ON STOCHASTIC ANALYSIS OF PROJECT-NETWORKS

W. Gaul

Universität Karlsruhe

ABSTRACT

If the activity-completion-times of a project-network are random variables the project-completion-time is a random variable the distribution function of which is difficult to obtain. Thus, efforts have been made to determine bounds for the mean and bounding distribution functions for the distribution function of the project-completion-time some results of which are shortly surveyed. Then, a new approach using stochastic programming for a cost-oriented project scheduling model is presented. Generalizing a well-known Fulkerson-approach planned execution-times for the random activity-completion-times are computed where nonconformity with the actual realizations impose compensation costs (gains). Taking into consideration a prescribed project-completion-time constraint the expected costs for performing the activities according to the planned execution-times are minimized. A solution procedure is described which constructs a sequence of nonstochastic Fulkerson project scheduling models. It is demonstrated by means of an example.

KEYWORDS: Network Programming, Scheduling Theory, Stochastic Programming.

1. INTRODUCTION

A project is described by a set of activities, a relation on this set representing restrictions between the activities and activity-completion-times.

A project-network, as graphtheoretical description of a project,

M. A. H. Dempster et al. (eds.), Deterministic and Stochastic Scheduling, 297–309.

is given by

$$D_{s,t} = (V,X,f,Y_X)$$

where (V,X,f) is a finite, directed, simple, acyclic, weakly connected graph with point set V, arc set X, incidence mapping $f=(f^1,f^2)$ with $f^i:X\rightarrow V$, i=1,2 ($f^1(x),f^2(x)$ denote the starting-, end point of $x\in X$) and single-element point basis s, single-element point contrabasis t, see e.g. Harary, Norman and Cartwright [12] for the graphtheoretical notations. X corresponds with the set of activities of the project at least after introducing dummy activities (The case where V corresponds with the activities is handled e.g. by MPM, Metra Potential Method, but not discussed here.). The restrictions between the activities are described by the chosen project-network's arc-adjacency relation.
$Y_X=(Y_x,x\in X)$ is a random vector defined on a probability space $(\Omega,\mathfrak{S},Pr)$ the components of which give the activity-completion-times (The more general case where additional stochastic aspects influence the project structure is handled e.g. by GERT, Graphical Evaluation and Review Technique, but not discussed here, see Neumann and Steinhardt [15] for a recent contribution.).
Such project-networks $D_{s,t}$ have proved to be an appropriate tool when a schedule for coordinating and supervising of the single activities of a project is needed. One of the aims of project scheduling is to determine the project-completion-time which is yielded by maximizing over the sums of the completion-times of those activities which form paths from s to t. Even under the assumption of stochastic independence for $Y_x,x\in X$, however, the distribution function of the project-completion-time is difficult to obtain (activities can be used by different paths). Thus, in Van Slyke [20] one of the first attempts to apply Monte-Carlo methods was described. Efforts which have been made to determine bounds for the mean and bounding distribution functions for the distribution function of the project-completion-time are shortly surveyed in section 2. Together with the well-known CPM, Critical Path Method, and PERT, Program Evaluation and Review Technique, approaches the results of Fulkerson [8], Clingen [4], Robillard and Trahan [16] and Devroye [5] concerning bounds for the mean of the project-completion-time are mentioned some of which are shown to be special cases of a more general result of Gaul [10]. The results of Kleindorfer [14] and Shogan [18] on bounding distribution functions for the distribution function of the project-completion-time close section 2. As in a typical planning situation execution-times for the activities have to be planned before the actual realizations of the random activity-completion-times are known, in section 3 a new approach of Cleef and Gaul [3] is presented using stochastic programming for a cost-oriented project scheduling model. Generalizing a well-known approach of Fulkerson [7], see also Ford and Fulkerson [6], planned execution-times for the random activity-completion-times are computed where nonconformity with the actual realizations impose compensation costs (gains). The planned execution-times are determined in such a way that the expected compensation costs together with a nonstochastic cost-term are minimized.

Using discrete random activity-completion times (e.g. as approximation of the actual ones) a solution procedure is described which constructs a finite sequence of nonstochastic Fulkerson project scheduling models. The size of the subproblems in the sequence is independent of the number of realizations of the activity-completion times. In section 4 the new approach is demonstrated by means of an example.

2. BOUNDS, BOUNDING DISTRIBUTION FUNCTIONS

Let m be the number of points of $D_{s,t}$.
For $D_{s,t}$ there exists a bijective labelling $l: V \to \{1,\dots,m\}$ with

$$l(s)=1,\ l(t)=m \text{ and } x \in X \Rightarrow l(f^1(x)) < l(f^2(x)).$$

In this section such a labelling is needed for the sequential determination of bounds and bounding distribution functions.
For graphtheoretical considerations sometimes the notation $(V(D), X(D))$ (omitting the incidence mapping and the random vector) is used for a network D with point set $V(D)$, arc set $X(D)$. With these abbreviations

$$D^1 \subset D^2 \text{ iff } V(D^1) \subset V(D^2),\ X(D^1) \subset X(D^2),$$

$$D^1 \overset{\cup}{\underset{\cap}{}} D^2 \text{ iff } V(D^1 \overset{\cup}{\underset{\cap}{}} D^2) = V(D^1) \overset{\cup}{\underset{\cap}{}} V(D^2),$$

$$X(D^1 \overset{\cup}{\underset{\cap}{}} D^2) = X(D^1) \overset{\cup}{\underset{\cap}{}} X(D^2)$$

