Перенумеруем последовательно вершины сети, начиная с входа сети, которому присваиваем номер 0, затем любую следующую вершину 1 и так далее. Выходу сети присваиваем номер n. После этого в некотором порядке нумеруем дуги сети $1, 2, \ldots, m$.

Исходные данные задаются следующим образом:

1. $n, k\Sigma$ - число вершин сети, без одной;

2. $m, k\Sigma$ - число дуг сети;

$\left.\begin{matrix} 3 \\ \vdots \\ m+2 \end{matrix}\right\}$ $\|c(j)\|, k\Sigma$ - вектор пропускных способностей $(1 \leq j \leq m)$;

$\left.\begin{matrix} m+3 \\ \vdots \\ 2m+2 \end{matrix}\right\} \|D(j)\|, k\Sigma$ – вектор длин дуг $(1 \leqslant j \leqslant m)$;

$\left.\begin{matrix} 2m+3 \\ \vdots \\ 3m+2 \end{matrix}\right\} \|n1(j)\|, k\Sigma$ – вектор, j -я компонента которого означает номер вершины, в которой начинается j -я дуга $(1 \leqslant j \leqslant m)$;

$\left.\begin{matrix} 3m+3 \\ \vdots \\ 4m+2 \end{matrix}\right\} \|k1(j)\|, k\Sigma$ – вектор, j -я компонента которого означает номер вершины, в которой кончается j -я дуга $(1 \leqslant j \leqslant m)$;

$4m+3$ $P, k\Sigma$ – величина груза.

Печать результатов

На каждом шаге итерации печатаются:

1/ k – номер итерации;

2/ $n1$ – массив из $k+2$ чисел, первое из которых 0 не имеет значения, каждое следующее из k чисел – это число дуг у выбранных путей, последнее – n ;

3/ массив выбранных путей: номера дуг, принадлежащих пути.

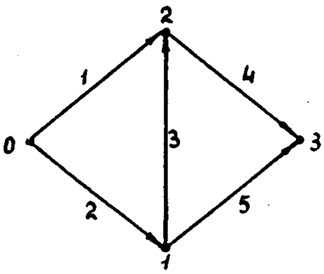
По окончании счета печатаются:

1/ постоянная B – для контроля;

2/ поток φ по сети;

3/ время перевозки T .

В качестве контрольной задачи рекомендуется использовать следующую. Исходные данные: 1/ n = 3;



2/ m = 5 ;

3/ p = 50;

4/ $C(1,1,1,1,1)$;

5/ $D(5,3,1,3,4)$;

Результат счета:

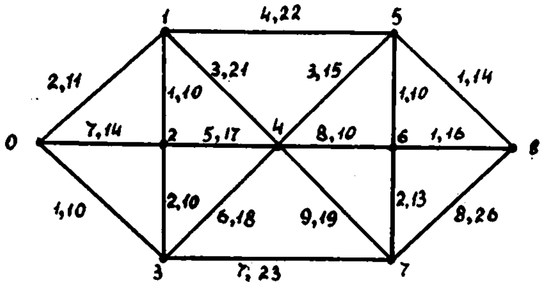
1/ $\mu_1 = \{1,4\}$;

2/ $\mu_2 = \{2,5\}$;

3/ T = 32.

По составленной программе был решен ряд задач.

Пример. Рассмотрим сеть G



$p = 50.$

Рис. 1. Сеть G

1. На первом шаге $k=1$ выбирается путь

$$\mu_1 = \{x_0, x_1, x_5, x_8\}; \quad B(G_1) = 1.$$

Сеть G_1 имеет вид

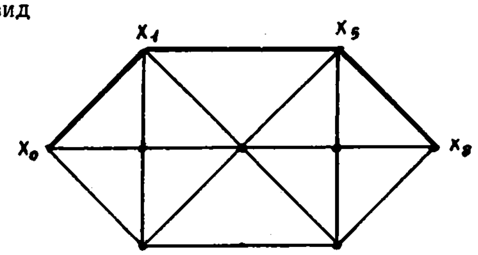


Рис. 2. Сеть G_1

2. На втором шаге $k=2$ выбирается путь

$$\mu_2 = \{x_0, x_3, x_4, x_6, x_8\}; \quad B(G_2) = 9;$$

Сеть G_2 имеет вид

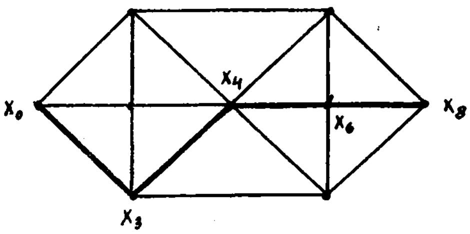


Рис. 3. Сеть G_2

3. На третьем шаге $k=3$ выбирается путь

$\mu_3=\{x_0,x_2,x_4,x_7,x_8\}$; $B(G_3)=54$ /рис. 4/; но $P=50<B(G_3)=54$, таким образом, μ_3 использовать нецелесообразно. Построенная сеть имеет вид, изображенный на рис. 3; T = 75.

