INSTITUT FÜR WIRTSCHAFTSTHEORIE
UND OPERATIONS RESEARCH
UNIVERSITÄT KARLSRUHE

ProGen/max: A New Problem Generator for Different Resource-Constrained Project Scheduling Problems with Minimal and Maximal Time Lags

Christoph Schwindt

Report WIOR-449

TECHNICAL REPORT

Kaiserstraße 12 · D - 76128 Karlsruhe · Germany

ProGen/max: A New Problem Generator for Different Resource-Constrained Project Scheduling Problems with Minimal and Maximal Time Lags

Christoph Schwindt

Report WIOR-449

July 1995

Abstract

We describe a new problem generator for three different types of resource-constrained project scheduling problems with minimal and maximal time lags. The cyclicity of the underlying project network requires special techniques for the parameter-driven generation of cycle structures. Two different approaches for the generation of cycle structures are presented: the direct method which inserts backward arcs into an acyclic network and the contraction method which constructs isolated strong components which are then contracted to a single node and incorporated into an acyclic network. Efficient algorithms for the direct method and the contraction method are provided.
The generation of the project's basic data as well as the resource demand and availability is based on ProGen by Kolisch et al.

Key Words

Project generator, activity-on-node networks, network measures, resource-constrained project scheduling, maximal time lags.

Contents

1. Introduction

Several network and problem generators for resource-constrained scheduling problems are known from literature (Kolisch et al. 1992, Demeulemeester et al. 1993, Agrawal et al. 1994). One of the best-known scheduling problems is the resource-constrained project scheduling problem (RCPSP). Until 1992, an inhomogeneous testset of Patterson (1984) has been used as a benchmark for algorithms. This testset includes problems published in Davis (1969), Patterson and Huber (1974), Davis and Patterson (1975), Talbot and Patterson, and Patterson (1984). Of course, these problems had not been generated using a unified and systematic approach controled by several network and resource-based parameters like network complexity or resource strength.

Additionally, the development of a new efficient branch-and-bound procedure for RCPSP by Demeulemeester and Herroelen (1992) has shown that all problems of the Patterson testset without exception belong to a class of "easy" problems, that is, they can be solved optimally within a very short amount of time. Kolisch et al. (1992) have shown that there are lower-sized problems which are much harder to solve.

Therefore, the empirical analysis of algorithms should be based on problem instances which have been generated systematically by a problem generator. The performance of the tested algorithms can then be evaluated depending on different problem measures.

The problem generator ProGen of Kolisch et al. creates problem instances of RCPSP or the generalized multi-mode problem MRCPSP. Several network measures like the number of nodes, the network complexity, the number of predecessors and successors of a node as well as parameters for the generation of the basic data and the resource constraints can be specified.

The network generator of Demeulemeester et al. creates acyclic weakly connected digraphs where each network structure (with given number of nodes and arcs) can be generated with exactly the same probability (strongly randomized networks). Due to the specific approach required for the strong randomness, other network measures such as redundancy cannot be observed.

Whereas the networks generated by Kolisch et al. and Demeulemeester et al. are so-called activity-on-node networks (A-on-N networks, that is, the activities are identified with the nodes of the project network, the arcs define time and precedence constraints), the generator of Agrawal et al. constructs activity-on-arc networks (A-on-A networks) for which the arcs correspond to the activities of a project. Control parameters are the number of nodes, the number of arcs, and the *CI*-index of reduction complexity (which is defined to be the minimum number of node reductions sufficient to reduce a series-parallel digraph to a single edge, cf. Bein et al. 1992).

An important generalization of RCPSP is problem RCPSP/max where maximal time lags between the start of activities define additional time constraints. Maximal time lags can, for instance, be used to model due dates, time-varying resource demands of activities, or time windows due to technological or organizational restrictions. For applications we refer to Neumann and Schwindt (1995).

A maximal time lag between activities i and j can be represented by a backward arc $<j,i>$ from the node corresponding to activity j to the node corresponding to activity i in the underlying A-on-N project network $\vec{N} = <V,E;c>$. $<j,i>$ is weighted with the negative corresponding maximal time lag T_{ij}^{max}. The introduction of backward arcs, however, generates cycle structures in $\vec{N}$. Thus, project networks modeling precedence and time con-

straints of RCPSP / max instances are no longer acyclic and we need specific techniques for the parameter-driven generation of cycle structures.
The new problem generator ProGen / max developed within this paper is partially based on ProGen for the basic data generation and the construction of acyclic network structures. The generation of resource demand and availability is adopted from ProGen without almost any modification and will only be sketched briefly. The main emphasis of this paper is on theoretical results for cyclic digraphs and methods incorporated in ProGen / max for an efficent construction of cyclic networks taking into account several measures like number, size, and density of cycle structures.
ProGen / max additionally includes an option which adapts the methodology of Agrawal et al. to the case of activity-on-node networks. All generated networks are based on a so-called skeleton with given reduction complexity *CI*. The skeleton is then modified by series and parallel expansion, which does not alter the reduction complexity *CI*. For details we refer to Agrawal et al. (1994).

The remainder of this paper is organized as follows. Section 2 is concerned with three different optimization models for resource-constrained project scheduling with minimal and maximal time lags, basic definitions of graph theory, theorems which will be used for the network generation algorithms, and several network measures which are known from literature. Two different approaches for the construction of cyclic networks, the direct and the contraction method, are presented in Section 3. In Section 4 we summarize the generation of resource constraints developed by Kolisch et al. (1992).

The problem generation can be outlined as follows:

Algorithm A1. Problem generation

(1) Basic data generation (cf. Subsection 3.1)

(2) Construction of the structure of the underlying project network (cf. Subsections 3.2 and 3.3)

(3) Determination of the activity durations (cf. Subsection 3.4)

(4) Determination of minimal and maximal time lags between activities (cf. Subsection 3.4)

(5) Generation of resource availability and resource demand (cf. Subsections 4.1 and 4.2, respectively)

2. Basic Concepts

2.1 Models for resource-constrained scheduling

ProGen / max generates instances of several types of multi-mode resource-constrained project scheduling problems with minimal and maximal time lags and renewable, non-renewable, and doubly-constrained resources: The resource-constrained project scheduling problem MRCPSP / max, the resource-leveling problem MRLP / max, and the resource investment problem MRIP / max. Of course, special cases, such as MRCPSP or RLP / max can be obtained as well, by fixing the number of maximal time lags to zero or, respectively, by restricting the number of possible execution modes for any activity to one.
In this subsection we give a formal definition of the three problems MRLP / max, MRCPSP / max, and MRIP / max. For more details and formulations as linear optimization problems with binary variables, we refer to Franck and Schwindt (1995).