describe the subnetwork, union-intersection-network notation. Now,

$$D_{i,j} \subset D_{1,m} \text{ is called subproject-network}$$

if $D_{i,j}$ is a project-network with point basis i and point contrabasis j. One has $V(D_{i,j}) \subset \{i,i+1,\dots,j-1,j\}$, $X(D_{i,j}) \subset f^{-1}(V(D_{i,j}) \times V(D_{i,j}) \cap f(X))$, the incidence mapping $f/_{X(D_{i,j})}$ and the random vector $Y_{X(D_{i,j})} = (Y_x, x \in X(D_{i,j}))$ are mostly omitted. If for $i,j \in V$ a subproject-network $D_{i,j}$ exists $\tilde{D}_{i,j}$ denotes the maximal one (notice $\tilde{D}_{1,m} = D_{1,m}$ the underlying project-network). A path $P_{i,j}$ with $V(P_{i,j}) = \{i_1,\dots,i_n \mid i_1=i,\ i_n=j\}$, $X(P_{i,j}) = \{x_1,\dots,x_{n-1} \mid f(x_\mu) = (i_\mu, i_{\mu+1}), \mu=1,\dots,n-1\}$ is a special subproject-network. $(P_{i,j})_k$ resp. $(P_{i,j})^k$, $k \in V(P_{i,j})$, gives the subpath of $P_{i,j}$ from i to k resp. k to j. Instead of $D_{f^1(x),f^2(x)}$ the arc notation x is used.

$$L(D_{i,j}) = \max_{P_{i,j} \subset D_{i,j}} \sum_{x \in X(P_{i,j})} Y_x \tag{1}$$

is the $D_{i,j}$-completion-time ($L(\tilde{D}_{1,m})$ is the project-completion-time). Next, for $v \in V$, $v>1$, consider subproject-network systems of the form

$$\delta_v = \{D_{i,v} \mid i<v\}.$$

With $B(\delta_v) = \{i \mid i \in V, D_{i,v} \in \delta_v\}$ call δ_v proper (with respect to $\tilde{D}_{1,m}$) if

$$\forall D^1_{i_1,v}, D^2_{i_2,v} \in \delta_v : D^1_{i_1,v} \cap D^2_{i_2,v} = \begin{cases} (\{i,v\},\emptyset) & \text{if } i_1=i_2=i, \\ (\{v\},\emptyset) & \text{otherwise,} \end{cases} \tag{2}$$

$$\forall P_{1,v} \subset \tilde{D}_{1,m}\ \exists\ D_{i,v} \in \delta_v : P_{1,v} = (P_{1,v})_i \cup (P_{1,v})^i \tag{3}$$

$$\text{with } (P_{1,v})_i \cap \delta_v \subset (B(\delta_v),\emptyset),\ (P_{1,v})^i \subset D_{i,v}\ .$$

Proper δ_v always exist, e.g. $\delta_v=\{\tilde{D}_{1,v}\}$ is proper. A useful property of proper δ_v is, see Gaul [10],

(4) $L(\tilde{D}_{1,v})=\max_{D_{i,v}\in\delta_v}\{L(\tilde{D}_{1,i})+L(D_{i,v})\}$ if δ_v is proper.

To define lower bounds for the $L(\tilde{D}_{1,v})$-mean let, for $X^*\subset X$, E_{X^*} denote the integration with respect to $Y^*_X=(Y_x,x\in X^*)$, E (without subscript) the expectation. Assume, for proper $\delta_v=\{D_{i,v}\}$, that $T_i,i\in B(\delta_v)$, are known lower bounds for $E\,L(\tilde{D}_{1,i})$, and that $X_v,\bar{X}_v$ is a partition of X with $X_v\subset X(\delta_v)$, then, under adequate stochastic independence assumptions

(5) $$E\,L(\tilde{D}_{1,v})\geq E_{X_v}\max_{D_{i,v}\in\delta_v}\{T_i+E_{\bar{X}_v}L(D_{i,v})\}=T(\delta_v,X_v,L(\tilde{D}_{1,v})).$$

For different choices of δ_v and X_v one gets well-known special cases:

$\delta^1_v=\{D_{iv}|D_{iv}$ coincides with x with $f(x)=(i,v)\},X^1_v=\emptyset$

yields

(6) $$T^1_v=T(\delta^1_v,\emptyset,L(\tilde{D}_{1,v}))=\max_{x\in\delta^1_v}\{T^1_i+E_{\bar{X}^1_v}Y_x\}=\max_{x\in\delta^1_v}\{T_i+EY_x\}.$$

Using recursive arguments, if $T^1_i,i\in B(\delta^1_v)$, are determined in the same way as described by (6) (with $T^1_1=0$), T^1_v gives the PERT lower bound of $E\,L(\tilde{D}_{1,v})$. If all $Y_x,x\in X$, have degenerate distributions, (6) describes the CPM-approach.

$\delta^2_v=\delta^1_v,\ X^2_v=X(\delta^2_v)$

yields

(7) $$T^2_v=T(\delta^2_v,X^2_v,L(\tilde{D}_{1,v}))=E_{X^2_v}\max_{x\in\delta^2_v}\{T^2_i+E_{\bar{X}^2_v}Y_x\}=E\max_{x\in\delta^2_v}\{T^2_i+Y_x\}.$$

Using recursive arguments, if $T^2_i,i\in B(\delta^2_v)$, are determined in the same way as described by (7) (with $T^2_1=0$), T^2_v gives the Fulkerson [8] lower bound, see also Clingen [4], of $E\,L(\tilde{D}_{1,v})$.