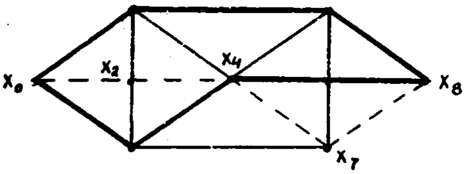


Рис. 4. Сеть G_3

Замечание. Для большинства практических приложений достаточно вычислять минимальное время T приближенно. Относительную погрешность σ, допускаемую при замене T величиной T_{i+1}, вычисленной на $i+1$ -м шаге, можно оценивать с помощью величины

$$\delta = \frac{T_i - T_{i+1}}{t_{i+1}},$$

поскольку

$$\delta = \frac{T_i - T_{i+1}}{t_{i+1}} \sim \frac{T_i - T}{T} = \sigma.$$

Последнее соотношение вытекает из определения постоянной $B(G)$ и неравенств

$$T_i > T_{i+1} \geqslant T \geqslant t_{i+1} > t_i .$$

Расчеты показали, что оценка относительной погрешности σ с по-

мощью величины δ дает достаточно хорошие результаты.

П р о г р а м м а

```
начало целый n,m,k ; ввод(n,m); начало целый i, j, l,ρ,s,n2,n3,ξ,ν,γ
вещ T,t1,Φ,α,β,P; целый массив d[1: n, 1:2];
вещ массив A[0:n, 0:n], C,C1,D,H1,K1,H2,K2[1:m];
процедура Σ(a,b,l); начало a:=0; для i:=1,...,l
цикл a:=a+b конец; ввод (C,D,H1, K1, P); t1:=0; k:=1;
для l:=0,...,n цикл для i:=0,...,n цикл { для
j:=1,...,m цикл если H1[j]=l∧K1[j]=i то {A[l,i]:=
D[j]; на Γ1} иначе A[l,i]:=₁₀18; Γ1:} ;
для i:=1,...,m цикл C1[i]:=0; Φ:=0; α:=0;
R: начало собств вещ массив n1,t,φ[0: k+1],нач,
кон [1: k+1, 1: m]; t[0]:=0; φ[0]:=0 ; начало целый λ,α,β;
целый массив p,l,t[0:n] ; вещ массив min[0:n], a[0:n,0:n];
процедура S(j) ; начало a[l[λ-1],j]:=a[l[λ-1],j]+min[λ-1]
конец; если k≥1 то для i:=0,...,n цикл для j:=0,..., n
цикл a[i,j]:=A[i,j]; λ:=0; t[0]:=0; p[0]:=0; l[0]:=0;
min[0]:=₁₀18; для j:=1,..., n цикл если min[λ] ≥ a[0,j]
        то {min[λ]:=a[0,j]; l[λ]:=j};
          если min [λ]≥₁₀ 18 то на ℒ2;
ℒ: если l[λ]<n то λ:=λ+1 иначе {n1[k]:=λ;
на ℒ1} ; если t[λ-1]≤l[λ-1] то t[λ]:=l[λ-1]
иначе { для i:=2,...,λ цикл если t[λ-i]≤l[λ-1]
то { для j:=0,...,i-2 цикл t[λ-j]:=t[λ-j-1];
 t[λ-i+1]:=l[λ-1]; на M}};
 M: для j:=0,...,t[1]-1 цикл S(j); для l:=2,...,λ
цикл для j:=t[i-1]+1,...,t[i]-1 цикл S(j); для
 j:=t[λ]+1,...,n цикл S(j); t[λ+1]:=n+1; min[λ]:=₁₀ 18;
для β:=0,..., λ цикл {i:=t[β] ; для α:=1,..., λ
   цикл { для j:=t[α-1]+1,...,t[α]-1 цикл если
 min[λ]≥a[i,j] то {min[λ]:=a[i,j]; l[λ]:=j;
   p[λ]:=i } ; для j:=t[α]+1,...,t[α+1]-1 цикл если
 min[λ]≥a[i,j] то {min [λ]:=a[i,j]; l[λ]:=j;
 p[λ]:=i}}}; на ℒ; ℒ1: α:=1; M2: d[α,1]:=p[λ]; d[α,2]:=l[λ];
если p[λ]=0 то на M3 ; для β:=1,...,n1[k]
цикл если p[λ]=l[λ-β] то {α:=α+1;
λ:=λ-β; на M2} ; M3: t[k]:=min[n1[k]];
n1[k]:=α конец; ν:=1; n1[k+1]:=n; R1 : начало
собств вещ массив μ[1:k+1, 1:n1[k+1]] ; перекл
q:=r1, r2 ; на q[ν]; r1: для j:=1,...,n1[k] цикл