We introduce the following notation:

$0, n+1$	dummy activities representing the start and the end of the project, respectively ($D_0 = D_{n+1} = 0$)
b_{jlm}	weight of arc $<j,l> \in E$ $(m \in M_j)$ with $b_{jlm} := \begin{cases} T_{jlm}^{min}, & \text{if there is a minimal time lag between activities } j \text{ and } l \\ -T_{lj}^{max}, & \text{if there is a maximal time lag between activities } l \text{ and } j \end{cases}$
c_i	integer-valued cost of one unit of resource i $(i \in R^\rho \cup R^\nu)$
D_{jm}	non-preemptable integer-valued duration of activity j scheduled in mode m $(j \in V, m \in M_j)$
E	arc set of the underlying project network $G' = < V, E >$
M_j	set of modes in which activity j can be performed $(j \in V)$
$R^\rho \cup R^\nu$	set of renewable, nonrenewable, and doubly-constrained resources
R_i^ρ	integer-valued limited capacity of renewable (doubly-constrained) resource i $(i \in R^\rho)$
R_i^ν	integer-valued limited capacity of nonrenewable (doubly-constrained) resource i $(i \in R^\nu)$
r_{ijm}^ρ	integer-valued per period usage of renewable (doubly-constrained) resource i performing activity j in mode m $(j \in V, i \in R^\rho, m \in M_j)$
r_{ijm}^ν	integer-valued total consumption of nonrenewable (doubly-constrained) resource i performing activity j in mode m $(j \in V, i \in R^\nu, m \in M_j)$
ST_j	start time of activity j $(j \in V)$
T	fixed project duration
$\overline{T}$	upper bound on the project duration

$T_{jlm}^{min}, T_{jl}^{max}$	minimal and maximal integer-valued time lags, respectively, between the start of activities j and l. The minimal time lag is depending on execution mode m of activity j
$V = \{0, \ldots, n+1\}$	set of activities which are to be performed. V at the same time corresponds to the node set of the underlying project network $\dot{G} = < V, E >$
$V(t)$	set of activities which are in progress at time t
x_{jm}	binary variable which is exactly one, if activity j is performed in mode m, zero otherwise $(j \in V, m \in M_j)$

MRCPSP / max can be stated as follows:

$$
(2.1) \quad (\text{MRCPSP / max}) \quad
\begin{cases}
\min & ST_{n+1} \\
\text{s.t.} & ST_l - ST_j \geq \sum_{m \in M_j} b_{jlm} x_{jm} \quad (< j, l > \in E) \\
& \sum_{j \in V(t)} \sum_{m \in M_j} r_{ijm}^{\rho} x_{jm} \leq R_i^{\rho} \quad (i \in R^{\rho}, t = 0, \ldots, \overline{T} - 1) \\
& \sum_{j \in V} \sum_{m \in M_j} r_{ijm}^{\nu} x_{jm} \leq R_i^{\nu} \quad (i \in R^{\nu}) \\
& \sum_{m \in M_j} x_{jm} = 1 \quad (j \in V) \\
& x_{jm} \in \{0, 1\} \quad (j \in V, m \in M_j)
\end{cases}
$$

The project duration is to be minimized. Minimal and maximal time lags between activities as well as limited availabilities of renewable, nonrenewable, and doubly-constrained resources have to be taken into account. Any job is performed in exactly one mode.

The resource leveling problem MRLP / max consists of the determination of a feasible schedule which minimizes a monotonously increasing function of the variation in time of resource requirements. Different objective functions can be found in literature. An objective function with many applications in practice is, for instance,

$$
(2.2) \quad f(ST_1, \ldots, ST_n) := \frac{1}{|R^{\rho}|} \sum_{i \in R^{\rho}} \max_{t = 0, \ldots, T-1} \sum_{j \in V(t)} r_{ijm}^{\rho} x_{jm}
$$

which corresponds to the mean resource availability which has to be provided if the resource availability (which is constant in time) is determined by the maximum resource requirements during the execution of the project.

MRLP / max can be stated as follows:

$$\text{(2.3)} \quad (\text{MRLP}/\max) \quad \begin{cases} \min & f(ST_1,\dots,ST_n) \\ \text{s.t.} & ST_l - ST_j \geq \sum_{m\in M_j} b_{jlm} x_{jm} \quad (\langle j,l\rangle \in E) \\ & ST_{n+1} \leq T \\ & \sum_{j\in V(t)} \sum_{m\in M_j} r^{\rho}_{ijm} x_{jm} \leq R^{\rho}_i \quad (i\in R^{\rho}, t=0,\dots,\overline{T}-1) \\ & \sum_{j\in V} \sum_{m\in M_j} r^{\nu}_{ijm} x_{jm} \leq R^{\nu}_i \quad (i\in R^{\nu}) \\ & \sum_{m\in M_j} x_{jm} = 1 \quad (j\in V) \\ & x_{jm} \in \{0,1\} \quad (j\in V, m\in M_j) \end{cases}$$

An objective function (depending on the schedule $(ST_1,\dots,ST_n)$) is to be minimized. Analogously to MRCPSP / max, precedence, time, and resource constraints must be observed and each activity is performed in exactly one mode. Additionally, the project has to be completed at a given point in time T.

The resource investment problem MRIP / max can be viewed as some kind of dualization of MRCPSP / max (cf. Demeulemeester 1992). The objective is the minimization of the costs for resource availability subject to the punctual completion of the project.

MRIP / max can be stated as follows:

$$\text{(2.4)} \quad (\text{MRIP}/\max) \quad \begin{cases} \min & \sum_{i\in R^{\rho}\cup R^{\nu}} c_i R^{\rho}_i \\ \text{s.t.} & ST_l - ST_j \geq \sum_{m\in M_j} b_{jl} x_{jm} \quad (\langle j,l\rangle \in E) \\ & ST_{n+1} \leq T \\ & \sum_{j\in V(t)} \sum_{m\in M_j} r^{\rho}_{ijm} x_{jm} \leq R^{\rho}_i \quad (i\in R^{\rho}, t=0,\dots,\overline{T}-1) \\ & \sum_{j\in V} \sum_{m\in M_j} r^{\nu}_{ijm} x_{jm} \leq R^{\nu}_i \quad (i\in R^{\nu}) \\ & \sum_{m\in M_j} x_{jm} = 1 \quad (j\in V) \\ & x_{jm} \in \{0,1\} \quad (j\in V, m\in M_j) \end{cases}$$

The only difference between MRIP / max and MRLP / max is made by the objective function.

2.2 Basic Definitions and Theorems

In this section, we provide some basic definitions and results for digraphs which are used in Section 3 for the generation of cycle structures. For an introduction into the theory of graphs and digraphs we refer to Bondy and Murty (1976), Berge (1985), or Neumann and Morlock (1993).

We introduce the following notation. The symbols refer to digraph $\dot{G} = < V, E >$ or to network $\dot{G} = < V, E; c >$, repectively:

$\boldsymbol{A}$	adjacency matrix
C	set of cycle structures
$C(i)$	cycle structure to which node $i \in V$ belongs; not defined, if $i \in V$ is not in the set of nodes of a cycle structure $C \in C$
c_{ij}	weight of arc $<i,j>$
$\delta^{-}(i)$	outdegree of node $i \in V$
$\delta^{+}(i)$	indegree of node $i \in V$
E	set of arcs
$\boldsymbol{I}$	identity matrix
$<i,j>$	arc from node $i \in V$ to node $j \in V$
$P(i)$	set of direct predecessors of node $i \in V$
R	set of sources
$\boldsymbol{R}$	reachability matrix
$R(i)$	set of nodes node $i \in V$ which are reachable from node i
$\bar{R}(i)$	set of nodes from which node $i \in V$ can be reached
S	set of sinks
$S(i)$	set of direct successors of node $i \in V$
V	set of nodes
$V(\dot{G})$	set of nodes of a digraph $\dot{G}$

We assume that digraph $\dot{G} = < V, E >$ is simple, that is, it contains no parallel arcs or directed loops.