Whereas it is easy to see that $\delta^1_v=\delta^2_v$ is proper, now, among the set of paths P_{iv} one has to choose

$\delta^3_v=\{P_{iv}|P_{iv}$ is path from i to $v,i<v\}$ proper, $X^3_v=X(\delta^3_v)$

which yields

(8) $$T^3_v=T(\delta^3_v,X^3_v,L(\tilde{D}_{1,v}))=E\max_{P_{iv}\in\delta^3_v}\{T^3_i+L(P_{iv})\}.$$

Again, using recursive arguments one gets a method suggest by Robillard and Trahan [16]. For the exact computation of $E\,L(\tilde{D}_{1,v})$ choose

$\delta^4_v=\{\tilde{D}_{1,v}\},X^4_v=X(\tilde{D}_{1,v})$

which yields

(9) $$T^4_v=T(\delta^4_v,X^4_v,L(\tilde{D}_{1,v}))=E[T^4_1+L(\tilde{D}_{1,v})]=E\,L(\tilde{D}_{1,v})$$

with $T^4_1=0$ as usual.

Under assumptions given in Gaul [10] one can show

$E\,L(\tilde{D}_{1,v})=T^4_v\geq T^3_v\geq T^2_v\geq T^1_v$

and construct improved lower bounds.

An easy method to determine upper bounds is given in Devroye [5]. Knowing EY_x, var Y_x, and using the recursive approach described in (6),

$$U_v' = \max_{x\in\delta_v^1}\{U_i'+EY_x\}+\sqrt{n_v \max_{x\in\delta_v^1}\{\overline{\operatorname{var} L(\tilde{D}_{1,i})}+\operatorname{var}Y_x\}},$$

$$(10)\qquad U_v'' = \max_{x\in\delta_v^1}\{U_i''+EY_x\}+ \dots$$

$$\dots+\sqrt{(n_v-1)\,[\max_{x\in\delta_v^1}\{2\,\overline{\operatorname{var} L(\tilde{D}_{1,i})}+\operatorname{var}Y_x\}+\min_{x\in\delta_v^1}\{2\,\overline{\operatorname{var} L(\tilde{D}_{1,i})}+\operatorname{var}Y_x\}]}$$

are shown to be upper bounds for $E\,L(\tilde{D}_{1,v})$ if $Y_x, x\in X$, are stochastically independent. Here, n_v is the number of elements of $B(\delta_v^1)$ and $\overline{\operatorname{var} L(\tilde{D}_{1,v})}$ is an upper bound for $\operatorname{var} L(\tilde{D}_{1,v})$ recursively defined by

$$\overline{\operatorname{var} L(\tilde{D}_{1,v})} = \sum_{x\in\delta_v^1}[\overline{\operatorname{var} L(\tilde{D}_{1,i})}+\operatorname{var} Y_x] \quad (\text{with } \overline{\operatorname{var} L(\tilde{D}_{1,1})}=0).$$

Lower and upper bounds for the mean and for higher moments of the project-completion-time can also be determined if one knows bounding distribution functions for the distribution function of the project-completion-time, see Kleindorfer [14] and Shogan [18]. With restriction to discrete random activity-completion-times and the abbreviations

$$p(y(v))=\Pr(Y_{X(\delta_v^1)}=y(v)),\quad y(v)=(y_x, x\in X(\delta_v^1)),$$

the following recursive definition of bounding distribution functions is possible:

Under the assumption of stochastic independence for $Y_{X(\delta_v^1)}=\{Y_x, x\in X(\delta_v^1)\}$, $v\in\{2,\dots,m\}$,

$$(11)\qquad \begin{aligned} &(a)\ F^u_{\tilde{D}_{1,v}}(t) = \sum_{y(v)} p(y(v))\,[\min_{x\in\delta_v^1} F^u_{\tilde{D}_{1,i}}(t-y_x)],\\ &(b)\ F^l_{\tilde{D}_{1,v}}(t) = \sum_{y(v)} p(y(v))\max\{0,[\sum_{x\in\delta_v^1} F^l_{\tilde{D}_{1,i}}(t-y_x)]-n_v+1\}\\ &\quad \text{with } F^l_{\tilde{D}_{1,1}}(t)=F^u_{\tilde{D}_{1,1}}(t)=\begin{cases}1, & t\geq 0,\\ 0, & \text{otherwise}\end{cases}\end{aligned}$$

fulfill

$$F^l_{\tilde{D}_{1,v}}(t)\leq F_{\tilde{D}_{1,v}}(t) = \Pr(L(\tilde{D}_{1,v})\leq t)\leq F^u_{\tilde{D}_{1,v}}(t),\qquad t\in\mathbb{R}.$$

Obviously, (11) is based on the well-known Frechet-bounds for $\Pr(\bigcap_{x\in\delta_v^1}\{L(\tilde{D}_{1,i})\leq t-y_x\})$.

Under the additional stochastic dependence assumption of association for $Y_x, x\in X(\delta_v^1)$, $v\in\{2,\dots,m\}$ (often used in context with reliability problems), an improved lower bounding distribution function

$$(12)\qquad \begin{aligned} &F^{l(ass)}_{\tilde{D}_{1,v}}(t) = \sum_{y(v)} p(y(v))\left[\prod_{x\in\delta_v^1} F^{l(ass)}_{\tilde{D}_{1,i}}(t-y_x)\right]\\ &\text{with } F^{l(ass)}_{\tilde{D}_{1,1}}(t)=\begin{cases}1, & t\geq 0,\\ 0, & \text{otherwise}\end{cases}\end{aligned}$$

can be computed which fulfills

$$F^l_{\tilde{D}_{i,v}}(t)\leq F^{l(ass)}_{\tilde{D}_{1,v}}(t)\leq F_{\tilde{D}_{1,v}}(t),\qquad t\in\mathbb{R}.$$

Of course, having established lower, upper bounding distribution functions the determination of lower, upper bounds for the mean and the variance of $L(\tilde{D}_{1,v})$ is straightforward and, thus, omitted. The Kleindorfer bounding distribution functions are also not explicitly reported because, although they were developped under the stronger assumption of stochastic independence for $Y_x, x \in X$, the Shogan bounding distribution functions (11a), (12) are tighter. In all cases, using recursive arguments and increasing v up to m, the desired results for the project-completion-time are obtained.