{l:=n1[k]-j+1; для i:=1,...,m цикл { если H1[i]=d[l,1]∧
K1[i]=d[l,2]∨H1[i]=d[l,2]∧K1[i]=d[l,1] то
 { нач [k,i]:=d[l,1]; кон [k,i]:=d[l,2]; μ[k,j]:=i}}};
для β:=0,...,n цикл для i:=1,...,n цикл для j:=1,...,
 n1[k] цикл {l:=μ[k,j] ; если нач [k,l]< H1[l]=β∧
```

```
кон [k,l]=k1[l]=i то A[i,β]:=D[l] иначе если нач [k,l]
   =k1[l]=i ∧ кон [k,l]=н1[l]=β то A[β,i]:=D[l]}; φ[k]:=⏨18;
для i:=1,...,n1[k] цикл {j:=μ[k,l]; если нач [k,j]=
   н1[j] ∧ кон [k,j]=k1[j] то { если φ[k] ≥ c[j]
то φ[k]:=c[j] } иначе если φ[k] ≥ c1[j] то φ[k]:=c1[j]};
вывод (k, n1, μ) ; для i:=1,...,n1[k]
цикл {j:=μ[k,i] ; если нач [k,j] = н1[j] ∧
кон [k,j]=k1[j] то {c[j]:=c[j]−φ[k]; c1[j]:=c1[j]+φ[k];
если c[j]=0 то {l:= нач [k,j]; s:= кон [k,j]; A[l,s]:=⏨18
}} иначе {c[j]:=c[j]+φ[k] ; c1[j]:=c1[j]−φ[k];
если c1[j]=0 то {l:= кон [k,j]; s:= нач[k,j]; A[l,s]:=⏨18}}};
если φ[k]=0 то на ℒ2 ;                  для i:=1,...,n1[k]
            цикл {j:=μ[k,i]; если нач [k,j] = k1[j] ∧
   кон [k,j]=н1[j] то {l:=i; на ℒ3}} ;
   Σ(t[k], D[μ[k,l]], n1[k]);
на ℒ4; ℒ3: p:=1; ℒ7: для i:=1,...,n1[p] цикл
если μ[k,l]=μ[p,i] то {j:=μ[p,i]; если нач [p,i]=н1[j] ∧
кон [p,j]=k1[j] то {s:= i ; на ℒ5 }}; ℒ8: p:=p+1;
если p=k то на ℒ4 иначе на ℒ7;
  ℒ5: n2:=s+n1[k]−l−1; n3=l+n1[p]−s−1; γ:=1;
  Г2: начало вещ массив ξ[1:n2], ξ1[1:n3];
переключатель Q:=τ1, τ2, τ3, τ4; на Q[γ]; τ1:
для i:=1,...,s−1 цикл ξ[i]:=μ[p,i];
для i:=s,...,n2 цикл ξ[i]:=μ[k,l+i−s+1];
  н2[ ]:= нач [p, ]; k2[ ]:= кон [p, ];
для i:=1,...,l−1 цикл ξ1[i]:=μ[k,i]; для i:=l,...,n3
        цикл ξ1[i]:=μ[p,s+i−l+1]; если φ[p] = φ[k] то
на Г иначе если φ[p]>φ[k] то на ℒ6 иначе на ℒ9;
   Г: n1[p]:=n2; n1[k]:=n3; ν:=2; γ:=2; на R1;
 τ2: для i:=s,...,n2 цикл {j:=μ[k,l+i−s+1];
нач  [p,j]:=  нач [k,j]  ; кон [p,j]:= кон [k,j]};
для i:=l,...,n3 цикл {j:=μ[p,s+i−l+1];
нач [k,j]:=н2[j];  кон [k,j]:=k2[j]}; μ[p, ]:=ξ[ ];
  α:=α−φ[p]×t[p]; Σ(t[p],D[μ[p,i]],n1[p]); μ[k, ]:=
  ξ1[ ]; Σ(t[k], D[μ[k,i]],n1[k]); α:=α+φ[p]×t[p];
на Г3; ℒ6: φ[p]:=φ[p]−φ[k]; n1[k]:=n2; n1[k+1]:=n3;
ν:=2; γ:=3; на R1; τ3: для i:=1,...,l−1 цикл {j:=μ[k,i];
нач [k+1,j]:= нач [k,j] ;   кон [k+1,j]:=кон [k,j]};
для i:=l,...,n3 цикл{j:=μ[p,s+i−l+1]; нач[k+1,j]:= нач [p,j];
кон [k+1,j]:= кон [p,j]} ; для i:=1,...,s−1 цикл{j:=μ[p,i];
нач [k,j]:= нач [p,j] ;    кон[k,j]:= кон[p,j]}; Σ(t[k],
D[μ[k,i]],n1[k]); μ[k+1, ]:= ξ1[ ]; α:=α−φ[k]×t[p]+φ[k]×
 t[k]; Σ(t[k+1],D[μ[k+1,i]],n1[k+1]); φ[k+1]:= φ[k];
   k:=k+1 ;                                   на Г3 ;
ℒ9: n1[p]:=n2; n1[k+1]:=n1[k]; n1[k]:=n3 ; φ[k+1]:=
```