Definition 1. Adjancency matrix $\boldsymbol{A}$ of a digraph

The adjacency matrix $\boldsymbol{A}$ of digraph $\dot{G} = < V, E >$ is defined to be the $n \times n$ matrix $\left(a_{ij}\right)_{i,j \in V}$ with $a_{ij} := \begin{cases} 1, \text{ if } < i, j > \in E \\ 0, \text{ otherwise} \end{cases}$.

Definition 2. Indegree and outdegree of node $i \in V$

The indegree $\delta^{-}(i)$ of node $i \in V$ is defined to be the number of (direct) predecessors of node i: $\delta^{-}(i) := |P(i)|$.

Analogously, the outdegree $\delta^{+}(i)$ of node $i \in V$ is defined to be the number of (direct) successors of node i: $\delta^{+}(i) := |S(i)|$

Definition 3. Reachability

A node $j \in V$ is called reachable from node i if $j = i$ or if there is a (directed) path W_{ij} with origin i and terminus j.

Definition 4. Reachability matrix of a digraph

The reachability matrix $\boldsymbol{R}$ of digraph $\vec{G} = <V, E>$ is defined to be the $n \times n$ matrix $\left(r_{ij}\right)_{i,j \in V}$ with $r_{ij} := \begin{cases} 1, & \text{if } j \text{ is reachable from } i \\ 0, & \text{otherwise} \end{cases}$.

Definition 5. Connectivity

Two nodes $i, j \in V$ are called connected if $r_{ij} = 1$ or $r_{ji} = 1$.

Definition 6. Subdigraph and subdigraph induced by node set

A digraph $\vec{G}' = <V', E'>$ represents a subdigraph of digraph $\vec{G} = <V, E>$ if $V' \subseteq V, E' \subseteq E$ and $(<i, j> \in E' \Rightarrow i, j \in V')$.
A digraph $\vec{G}'' = <V'', E''>$ represents the (unique) subdigraph of digraph $\vec{G} = <V, E>$ induced by node set V'' if $V'' \subseteq V$ and $(<i, j> \in E'' \Leftrightarrow i, j \in V'', <i, j> \in E)$.

Definition 7. Weak component

A weak component $\vec{G}' = <V', E'>$ of $\vec{G}$ is defined to be a maximal subdigraph of $\vec{G}$ (with respect to $|V'|$) induced by node set V' for which all nodes $i, j \in V'$ are connected. A digraph $\vec{G}$ which constitutes a weak component of itself is called weakly connected.

Remark 1.

If there is a subdigraph $\vec{G}' = <V', E'>$ of digraph $\vec{G} = <V, E>$ with $V' \subset V$, such that $\vec{G}'$ is a weak weak component of $\vec{G}$, then $\vec{G}$ is no weak component.

Definition 8. Network

An arc-weighted digraph $\vec{G} = <V, E; c>$ with $c: E \to \mathrm{IR}$ is called network if the underlying digraph $\vec{G} = <V, E>$ is weakly connected.

Definion 9. Strong component

A strong component $\dot{G}' = < V', E' >$ of $\dot{G}$ is defined to be a maximal subdigraph of $\dot{G}$ (with respect to $|V'|$) for which all nodes $i, j \in V'$ are mutuallay reachable. A digraph $\dot{G}$ which constitutes a strong component of itself is called strongly connected.

Remark 2.

Obviously, any strong component is a weak component, too.

Definition 10. Cycle structure

A cycle structure $C = < V', E' >$ of $\dot{G}$ is a strong component of $\dot{G}$ with $|V'| \geq 2$.

Remark 3.

A cycle structure $C = C(i) = < V', E' >$ with $i \in V'$ is the subdigraph of $\dot{G}$ induced by the node set $V' = \bar{R}(i) \cap R(i)$ with $\{i\} \subset V'$.

Definition 11. Contraction of a cycle structure.

The contraction of a cycle structure C in a digraph $\dot{G} = < V, E >$ is the assignment of a digraph $\dot{G}' = < V', E' >$ to $\dot{G}$ such that

$$
\begin{aligned}
&(i) && V' := V \setminus V(C) \cup \{c\} \\
& && E' := E \setminus \left\{ < i, j > \in E \,\middle|\, \{i, j\} \cap V' \neq \emptyset \right\} \\
&(ii) && \quad \cup \left\{ < c, j > \,\middle|\, \exists < i, j > \in E : i \in V(C) \right\} \\
& && \quad \cup \left\{ < i, c > \,\middle|\, \exists < i, j > \in E : j \in V(C) \right\}
\end{aligned}
$$

c is referred to as the contracted cycle structure C.

Definition 12. Expansion of a contracted cycle structure

Let c be the contracted cycle structure C in a digraph $\dot{G} = < V, E >$ and $\dot{G}' = < V', E' >$ a digraph with subdigraph $\dot{G}$. By the expansion of c with respect to $\dot{G}'$ we mean the assignment of a digraph $\dot{G}'' = < V'', E'' >$ to $\dot{G}$ such that

$$
\begin{aligned}
&(i) && V'' := V \cup (V(C) \cap V') \setminus \{c\} \\
&(ii) && E'' := E \cup \left\{ < i, j > \in E' \,\middle|\, \{i, j\} \cap V'' \neq \emptyset \right\} \setminus \left\{ < i, j > \in E \,\middle|\, c \in \{i, j\} \right\}.
\end{aligned}
$$

Definition 13. Redundant arc

An arc $<i,j>\in E$ is said to be redundant in $\dot{G}=<V,E>$ if there is a (directed) path W_{ij} in $\dot{G}$ which contains more than one arc.

Remark 4.

Obviously, it holds that: $<i,j>$ redundant $\Leftrightarrow$ $r'_{ij}=1$ for $\dot{G}'=<V,E\setminus\{<i,j>\}>$ with reachability matrix $\boldsymbol{R}'$.

Definition 14. Redundancy generating arc

An arc $<i,j>\in E$ is said to be redundancy generating in $\dot{G}=<V,E>$ if $<i,j>$ is redundant in $\dot{G}$ or if there are nodes $k,l\in V$ and a corresponding (directed) path W_{kl} in $\dot{G}$ such that $<k,l>$ is redundant in $\dot{G}$ and $<i,j>$ is part of W_{kl}.

Remark 5 (cf. Kolisch et al. 1992).

$<i,j>$ is redundancy generating in $\dot{G}$ $\Leftrightarrow$ one of the following four cases is true

(*i*) $j\in R(i)\setminus S(i)$
(*ii*) $\exists l\in R(j): P(l)\cap\bar{R}(i)\neq\emptyset$
(*iii*) $P(j)\cap\bar{R}(i)\neq\emptyset$
(*iv*) $S(i)\cap R(j)\neq\emptyset$

Theorem 1.