3. STOCHASTIC PROGRAMMING PROJECT SCHEDULING

Knowing the difficulties originating from the stochastic description of project scheduling problems as discussed in section 2 the question arises whether a new approach might be more appropriate. As in a typical planning situation one has to plan execution-times for the single activities under cost-viewpoints before the actual realizations of the random activity-completion-times are known a "two-stage stochastic programming with simple recourse" approach was described in Cleef and Gaul [3] which generalizes the nonstochastic Fulkerson [7] project scheduling model. A first attempt to apply stochastic programming to project scheduling was formulated by Charnes, Cooper and Thompson [1] within a "chance-constrained stochastic programming" approach.
For an introduction to stochastic programming Kall [13], for an extensive bibliography on papers dealing with various topics of stochastic programming Stancu-Minasian and Wets [19] are recommended, for considerations where the here described stochastic programming model is used for the general linear case with simple recourse see Cleef [2].
The new stochastic programming project scheduling approach is formulated as follows:
For the arcs of the given project-network $D_{s,t}=(V,X,f,Y_X)$ assume

(13) $\qquad Pr(Y_x \geq y_x^o) = 1, \quad x \in X,$

where $y_x^o \geq 0$ is the lowest possible (crash) completion-time.
In the nonstochastic case, see Fulkerson [7], Y_x have degenerate distributions at $y_x \geq y_x^o$ but if one puts up with additional costs for extra efforts assumed to be describable by linear cost-functions of the form

(14) $\qquad c(d_x)=b_x-o_x d_x \;, \quad d_x \in [y_x^o, y_x], \quad x \in X,$

where $b_x, o_x \geq 0$ are known values allowing for costs of needed resources (machines, material, staff etc.),

planned execution-times d_x can be determined which minimize the total costs $\sum_{x \in X} c(d_x)$ under a project-completion-time constraint $\lambda > 0$.
In the stochastic case assume that X_d, X_r is a partition of X into the sets of arcs with deterministic or random activity-completion-times. X_d contains the dummy activities (with $y_x^o=y_x=0$, $b_x=0$), $X_r=\emptyset$ gives the nonstochastic Fulkerson-approach. For $x \in X_r$

additional costs for compensating nonconformity between the actual realizations of the activity-completion-times $Y_x(\omega)$, $\omega\in\Omega$, and the planned execution-times d_x (which have to be determined before the realizations are known)

$$(15)\qquad \varphi_{d_x}(Y_x(\omega))=\begin{cases} q_x^+(Y_x(\omega)-d_x) & > \\ 0 & ,\ Y_x(\omega) = d_x\ ,\quad \omega\in\Omega,\ x\in X_r, \\ -q_x^-(Y_x(\omega)-d_x) & < \end{cases}$$

have to be taken into consideration where q_x^+, q_x^- are known compensation cost-terms satisfying

$$(16)\qquad -q_x^+ < q_x^- \leq o_x \quad , \quad x\in X_r.$$

With

$$\Phi_{d_x}=\begin{cases} E\varphi_{d_x}+c(d_x) & , \quad x\in X_r, \\ c(d_x) & , \quad x\in X_d, \end{cases}$$

the following SPPS, Stochastic Programming Project Scheduling, approach can be formulated:

$$\sum_{x\in X} \Phi_{d_x} = \min$$

$$(17)\qquad \begin{aligned} d_x+\pi_{f^1(x)}-\pi_{f^2(x)} &\leq 0 \quad , \quad x\in X, \\ -\pi_s + \pi_t &\leq \lambda \\ y_x^o \leq d_x \leq y_x & \quad , \quad x\in X. \end{aligned}$$

Here, π_i, $i\in V$, are upper time-bounds when with respect to the planned execution-times d_x all activities x with $f^2(x)=i$ have to be completed. For an appropriate choice of y_x for $x\in X_r$ see (18). (17) describes a linear program if one assumes that for $x\in X_r$, Y_x are (or are approximated by) discrete random activity-completion-times. With the realizations and probabilities

$$y_x^k \text{ with } Pr(Y_x=y_x^k)=p_x(k)>0,\ k=1,\dots,r_x,\ \sum_{k=1}^{r_x} p_x(k)=1$$

and, for computational convenience, with the choice of y_x^o, $y_x^{r_x+1}$ with $p_x(0)=p_x(r_x+1)=0$ according to

$$(18)\qquad 0\leq y_x^o<y_x^1<y_x^2<\dots<y_x^{r_x}<y_x\ (=y_x^{r_x+1}=\max\{\max\{y_x^{r_x}\},\lambda\}+1),$$

the reformulation of (17) gives the following large (dependent on the number of realizations of the random variables) linear program:

$$\sum_{x\in X_r}\ \sum_{k=1}^{r_x}\Big[p_x(k)\,[q_x^+u_x^+(k)+q_x^-u_x^-(k)\,]+c(d_x)\Big]+\sum_{x\in X_d} c(d_x)=\min,$$

$$(19)\qquad \begin{aligned} d_x+\pi_{f^1(x)}-\pi_{f^2(x)} &\leq 0 \quad , \quad x\in X\ , \\ -\pi_s + \pi_t &\leq \lambda \\ d_x+u_x^+(k)-u_x^-(k) &= y_x^k \quad , \quad x\in X_r,\quad k=1,\dots,r_x, \\ -d_x &\leq -y_x^o,\ \ x\in X\ , \\ d_x &\leq y_x \quad , \quad x\in X\ , \\ u_x^+(k),u_x^-(k) &\geq 0 \quad , \quad x\in X\ ,\quad k=1,\dots,r_x. \end{aligned}$$