$\varphi[k]-\varphi[p]$; $\varphi[k]:=\varphi[p]$; $\nu:=2$; $\gamma:=4$; на $R1$;
$\tau 4$: нач $[k+1,]:=$ нач $[k,]$; кон $[k+1,]:=$ кон $[k,]$;
для $l:=s,\dots,n2$ цикл $\{j:=\mu[k,l+l-s+1]$;
нач $[p,j]:=$ нач $[k,j]$; кон $[p,j]:=$ кон $[k,j]\}$;
для $i:=l,\dots,n3$ цикл $\{j:=\mu[p,s+i-l+1]$;
нач $[k,j]:=H2[j]$; кон $[k,j]:=k2[j]\}$;
$\mu[p,]:=\xi[\,]$; $\mu[k+1,]:=\mu[k,]$; $\mu[k,]:=\xi 1[\,]$;
$\Phi:=\Phi+\varphi[p]$; $\alpha:=\alpha-\varphi[p]\times t[p]$; $\Sigma(t[p],D[\mu[p,i]],$
$n1[p])$; $t[k+1]:=t[k]$; $\Sigma(t[k],D[\mu[k,i]],n1[k])$;
$\alpha:=\alpha+\varphi[p]\times t[p]+\varphi[k]\times t[k]$; $k:=k+1$; $p:=p+1$;
на $\varkappa 7$ конец конец ;
$\Gamma 3$: для $i:=1,\dots,k$ цикл если $t1<t[i]$ то $t1:=t[i]$;
на $\varkappa 10$; $\varkappa 4$: если $t[k]=t[k-1]$ то на $\Gamma 4$ иначе
если $t[k]>t[k-1]$ то на $\varkappa 13$; $\varkappa 13$: $t1:=t[k]$; $\varkappa 10$: $B:=$
$\Phi\times(t1+1)-\alpha+1$; если $p>B$ то на $\varkappa 11$ иначе на $\varkappa 12$;
$\varkappa 11$: если $P=B$ то $\{T:=t1$; на $\Gamma 2\}$;
$\Gamma 4$: $\{\Phi:=\Phi+\varphi[k]$; $\alpha:=\alpha+\varphi[k]\times t[k]$; $k:=k+1$; на $R\}$;
$\varkappa 12$: если $k>2$ то на $\varkappa 2$ иначе $\{T:=t[1]$;
на $\Gamma 2\}$; $\varkappa 2$: $T:=\mathrm{entier}((P-1+\alpha)/\Phi)$;
$\Gamma 2$: вывод (t,B,T,φ) конец конец конец ж

Поступила в редакцию 10.I.1969 г.

Литература

1. В.А.Евстигнеев. Транспортная задача по времени I.-Дискретный анализ, Новосибирск, 1968, вып. 13.

2. T.C.Hu, Minimum-cost flows in convex-cost networks, "Naval Research Logist Quart". vol.13, N 1, 1966, 1-10.

3. В.А.Евстигнеев, В.П.Ясакова. Об одном алгоритме на графах. Сб. "Управляемые системы" /см. настоящий сборник/.