Let $\dot{G}=<V,E>$ be an acyclic digraph with reachability matrix $\boldsymbol{R}$ and $<i,j>\notin E$. Let $\dot{G}'=<V,E\cup\{<i,j>\}>$ be the digraph which is obtained by the insertion of $<i,j>$ into $\dot{G}$. Let be $\rho_{ij}^{(1)},\rho_{ij}^{(2)},\rho_{ij}^{(3)}$ $(i,j\in V)$ the following indices with corresponding $n\times n$ matrices $\mathrm{P}^{(1)},\mathrm{P}^{(2)}$, and $\mathrm{P}^{(3)}$:

$$(2.5)\quad \begin{cases} \rho_{ij}^{(1)} := \delta\left(\sum_{k\in\bar{R}(i)} \sum_{l\in R(j)} a_{kl} \right) \\ \rho_{ij}^{(2)} := \delta\left(\sum_{k\in P(i)} \sum_{l\in\bar{R}(i)} a_{kl} \right) \\ \rho_{ij}^{(3)} := \delta\left(\sum_{k\in S(i)} \sum_{l\in R(j)} a_{kl} \right) \end{cases} \quad (i,j\in V) \text{ with } \delta(x) := \begin{cases} 1, & \text{if } x>0 \\ 0, & \text{otherwise} \end{cases}.$$

In $\vec{G}'$ we obtain the following equivalences:

(i) $\quad j\in R(i)\setminus S(i) \Leftrightarrow r_{ij}=1$ in $\vec{G}$

(ii) $\quad (\exists l\in R(j): P(l)\cap\bar{R}(i)\neq\emptyset) \Leftrightarrow \rho_{ij}^{(1)}=1$ in $\vec{G}$

(iii) $\quad P(j)\cap\bar{R}(i)\neq\emptyset \Leftrightarrow \rho_{ij}^{(2)}=1$ in $\vec{G}$

(iv) $\quad S(i)\cap R(j)\neq\emptyset \Leftrightarrow \rho_{ij}^{(3)}=1$ in $\vec{G}$

Corollary 1.

Let Let $\vec{G}=\langle V,E\rangle$ be an acyclic digraph with $\langle i,j\rangle\notin E$. The insertion of arc $\langle i,j\rangle$ will generate redundancy (that is, $\langle i,j\rangle$ will be redundancy generating in $\vec{G}'=\langle V,E\cup\{\langle i,j\rangle\}\rangle$) exactly if $\rho_{ij} := r_{ij}+\rho_{ij}^{(1)}+\rho_{ij}^{(2)}+\rho_{ij}^{(3)}>0$.

Theorem 2.

The maximal number of arcs $\langle i,j\rangle$ which can be inserted into an acyclic digraph $\vec{G}=\langle V,E\rangle$ without redundant arcs, such that $\langle i,j\rangle$ is redundant in digraph $\vec{G}'=\langle V,E\cup\{\langle i,j\rangle\}\rangle$ and $\vec{G}'$ is acyclic is

$$(2.6)\quad m_{red}^{max} = \sum_{i,j\in V} r_{ij} - |V| - |E|.$$

Proof.

With Remark 4 and

(i) $\quad r_{ii}=1\ \forall i\in V$ and

(ii) $\quad \langle i,j\rangle\in E \Rightarrow r_{ij}=1$

we obtain

(2.7) $$m_{red}^{max} = \sum_{i \in V} \sum_{\substack{j \in V \setminus \{i\} \\ <i,j> \notin E}} r_{ij} = \sum_{i,j \in V} r_{ij} - \sum_{i \in V} r_{ii} - \sum_{<i,j> \in E} r_{ij} = \sum_{i,j \in V} r_{ij} - |V| - |E|$$ ❑

Theorem 3.

The maximal number of arcs $<i,j>$ which can be inserted into an acyclic digraph $\dot{G} = <V,E>$ without redundant arcs, such that $<i,j>$ is redundancy generating in $\dot{G}' = <V, E \cup \{<i,j>\}>$ and $\dot{G}'$ is acyclic is

(2.8) $$m_{redGen}^{max} = \sum_{\substack{i,j \in V \\ r_{ji}=0}} (r_{ij} + \rho_{ij}^{(1)} + \rho_{ij}^{(2)} + \rho_{ij}^{(3)}$$
$$-r_{ij}\rho_{ij}^{(1)} - r_{ij}\rho_{ij}^{(2)} - r_{ij}\rho_{ij}^{(3)} - \rho_{ij}^{(1)}\rho_{ij}^{(2)} - \rho_{ij}^{(1)}\rho_{ij}^{(3)} - \rho_{ij}^{(2)}\rho_{ij}^{(3)}$$
$$+r_{ij}\rho_{ij}^{(1)}\rho_{ij}^{(2)} + r_{ij}\rho_{ij}^{(1)}\rho_{ij}^{(3)} + r_{ij}\rho_{ij}^{(2)}\rho_{ij}^{(3)} + \rho_{ij}^{(1)}\rho_{ij}^{(2)}\rho_{ij}^{(3)}$$
$$-r_{ij}\rho_{ij}^{(1)}\rho_{ij}^{(2)}\rho_{ij}^{(3)})$$
$$-|E|$$

The proof basically consists of the application of Corollary 1.

Theorem 4.

Let $\dot{G} = <V,E>$ be a digraph with reachability matrix $\boldsymbol{R}$. Let $\boldsymbol{R}^2 := \left(r_{ij}^{(2)}\right)_{i,j \in V}$ be the squared reachability matrix. If $r_{ij} = 1$, the cardinal number of the node set V' of the cycle structure $C(i) = C(j)$ which results from the insertion of arc $<j,i>$ into $\dot{G}$ is

(2.9) $$|V'| = |V(C(i))| = |V(C(j))| = r_{ij}^{(2)}$$

Proof.

Due to $r_{ij} = 1$, the insertion of arc $<j,i>$ generates the cycle structure $C(i) = C(j)$ with node set $R(i) \cap \bar{R}(i) = R(j) \cap \bar{R}(j)$.

(*i*) $r_{ij}^{(2)} \le |V'|$:

$$r_{ij}^{(2)} = \sum_{k \in V} r_{ik} r_{kj} = \left|\{l \in V | l \in R(i) \cap \bar{R}(j)\}\right|.$$

After the insertion of $<j,i>$ we obtain: $k \in R(j) \cap \bar{R}(i) \; \forall k \in \{l \in V | l \in R(i) \cap \bar{R}(j)\}$.

$\Rightarrow k \in R(i) \cap \bar{R}(i) = R(j) \cap \bar{R}(j) = V(C(i)) \; \forall k \in \{l \in V | l \in R(i) \cap \bar{R}(j)\}$
$\Rightarrow \{l \in V | l \in R(i) \cap \bar{R}(j)\} \subseteq V(C(i))$
$\Rightarrow r_{ij}^{(2)} \leq |V(C(i))| = |V'|$

(ii) $r_{ij}^{(2)} \geq |V'|$

After the insertion of $< j, i >$, let $k \in V$ be a node of the cycle structure $C(i)$. Then, k is part of a path W_{ij} from node i to node j.
$\Rightarrow \forall k \in V(C(i))\colon r_{ik} r_{kj} = 1$
$\Rightarrow \sum_{k \in V(C(i))} r_{ik} r_{kj} = |V(C(i))|$
$\Rightarrow r_{ij}^{(2)} \geq |V(C(i))| = |V'|$

(i), (ii) $\Rightarrow r_{ij}^{(2)} = |V'|$ ❑

Remark 6.