Instead of handling (19) a finite sequence of Fulkerson project scheduling models (independent of the number of realizations of the random variables) is solved. For the n-th subproblem select

$$(20)\qquad s_x^n\in\{0,1,\dots,r_x\},\ x\in X_r\ (\text{with } s_x^n=0,\ x\in X_d),$$

denote

$$(21)\qquad \alpha_x=\begin{cases} y_x^{s_x^n} \\ y_x^o \end{cases},\quad \beta_x=\begin{cases} y_x^{s_x^n+1} \\ Y_x \end{cases},\quad \gamma_x=\begin{cases} O(s_x^n) & ,\ x\in X_r, \\ o_x & ,\ x\in X_d, \end{cases}$$

with $O(s_x^n)=q_x^+-(q_x^++q_x^-)\Pr(Y_x\leq s_x^n)+o_x$,
and consider

$$\text{SUB}\ (s_x^n)\qquad \begin{aligned} &\sum_{x\in X} \gamma_x d_x = \max, \\ &d_x+\pi_{f^1(x)}-\pi_{f^2(x)} \leq 0, \\ &-\pi_s \quad +\pi_t \leq \lambda, \\ &-d_x \leq -\alpha_x, \\ &d_x \leq \beta_x, \end{aligned}$$

and its dual

$$\text{DSUB}\ (s_x^n)\qquad \begin{aligned} &\lambda v+\sum_{x\in X} [\beta_x g_x-\alpha_x h_x] = \min, \\ &w_x +g_x -h_x = \gamma_x \qquad ,\ x\in X, \\ &\sum_{\{x|f^1(x)=i\}} w_x-\sum_{\{x|f^2(x)=i\}} w_x=\begin{cases} v & ,\ i=s \\ 0 & ,\ i\neq s,t \\ -v & ,\ i=t \end{cases},\ i\in V, \\ &w_x,\ g_x,\ h_x \geq 0 \qquad ,\ x\in X, \\ &v \geq 0. \end{aligned}$$

DSUB(s_x^n) has restrictions which remind of flow problems in networks, SUB(s_x^n) has restrictions which coincide with those of SPPS except for the random activity-completion-times where the variation of d_x-values is bounded by subsequent realizations $y_x^{s_n}$, $y_x^{s_n+1}$. Obviously, optimal solutions of SUB(s_x^n) are feasible for SPPS, thus, the question arises under which conditions optimal solutions of SUB(s_x^n) are also optimal for SPPS.

A sufficient optimality condition is the following:

Let d_x^*, $x\in X$, π_i^*, $i\in V$, be optimal for SUB(s_x^n) and w_x^*, g_x^*, h_x^*, $x\in X$, v^* be optimal for DSUB(s_x^n).

If

$$(22)\qquad \begin{aligned} &g_x^* \leq (q_x^++q_x^-)p_x(s_x^n+1) \qquad ,\ x\in X_r, \\ &h_x^* \leq (q_x^++q_x^-)p_x(s_x^n) \qquad ,\ x\in X_r \text{ with } s_x^n>0, \end{aligned}$$

then d_x^*, $x\in X$, π_i^*, $i\in V$, is optimal for SPPS.

If (22) fails to be satisfied new problems SUB(s_x^{n+1}), DSUB(s_x^{n+1}) have to be selected which allow improvements. The selection instructions use properties of the out-of-kilter algorithm applied to the following modified graph

$(\tilde{V},\tilde{X},\tilde{f})$ with

$\tilde{V}=V$,

$\tilde{X}=\{x_1|x\in X\}\cup\{x_2|x\in X\}\cup\{x_o\}$,

$$\tilde{f}(z)=\begin{cases} f(x) & ,\ z=x_k\ ,\ x\in X,\ k=1,2 \\ (t,s) & ,\ z=x_o \end{cases},\quad z\in\tilde{X},$$

because with

$$c_z=\begin{cases}-\beta_x & , \quad z=x_1 \quad , \quad x\in X \\ \lambda & , \quad z=x_o \\ -\alpha_x & , \quad z=x_2 \quad , \quad x\in X\end{cases} \quad , z\in\tilde{X}$$

and

$$l_z=\begin{cases}\gamma_x & , \quad z=x_1 \quad , \quad x\in X \\ \infty & , \quad \text{otherwise}\end{cases} \quad , z\in\tilde{X}$$

the following circulation problem in $(\tilde{V},\tilde{X},\tilde{f})$:

$$\sum_{z\in\tilde{X}} c_z w_z = \min,$$

CIRC(s^n_x)

$$\sum_{\{z|\tilde{f}^1(z)=i\}} w_z - \sum_{\{z|\tilde{f}^2(z)=i\}} w_z = 0 \quad , \quad i\in\tilde{V},$$

$$0\leq w_z\leq l_z \quad , \quad z\in\tilde{X},$$

is equivalent to DSUB(s^n_x).