For $i \neq j$ $r_{ij}^{(2)} \leq 1$ implies $r_{ij} = 0$, since
$r_{ij}^{(2)} = \sum_{k \in V} r_{ik} r_{kj} = \sum_{k \in V \setminus \{i,j\}} r_{ik} r_{kj} + r_{ii} r_{ij} + r_{ij} r_{jj} \geq r_{ii} r_{ij} + r_{ij} r_{jj} = 2 r_{ij} > 1$ if $r_{ij} = 1$.

Corollary 2.

Let $\dot{G} = < V, E >$ be a digraph with squared reachability matrix $\boldsymbol{R}^2$ and set of cycle structures C. The number $\Gamma := |C|$ of cycle structures in $\dot{G}$ is

$$(2.10) \quad \Gamma = \sum_{\substack{i \in V \\ r_{ii}^{(2)} > 1}} \frac{1}{r_{ii}^{(2)}}.$$

Corollary 3.

Let $\dot{G} = < V, E >$ be a digraph with adjacency matrix $\boldsymbol{A}$ and the squared reachability matrix $\boldsymbol{R}^2$. Arc $< i, j > \in E$ is redundant in $\dot{G}$ exactly if $a_{ij} = 1$ and $r_{ij}^{(2)} > 2$.

Lemma 1.

Let $\dot{G}$ be an acyclic weak component with two sinks (that is, $|S| \geq 2$). Then, for any sink $s \in S$, there are a further sink $s' \in S, s' \neq s$ and a source $r \in R$, such that $r \in \overline{R}(s) \cap \overline{R}(s')$.

Proof.

Since $\dot{G}$ is acyclic, for each sink $s \in S$ there is a source $r \in R$, such that $r \in \overline{R}(s)$.
Let us assume that for a given sink s there are no sink $s' \in S, s' \neq s$ and source $r \in R$, such that $r \in \overline{R}(s) \cap \overline{R}(s')$.
$\Rightarrow \overline{R}(s) \cap \overline{R}(s') = \emptyset \ \forall s' \in S: s' \neq s$
$\Rightarrow$ the subdigraph $\dot{G}'$ of $\dot{G}$ induced by $\overline{R}(s)$ is a weak component of $\dot{G}$, and $\dot{G}' \neq \dot{G}$ since $s' \notin \overline{R}(s)$.
$\Rightarrow$ $\dot{G}$ is not weakly connected, which contradicts the prerequisites. ❏

Lemma 2.

Let $\dot{G}$ be an acyclic weak component with two sources (that is, $|R| \geq 2$). Then, for any source $r \in R$, there are a further source $r' \in R, r' \neq r$ and a sink $s \in S$, such that $s \in R(r) \cap R(r')$.

The proof can be led analogously to Lemma 1.

Theorem 5.

Let $\dot{G}$ be an acyclic weak component with two sources and two sinks (that is, $|R| \geq 2$, $|S| \geq 2$). Then, there is always a sink $s \in S$ with $|\overline{R}(s) \cap R| \geq 2$, and for each sink $s \in S$ with $|\overline{R}(s) \cap R| \geq 2$ there are two corresponding sources $r, r' \in R, r \neq r'$ with $s \in R(r) \cap R(r')$, where r can be chosen such that there is a sink $s' \in S, s' \neq s$ with $r \in \overline{R}(s) \cap \overline{R}(s')$.

Proof.

Since $\dot{G}$ is acyclic and weakly connected with $|R| \geq 2$ and $|S| \geq 2$, the prerequisites of Lemmata 1 and 2 are met.
From Lemma 2 it follows that:
$\exists\, s \in S: (\exists\, r_1, r_2 \in R, r_1 \neq r_2: s \in \overline{R}(r_1) \cap \overline{R}(r_2))$.
With Lemma 1 we obtain that there is a further sink $s' \in S, s' \neq s$ such that there is a source $\hat{r} \in R$ with $\hat{r} \in \overline{R}(s) \cap \overline{R}(s')$.

(i) $\hat{r} \neq r_1, \hat{r} \neq r_2$

$\Rightarrow r_1, r_2, \hat{r} \in \overline{R}(s)$ and $\hat{r} \in \overline{R}(s')$.
By setting $r := \hat{r}, r' := r_1$ or $r' := r_1$, we obtain $s \in R(r) \cap R(r')$ and $r \in \overline{R}(s) \cap \overline{R}(s')$.

(ii) $\hat{r} = r_1$

$\Rightarrow r_2, \hat{r} \in \bar{R}(s)$ and $\hat{r} \in \bar{R}(s')$.
By setting $r := \hat{r}, r' := r_2$, we obtain $s \in R(r) \cap R(r')$ and $r \in \bar{R}(s) \cap \bar{R}(s')$.

(iii) $\hat{r} = r_2$

$\Rightarrow r_1, \hat{r} \in \bar{R}(s)$ and $\hat{r} \in \bar{R}(s')$.
By setting $r := \hat{r}, r' := r_1$, we obtain $s \in R(r) \cap R(r')$ and $r \in \bar{R}(s) \cap \bar{R}(s')$. ❑

Corollary 4.

Let $\dot{G}$ be an acyclic weak component with two sources and two sinks. Then, there are sources $r, r' \in R, r \neq r'$ and sinks $s, s' \in S, s \neq s'$ such that $r \in \bar{R}(s) \cap \bar{R}(s')$ and $s \in R(r')$.

Theorem 6.

Let $\dot{G}$ be an acyclic weak component with two sources and two sinks. Let $r, r' \in R, r \neq r'$ and $s, s' \in S, s \neq s'$ with $r \in \bar{R}(s) \cap \bar{R}(s')$ and $s \in R(r')$.
Then, the insertion of arc $< s, r >$ generates a cycle structure $C(r) = C(s)$ such that the contraction of $C(r)$ in $\dot{G}$ yields a digraph $\dot{G}'$ in which the contracted cycle structure c of $C(r)$ neither constitutes a source nor a sink.

Proof.

From Corollary 3 it follows that there are sources $r, r' \in R, r \neq r'$ and sinks $s, s' \in S, s \neq s'$ with $r \in \bar{R}(s) \cap \bar{R}(s')$ and $s \in R(r')$. We obtain:

(i) $r \in \bar{R}(s), s' \notin C(r) \Rightarrow c \in \bar{R}(c)$

(ii) $s \in R(r'), r' \notin C(r) \Rightarrow c \in R(r')$

(i), (ii) $\Rightarrow s' \in R(c)$ and $r' \in \bar{R}(c)$ which implies $R(c) \setminus \{c\} \neq \emptyset$ and $\bar{R}(c) \setminus \{c\} \neq \emptyset$. Hence, c neither constitutes a sink nor a source. ❑

Theorem 7.

Any acyclic digraph possesses one source and one sink.

Proof.

Let us assume that there is an acyclic digraph $\dot{G} = \langle V, E \rangle$ with $V = \{1, \ldots, n\}$ which has no sink, that is, any node is positively incident with at least one other node. Starting with node i_1 there is a node i_2 with $\langle i_1, i_2 \rangle \in E$, for node i_2 there is a node $i_3 \notin \{i_1, i_2\}$ ($\dot{G}$ is acyclic) with $\langle i_2, i_3 \rangle \in E$, etc. Finally, there is a node $i_n \notin \{i_1, \ldots, i_{n-1}\}$ with $\langle i_{n-1}, i_n \rangle \in E$. i_n is positively incident with a node $i_{n+1} \in V = \{i_1, \ldots, i_n\}$ which represents a contradiction to the acyclicity of $\dot{G}$.
The proof for the existence of a source can be led analogously. ❑

2.3 Network Measures

The structure of the underlying network generally has a strong impact on the time which an exact algorithm needs to solve a sequencing problem as well as on the deviation of a heuristic solution from the optimum.
In literature, a large number of network measures can be found which describe the size, the logic, and the shape of networks (cf. Thesen 1977, Elmaghraby and Herroelen 1980, Davis 1975, Kaiman 1974, Kutulus and Davis 1982, Patterson (1976)).
Table 1 summarizes the control parameters used by ProGen / max for the generation of the project network:

Measure	Definition
Number of nodes	$\lvert V \rvert$
Estimator for the restrictiveness (see below)	$RT = \dfrac{2 \sum_{i,j \in V} r_{ij} - 6(\lvert V \rvert - 1)}{(\lvert V \rvert - 2)(\lvert V \rvert - 3)}$
Degree of redundancy	$\rho = \dfrac{\lvert E \rvert - m_{nonRed}}{\lvert E \rvert}$
Number of predecessors and number of successors of a node	$\lvert P(i) \rvert, \lvert S(i) \rvert \ (i \in V)$
Number of cycle structures	$\Gamma := \lvert C \rvert = \sum_{\substack{i \in V \\ r_{ii}^{(2)} > 1}} \dfrac{1}{r_{ii}^{(2)}}$
Percentage of backward arcs	$\left\lvert \left\{ \langle j, i \rangle \in E \,\middle\vert\, r_{ij} = 1 \right\} \right\rvert$
Number of nodes in a cycle structure	$r_{ii}^{(2)} > 1 \ (i \in V)$

Table 1. Network measures

Instead of the most commonly used network complexity (that is, the ratio of the number of arcs to the number of nodes), we employ one of the approximations for the restrictiveness developed by Thesen (1977).

Definition 15: Restrictiveness of a digraph

Let $\dot{G} = < V, E >$ be a weakly connected digraph with exactly one source 0 and exactly one sink n+1 and node set $V = \{0,1,\ldots,n,n+1\}$. With Π we denote the number of possible perturbations $(i_1, i_2, \ldots, i_n)$ of $V' = \{1,\ldots,n\} \subseteq V$ such that $k < l \Rightarrow i_k \notin R(i_l)$. The restrictiveness is defined as $\mathrm{P} := 1 - \frac{\log \Pi}{\log n!}$.

Remark 7.

$\mathrm{P} \in [0,1]$, $\mathrm{P} = 0$ for parallel digraphs, and $\mathrm{P} = 1$ for series digraphs (cf. Thesen 1977).

Representing an exact measure for the cardinal number of the solution space, P would be an appropriate index of network complexity. The determination of Π, however, constitutes a hard combinatorial problem. That is why Thesen has tested a set of over 40 different estimators for P. With RT we denote that estimator which yielded the lowest mean relative error (with respect to P) in the empirical analysis of Thesen:

Definition 16. Estimator RT for the restrictiveness

Let $\dot{G} = < V, E >$ be a weakly connected digraph with exactly one source 0 and exactly one sink n+1, node set $V = \{0,1,\ldots,n,n+1\}$ and reachability matrix $\boldsymbol{R}$. Imaginary (undirected) edges with incident nodes $i, j \in V$ between which no precedence relation has been established (that is, $i \notin \bar{R}(j) \cup R(j)$) are called disjunctive arcs. Let n_d be the number of disjunctive arcs in digraph $\dot{G}$ and let $n_d^{max} = \frac{n(n-1)}{2}$ be the maximal number of possible disjunctive arcs in a weakly connected digraph with node set $V = \{0,1,\ldots,n,n+1\}$. Then, the restrictiveness estimator RT is defined to be

$$(2.11) \quad RT := 1 - \frac{n_d}{n_d^{max}} = 1 - \frac{(n+2)(n+3) - 2\sum_{i,j \in V} r_{ij}}{n(n-1)} = \frac{2\sum_{i,j \in V} r_{ij} - 6(n+1)}{n(n-1)}.$$

Thesen gives no theoretical reasons for the good performance of RT. The properties of RT stated in the following theorem confirm the results obtained by the empirical analysis.

Theorem 8.

For the restrictiveness estimator *RT* the following properties apply:

(*i*) $RT \in [0,1]$.

(*ii*) *RT*=0 exactly for parallel digraphs and (the unique) redundant extension.

(*iii*) *RT*=1 exactly for series digraphs and redundant extensions.

(*iv)* The insertion of a non-redundant arc increases *RT*.

(*v*) The insertion of a redundant arc does not affect *RT*.

Proof.

Let $R=\{0\}, S=\{n+1\} \Rightarrow r_{0j}=1\ \forall j \in V, r_{i,n+1}=1\ \forall i \in V, r_{i0}=0\ \forall i \in V,$ and $r_{n+1,j}=0\ \forall j \in V$.

(*ii*) parallel digraphs:

$\dot{G}$ is a parallel digraph $:\Leftrightarrow R(i) \cap V = \{i, n+1\}\ \forall i \in V \setminus \{0, n+1\}$
$\Rightarrow r_{ij}=0\ \forall i,j \in V \setminus \{0,n+1\}, i \neq j$. We obtain:

$$RT = 1-\frac{(n+2)(n+3)-2\sum_{i,j\in V} r_{ij}}{n(n-1)} = 1-\frac{(n+2)(n+3)-2\left(\sum_{i,j\in V\setminus R\setminus S} r_{ij}+\sum_{i\in R,j\in V} r_{ij}+\sum_{i\in V\setminus R,j\in S} r_{ij}\right)}{n(n-1)}$$
$$=1-\frac{(n+2)(n+3)-2(n+n+2+n+1)}{n(n-1)}$$
$$=1-\frac{(n+2)(n+3)-6n-6}{n(n-1)}=1-\frac{n^2-n}{n(n-1)}=1-1=0$$

(*iii*) series digraphs:

$\dot{G}$ is a series digraph:$\Leftrightarrow$ there is exactly one (directed) path $W_{0,n+1}$ from the unique source 0 to the unique sink *n*+1. It can easily be seen, that in that case $\sum_{i,j\in V} r_{ij} = \frac{|V|(|V|+1)}{2} = \frac{(n+2)(n+3)}{2}$. For *RT* we obtain:

$$RT = 1-\frac{(n+2)(n+3)-2\sum_{i,j\in V} r_{ij}}{n(n-1)} = 1-\frac{(n+2)(n+3)-2\frac{1}{2}(n+2)(n+3)}{n(n-1)}=1$$

(i)

Since $\dot{G}$ is weakly connected and acyclic, it holds that $3n+3 \le \sum_{i,j \in V} r_{ij} \le \frac{(n+2)(n+3)}{2}$.