A well-known solution procedure for CIRC(s^n_x) is the out-of-kilter algorithm which consists of an initial phase, a labeling phase, a circulation-alteration phase and a point value-alteration phase. It is easy to check that having obtained optimal point values $\tau^*_i, i\in\tilde{V}$, and an optimal circulation $w^*_z, z\in\tilde{X}$ (by application of the out-of-kilter algorithm to CIRC(s^n_x))

(23) $\quad w^*_x=w^*_{x_1}+w^*_{x_2},\ g^*_x=\gamma_x-w^*_{x_1},\ h^*_x=w^*_{x_2},\ x\in X,\ v^*=w^*_{x_o},$

is optimal for DSUB(s^n_x),

(24) $\quad \pi^*_i=-\tau^*_i,\ i\in V,\ d^*_x=\min\{\beta_x, \pi^*_{f^2(x)}-\pi^*_{f^1(x)}\},\ x\in X,$

is optimal for SUB(s^n_x).

Using CIRC(s^n_x), (23), (24) the optimality-condition (22) can be reformulated as

$$w^*_{x_1}\geq\gamma_x-(q^+_x+q^-_x)p_x(s^n_x+1) \quad , \quad x\in X_r,$$

(25)

$$w^*_{x_2} \quad (q^+_x+q^-_x)p_x(s^n_x) \quad , \quad x\in X_r \text{ with } s^n_x>0.$$

If (25) fails to be satisfied,

$$X^+_r=\{x\in X_r \mid 0\leq w^*_{x_1}<\gamma_x-(q^+_x+q^-_x)p_x(s^n_x+1)\},$$

$$X^-_r=\{x\in X_r \mid w^*_{x_2}>(q^+_x+q^-_x)p_x(s^n_x),\ s^n_x>0\},$$

$$X^o_r= X_r\setminus\{X^+_r\cup X^-_r\}$$

is a partition of X_r from which one gets

$$s^{n+1}_x=\begin{cases}s^n_x+1 & , \quad x\in X^+_r, \\ s^n_x-1 & , \quad x\in X^-_r \text{ (with } s^{n+1}_x\equiv 0,\ x\in X_d), \\ s^n_x & , \quad x\in X^o_r,\end{cases}$$

and new problems SUB(s^{n+1}_x), DSUB(s^{n+1}_x), CIRC(s^{n+1}_x) for which improvements are possible if a "modified" out-of-kilter algorithm is used. It can be shown, see Cleef and Gaul [3]:

Let $(d^*_x)^{n+1}$, $x\in X$, $(\pi^*_i)^{n+1}$, $i\in V$, (resp. $(d^*_x)^n$, $x\in X$, $(\pi^*_i)^n$, $i\in V$) be optimal for SUB (s^{n+1}_x) (resp. SUB (s^n_x)) obtained by subsequent applications of the "modified" out-of-kilter algorithm.

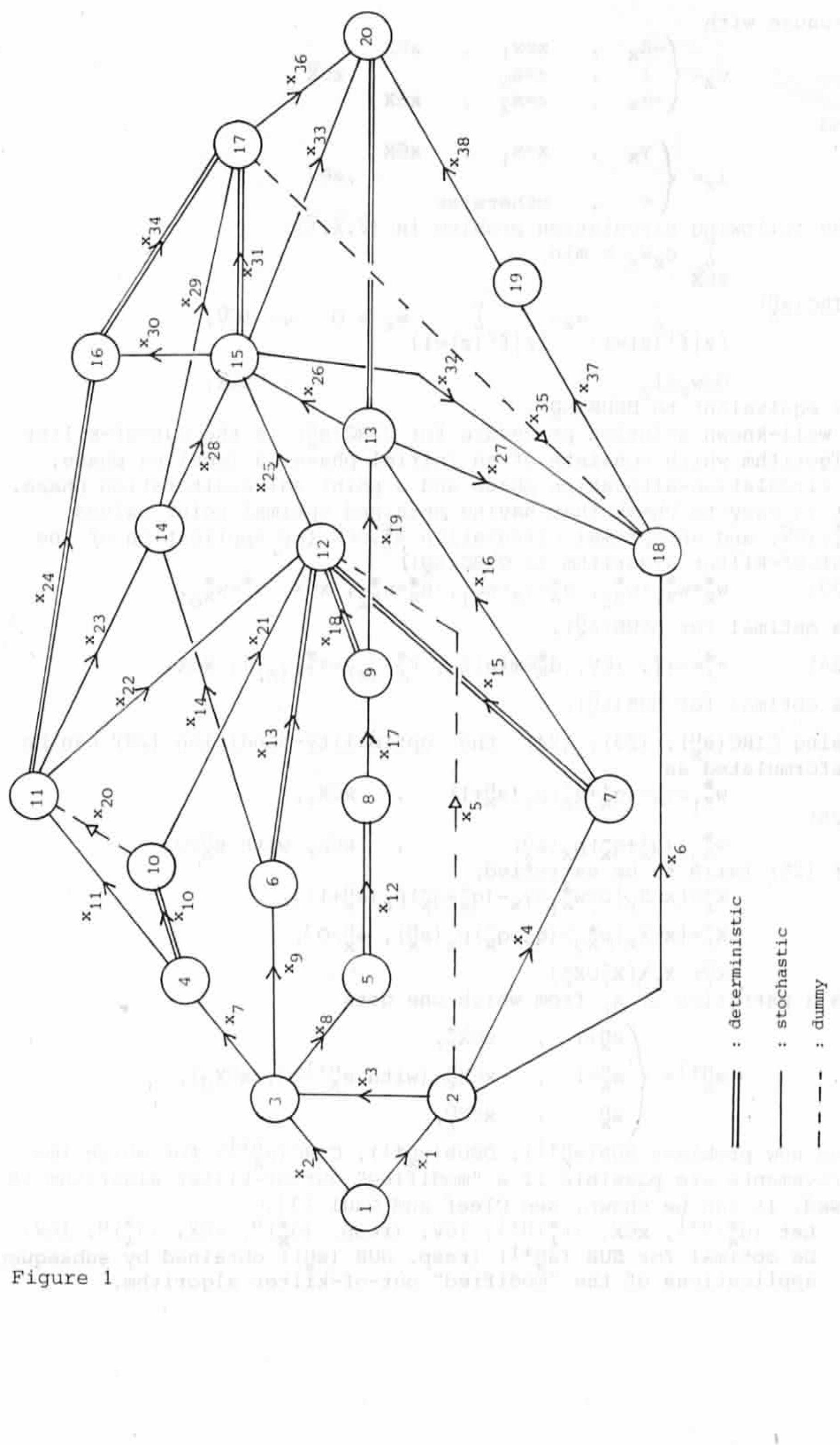

Figure 1

activity x_i	realizations $y^k_{x_i}$	probabilities $p^k_{x_i}$	$y^0_{x_i}$	b_{x_i}	a_{x_i}	$q^+_{x_i}$	$q^-_{x_i}$	$g^n_{x_i}$ n=0,...,7	$d^*_{x_i}$
x_1	3, 5, 10, 13, 20	0.200, 0.300, 0.300, 0.150, 0.050	1	500	30	5	- 3	0 0 0 0 0 0 0 0	1
x_2	3, 5, 10, 13, 20	0.200, 0.300, 0.300, 0.150, 0.050	1	1200	25	4	- 4	0 1 2 2 2 2 2 3	11
x_3	4, 6, 8, 10, 12	0.150, 0.250, 0.250, 0.200, 0.150	2	400	23	10	- 4	0 1 1 1 2 2 2 3	10
x_4	2, 3, 5, 6, 7	0.100, 0.200, 0.500, 0.100, 0.100	1	800	15	10	1	0 1 2 2 2 2 2 2	3
x_5	0, dummy activity	-	0	0	0	-	-	0 ——— 0	0
x_6	2, 4, 8, 16, 22	0.100, 0.200, 0.250, 0.250, 0.200	1	1500	5	11	5	0 1 2 3 4 4 4 4	22
x_7	6, 9, 15, 20, 25	0.175, 0.550, 0.200, 0.025, 0.050	3	3000	20	4	20	0 1 1 1 1 1 0 0	3
x_8	6, 7, 8, 12, 18	0.150, 0.075, 0.300, 0.300, 0.175	2	1550	10	10	0	0 1 2 2 2 1 0 0	4
x_9	6, 9, 15, 20, 25	0.175, 0.550, 0.200, 0.025, 0.050	3	1100	4	6	2	0 1 2 1 1 1 1 0	4
x_{10}	12, deterministic	-	6	1000	8	-	-	0 ——— 0	6
x_{11}	2, 3, 5, 6, 7	0.100, 0.200, 0.500, 0.100, 0.100	1	1100	1	6	0	0 1 2 3 4 4 3 3	6
x_{12}	13, deterministic	-	5	1800	12	-	-	0 ——— 0	5
x_{13}	12, deterministic	-	3	1800	14	-	-	0 ——— 0	12
x_{14}	3, 5, 10, 13, 20	0.200, 0.300, 0.300, 0.150, 0.050	1	600	3	14	2	0 1 2 2 2 2 2 2	10
x_{15}	24, deterministic	-	6	5000	18	-	-	0 ——— 0	23
x_{16}	6, 9, 15, 20, 25	0.175, 0.550, 0.200, 0.025, 0.050	3	300	1	5	- 3	0 1 2 3 3 3 3 3	20
x_{17}	3, 5, 10, 13, 20	0.200, 0.300, 0.300, 0.150, 0.050	1	2000	9	3	1	0 0 0 0 0 0 0 0	1
x_{18}	18, deterministic	-	6	2000	10	-	-	0 ——— 0	6
x_{19}	4, 6, 8, 10, 12	0.150, 0.250, 0.250, 0.200, [illegible]	2	[illegible]	4	2	0	0 1 1 0 0 0 0 0	1
x_{20}	0, dummy activity	-	0	0	0	-	-	0 ——— 0	0
x_{21}	2, 3, 5, 6, 7	0.100, 0.200, 0.500, 0.100, 0.100	1	700	9	12	2	0 1 2 3 4 5 5 5	9
x_{22}	6, 7, 8, 12, 18	0.175, 0.550, 0.200, 0.025, 0.050	2	2000	11	10	8	0 1 2 2 2 2 2 2	7
x_{23}	6, 9, 15, 20, 25	0.175, 0.550, 0.200, 0.025, 0.050	3	800	3	6	1	0 0 0 0 0 0 0 0	5
x_{24}	9, deterministic	-	3	1000	4	-	-	0 ——— 0	9
x_{25}	2, 3, 5, 6, 7	0.100, 0.200, 0.500, 0.100, 0.100	1	400	5	4	2	0 0 0 0 0 0 0 0	1
x_{26}	4, 6, 8, 10, 12	0.150, 0.250, 0.250, 0.200, 0.150	2	200	1	2	0	0 1 0 1 1 1 1 1	4
x_{27}	6, 7, 14, 15, 30	0.300, 0.100, 0.250, 0.250, 0.100	2	500	2	10	2	0 1 2 2 2 2 2 2	12
x_{28}	6, 9, 15, 20, 25	0.175, 0.550, 0.200, 0.025, 0.050	3	300	15	2	15	0 0 0 0 0 0 0 0	3
x_{29}	2, 3, 5, 6, 7	0.200, 0.300, 0.300, 0.150, 0.050	1	600	3	8	3	0 1 2 3 4 4 4 4	7
x_{30}	2, 4, 8, 16, 22	0.175, 0.550, 0.200, 0.025, 0.050	1	800	2	3	2	0 0 0 0 0 0 0 0	1
x_{31}	15, deterministic	-	3	1200	5	-	-	0 ——— 0	8
x_{32}	6, 9, 15, 20, 25	0.100, 0.200, 0.500, 0.100, 0.100	3	1000	2	8	1	0 1 1 1 1 1 1 1	8
x_{33}	17, deterministic	-	5	2400	1	-	-	0 ——— 0	12
x_{34}	25, deterministic	-	7	1000	2	-	-	0 ——— 0	7
x_{35}	0, dummy activity	-	0	0	0	-	-	0 ——— 0	0
x_{36}	2, 4, 5, 8, 15	0.200, 0.300, 0.300, 0.150, 0.050	0	600	4	8	3	0 1 2 1 1 1 1 2	4
x_{37}	6, 9, 15, 20, 25	0.150, 0.250, 0.250, 0.200, 0.150	3	400	30	16	- 2	0 0 0 0 0 0 0 0	3
x_{38}	2, 3, 5, 6, 7	0.100, 0.200, 0.500, 0.100, 0.100	1	450	25	15	- 3	0 0 0 0 0 0 0 0	1