(iv)

Obviously, the insertion of any arc $<i,j>(i,j \in V)$ into $\dot{G}$ cannot decrease $\sum_{i,j \in V} r_{ij}$, since there is no $k \in V$ for which elements are removed from the set $R(k)$.
If $<i,j>$ is non-redundant, r_{ij} will be set from 0 to 1, which increases $\sum_{i,j \in V} r_{ij}$.

(v)

If $<i,j>$ is redundant, there is a (directed) path W_{ij} from i to j which contains more than one arc. In that case, there will be no $(k,l) \in V \times V$ such that r_{kl} is set from 0 to 1. ❑

Remark 8.

Let $ST := (n!)^{1-RT}$ be the estimator of Π based on RT. For cases (ii) and (iii) of Theorem 8, ST represents the exact number of feasible perturbations Π. Properties (iv) and (v) give a (partial) explanation for the good behaviour of estimator ST even for digraphs which are not parallel or series, since the insertion of non-redundant arcs always decreases the number of feasible perturbations, whereas the introduction of additional redundant arcs does not influence the precedence constraints.

Because of the good approximation of the state space size, ST probably has a strong impact on the hardness of instances of project scheduling problems. Recently, de Reyck and Herroelen (1994) investigated the relation between the hardness of problem instances and the reduction complexity CI of the underlying project network for RCPSP and the time/cost tradeoff problem. We conjecture that the more intuitive measure ST will even play a more important role for the computing effort required to solve instances of RCPSP(/max). This hypothesis has to be verified by a future computational study comparing ST and CI with respect to their impact on the hardness of problem instances.

3. Generation of the Basic Data and the Network

3.1 Generation of the Basic Data

The user of ProGen/max has to enter values for the following basic data which will be used for the construction of the project network and the generation of the resource data:

n^{min}, n^{max}	minimal and maximal number of activities
M^{min}, M^{max}	minimal and maximal number of modes per activity
R^{min}, R^{max}	minimal and maximal number of resources
$p_\rho \in [0,1],\ p_\nu \in [0,1]$	percentage of renewable and nonrenewable resources, respectively (the percentage of doubly-constrained resources is $p_\delta := 1 - p_\rho - p_\nu$)
c^{min}, c^{max}	minimal and maximal value for the generation of cost coefficients c_i $(i \in R^\rho \cup R^\nu)$

Let $rand\{a,\ldots,b\}\ (a,b \in \mathrm{IN}_0)$ be an integer pseudo random number out of the set $\{a,\ldots,b\}$ and let $rand[a,b]\ (a,b \in \mathrm{IR})$ be a real pseudo random number out of the interval $[a,b]$, both based on a uniformly distributed pseudo random number generated with the congruence-generator of Schrage (cf. Schrage 1979). With $\mathrm{int}(x)\ (x \geq 0)$ we denote the rounded value of x: $\mathrm{int}(x) := \max\left\{\arg \min_{z \in \mathrm{IN}_0} \{|z - x|\}\right\}$.

The basic data is calculated as follows:

- number of activities: $n := rand\{n^{min}, n^{max}\}$
- number of modes of activity j: $|M_j| := rand\{M^{min}, M^{max}\}\ (j \in V \setminus \{0, n+1\})$
- number of resources: $|R^\rho \cup R^\nu| := rand\{R^{min}, R^{max}\}$
- number of renewable resources: $|R^\rho \setminus R^\nu| := \mathrm{int}(p_\rho |R^\rho \cup R^\nu|)$
- number of nonrenewable resources: $|R^\nu \setminus R^\rho| := \mathrm{int}(p_\nu |R^\rho \cup R^\nu|)$
- number of doubly-constrained resources: $|R^\nu \cap R^\rho| := \mathrm{int}(p_\delta |R^\rho \cup R^\nu|)$
- cost coefficients: $c_i := rand\{c^{min}, \ldots, c^{max}\}\ (i \in R^\rho \cup R^\nu)$

3.2 Acyclic Network Structure

Let $\dot{G} = \langle V, E \rangle$ be the weakly connected digraph which represents the topological structure of a project network $\dot{N} = \langle V, E; c \rangle$ under consideration. Precedence and time constraints of corresponding instances of problems MRCPSP/max, MRLP/max, and MRIP/max (cf. (2.1), (2.3), and (2.4)) are given by $\dot{G}$ and minimal and maximal time lags between activities.

Obviously, several minimal time lags ${}^{1}T_{ij}^{min},\ldots,{}^{n_{ij}}T_{ij}^{min}$ between two activities i and j can be replaced by $\max_{\nu=1,\ldots,n_{ij}} {}^{\nu}T_{ij}^{min}$, whereas several maximal time lags $T_{ij}^{max},\ldots,{}^{n_{ij}}T_{ij}^{max}$ between two activities i and j can be replaced by $\min_{\nu=1,\ldots,n_{ij}} {}^{\nu}T_{ij}^{max}$. A minimal time lag T_{ij}^{min} between the start of activity i and the start of activity j can be represented by an arc $<i,j>$ weighted by $c_{ij} := T_{ij}^{min} \geq 0$. A maximal time lag T_{ij}^{max} between the start of activity i and the start of activity j can be represented by a (backward) arc $<j,i>$ weighted by $c_{ji} := -T_{ij}^{max} \leq 0$. Negative minimal time lags can be considered as positive maximal time lags and vice versa (cf. Neumann and Schwindt 1995).
If there is a minimal time lag $T_{ij}^{min} > 0$ and a maximal time lag $T_{ji}^{max} > 0$ which both would be represented by an arc $<i,j>$, the maximal time lag T_{ji}^{max} can be neglected since it will always be met if we observe the minimal time lag T_{ij}^{min}. That is why there will be no parallel arcs in digraph $\dot{G}$ and the corresponding project network $\dot{N}$. Since a precedence or a time constraint concerning a single activity does not make sense in project scheduling, we can rule out the case of loops and the generation of the network structure can be limited to the case of simple digraphs.

We consider two methods for the generation of cyclic network structures.
The first algorithm, called *direct method* in the following, generates an acyclic, generally not connected digraph, inserts backward arcs to create cycle structures and introduces a supersource and a supersink in order to obtain a weakly connected digraph, which represents the structure of the project network.
The second algorithm, called *contraction method* in the following, first creates isolated cycle structures which are then contracted, that is, shrunk down to a single node. With the contracted cycle structures and the nodes not employed during the first step we construct an acyclic, generally disconnected digraph, similarly to the direct method. Subsequently, the contracted cycle structures are expanded and integrated into the digraph. Finally, we introduce a supersource and a supersink in order to obtain a weakly connected digraph.