$i = 40$

Table 1

If during the application of the "modified" out-of-kilter algorithm to $CIRC(s_x^{n+1})$

(a) a point value-alteration phase was performed then $(d_x^*)^{n+1}$, $x \in X$, $(\pi_i^*)^{n+1}$, $i \in V$, gives a better solution of SPPS than $(d_x^*)^n$, $x \in X$, $(\pi_i^*)^n$, $i \in V$,

(b) no point value-alteration phase was performed then $(d_x^*)^n$, $x \in X$, $(\pi_i^*)^n$, $i \in V$, is optimal for SPPS.

To ensure finiteness a restriction to rational data for the description of SPPS has to be made.

4. EXAMPLE

The dimension of the following example indicates that the suggested approach can handle problems of application-relevant size. For the project-digraph of Fig. 1 the random activity-completion-times are assumed to have five realizations which together with its associated probabilities and the crash completion-times y_x^o, the cost-terms b_x, o_x, the compensation cost-terms q_x^+, q_x^- are given in Tab. 1. Taking into consideration a project-completion-time constraint $\lambda=40$ the optimal solution d_x^* was optained in seven iterations starting with $s_x^o \equiv 0$, $x \in X$. For the computation about 15 seconds CPU-time on UNIVAC 1108 were needed. For other examples see Cleef and Gaul [3] or Gaul [11].

REFERENCES

[1] Charnes, A., Cooper, W.W., Thompson, G.L. (1964) Critical Path Analysis via Chance-Constrained and Stochastic Programming. Operations Research 12, pp. 460 - 470.

[2] Cleef, H.J. (1981) A Solution Procedure for the Two-Stage Stochastic Program with Simple Recourse. Zeitschrift für Operations Research 25, pp. 1 - 13.

[3] Cleef, H.J., Gaul, W. (1980) Project Scheduling via Stochastic Programming. SFB-72-Preprint 367, University of Bonn.

[4] Clingen, C.T. (1964) A Modification of Fulkerson's PERT Algorithm. Operations Research 12, pp. 629 - 632.

[5] Devroye, L.P. (1979) Inequalities for the Completion Times of Stochastic PERT Networks. Mathematics of Operations Research 4, pp. 441 - 447.

[6] Ford, L.R., Fulkerson, D.R. (1962) Flows in Networks. Princeton University Press.

[7] Fulkerson, D.R. (1961) A Network Flow Computation for Project Cost Curves. Management Sc. 7, pp. 167 - 178.

[8] Fulkerson, D.R. (1962) Expected Critical Path Length in PERT Type Networks. Operations Research 10, pp. 808 - 817.

[9] Fulkerson, D.R. (1964) Scheduling in Project Networks. Proc. IBM Scient. Comp. Symp. Comb. Problems, pp. 73 - 92.

[10] Gaul, W. (1981 a) Bounds for the Expected Duration of a Stochastic Project Planning Model. J. Infor. & Optimiz. Sc. 2, pp. 45 - 63.

[11] Gaul, W. (1981 b) Stochastische Projektplanung und Marketingprobleme. Paper presented at: 6. Symposium Operations Research, Augsburg.

[12] Harary, F., Norman, F.Z., Cartwright, D. (1965) Structural Models: An Introduction to the Theory of Directed Graphs. Wiley.

[13] Kall, P. (1976) Stochastic Linear Programming. Springer Verlag.

[14] Kleindorfer, G.B. (1971) Bounding Distributions for a Stochastic Acyclic Network. Operations Research 19, pp. 1586 - 1601.

[15] Neumann, K., Steinhardt, U. (1975) GERT-Networks and the Time-Oriented Evaluation of Projects. Lecture Notes in Economics and Mathematical Systems 172. Springer Verlag.

[16] Robillard, P., Trahan, M. (1976) Expected Completion Time in PERT Networks. Operations Research 24, pp. 177 - 182.

[17] Robillard, P., Trahan, M. (1977) The Completion Time of PERT Networks. Operations Research 25, pp. 15 - 29.

[18] Shogan, A.W. (1977) Bounding Distributions for a Stochastic PERT Network. Networks 7, pp. 359 - 381.

[19] Stancu-Minasian, I.M., Wets, M.J. (1976) A Research Bibliography in Stochastic Programming. Operations Research 24, pp. 1078 - 1119.

[20] Van Slyke, R.M. (1963) Monte Carlo Methods and the PERT Problem. Operations Research 11, pp. 839 - 860.