***Algorithm A2**. Direct method*

(1) Generation of an acyclic digraph without redundancy (cf. Algorithm A4)

- (1.1) Selection of sources and sinks (nodes which will correspond to initial and terminal activities)
- (1.2) Generation of direct predecessors
- (1.3) Generation of direct successors
- (1.4) Insertion of additional arcs such that the digraph remains without redundany

(2) Insertion of redundancy-generating arcs (cf. Algorithm A5)

(3) Creation of cycle structures (cf. Algorithm A6)

(3.1) Creation of new cycle structures
(3.2) Extension of cycle structures
(3.3) Densification of cycle structures.

(4) Weakly connection of the digraph (cf. Algorithm A7)

Algorithm A3. *Contraction method (cf. Algorithm A8)*

(1) Generation of cycle structures

(1.1) Generation of several isolated acyclic digraphs (cf. Algorithms A3, A4)
(1.2) Transformation of the isolated acyclic digraphs into cycle structures (cf. Algorithm A9)

(2) Contraction of the cycle structures

(3) Generation of a single acyclic digraph based on the contracted cycle structures and further conventional nodes (cf. Algorithm A3, A4)

(4) Expansion of the contracted cycle structures and integration into the digraph

(5) Weak connection of the digraph (cf. Algorithm A7)

Subsection 3.2 deals with the generation of an acyclic digraph, which will be used in the direct and in the contraction method. In Subsection 3.3 we treat the case of cycle structures.

3.2 Acyclic Network Structure

Let us be given the following input data for the generation of acyclic digraph $\dot{G} = < V, E >$:

$V = \{1,\ldots,n\}$	set of nodes
$P_{n+1}^{min}, P_{n+1}^{max}$	minimum and maximum number of sinks in $\dot{G}$
S_0^{min}, S_0^{max}	minimum and maximum numner of sources in $\dot{G}$
P_j^{max}	maximum number of non-redundant arcs entering node $j \in V$
S_i^{max}	maximum number of non-redundant arcs leaving node $i \in V$
RT	restrictiveness of Thesen for acyclic weak components
ρ	degree of redundancy in $\dot{G}$, that is, the percentage of redundant arcs in E relative to m_{red}^{max}

Algorithm A4. Generation of an acyclic digraph without redundancy

Set node set $V := \{1, \ldots, n\}$ and the arc set $E := \emptyset$. Initialize adjacency matrix $\boldsymbol{A} := \boldsymbol{0}$ and reachability matrix $\boldsymbol{R} := \boldsymbol{I}$.

(1) *Generation of sources and sinks*

Determine a random number $r \in \left\{S_0^{min}, \ldots, S_0^{max}\right\}$ of sources and a random number $s \in \left\{P_{n+1}^{min}, \ldots, P_{n+1}^{max}\right\}$ of sinks.
Let nodes $1, \ldots, r$ be the sources of G': $P_j^{min} := P_j^{max} := 0 \; \forall j \in R := \{1, \ldots, r\}$.
Let nodes $n - s + 1, \ldots, n$ be the sinks of G': $S_i^{min} := S_i^{max} := 0 \; \forall i \in S := \{n - s + 1, \ldots, n\}$.

(2) *Generation of direct predecessors*

Let V_p be a random perturbation of node set $V \setminus R$.

WHILE $V_p \neq \emptyset$ DO
  $j :=$ Head(V_p).
  $V_p := V_p \setminus \{j\}$.
  Determine set P_j of possible predecessors of node *j*:
  $P_j := \left\{ i \in V \middle| \rho_{ij} = 0, r_{ji} = 0, \delta^+(i) = 0 \right\}$.
  IF $P_j = \emptyset$ THEN
    $P_j := \left\{ i \in V \middle| \rho_{ij} = 0, r_{ji} = 0, \delta^+(i) < S_i^{max} \right\}$.
  END (* IF *).
  Select randomly a node $i \in P_j$.
  Insert arc $< i, j >$ into G': $E := E \cup \{< i, j >\}$.
  Update sets $P_j, P, S, \bar{R}$, and R :
  $P_j := P_j \setminus (\bar{R}(i) \cup R(i))$
  $P(j) := P(j) \cup \{i\}$.
  $S(i) := S(i) \cup \{j\}$.
  $\bar{R}(l) := \bar{R}(l) \cup \{k\} \; \forall (k, l): k \in \bar{R}(i), l \in R(j)$.
  $R(k) := R(k) \cup \{l\} \; \forall (k, l): k \in \bar{R}(i), l \in R(j)$.
  Update matrices $\boldsymbol{A}$, $\boldsymbol{R}$, and $\boldsymbol{R}^2$ (cf. Figure 1):
  $\boldsymbol{A}$: $a_{ij} := 1$.
  $\boldsymbol{R}$: $r_{kl} := 1 \; \forall (k, l): k \in \bar{R}(i), l \in R(j)$.
  $\boldsymbol{R}^2$: $r_{kl}^{(2)} := \left| R(k) \cap \bar{R}(l) \right| \; \forall (k, l): k \in \bar{R}(i), l \in R(j)$.
END (* WHILE *).

Fig. 1. Insertion of arc $<i,j>$

(3) *Generation of direct successors*

Let V_p be a random perturbation of node set $V \setminus S$.

WHILE $V_p \neq \emptyset$ DO

$i :=$ Head(V_p).

$V_p := V_p \setminus \{i\}$.

Determine set S_i of possible successors of node i:

$S_i := \left\{ j \in V \middle| \rho_{ij} = 0, r_{ji} = 0, \delta^-(j) < P_j^{max} \right\}$.

Select randomly a node $j \in S_i$.

Insert arc $<i,j>$ into $\dot{G}$: $E := E \cup \{<i,j>\}$.

Update sets $S_i, P, S, \bar{R}$, and R:

$S_i := S_i \setminus (\bar{R}(j) \cup R(j))$

$P(j) := P(j) \cup \{i\}$.

$S(i) := S(i) \cup \{j\}$.

$\bar{R}(l) := \bar{R}(l) \cup \{k\} \; \forall (k,l): k \in \bar{R}(i), l \in R(j)$.

$R(k) := R(k) \cup \{l\} \; \forall (k,l): k \in \bar{R}(i), l \in R(j)$.

Update matrices $\boldsymbol{A}$, $\boldsymbol{R}$, $\boldsymbol{R}^2$:

$\boldsymbol{A}$: $a_{ij} := 1$.

$\boldsymbol{R}$: $r_{kl} := 1 \; \forall (k,l): k \in \bar{R}(i), l \in R(j)$.

$\boldsymbol{R}^2$: $r_{kl}^{(2)} := \left| R(k) \cap \bar{R}(l) \right| \; \forall (k,l): k \in \bar{R}(i), l \in R(j)$.

END (* WHILE *).

(4) *Generation of additional non-redundancy generating arcs*

Determine the estimator for the restrictiveness rt of $\dot{G}$:

$rt = \dfrac{2\sum_{i,j\in V} r_{ij} - 2n}{n(n-1)}$ (cf. 2.11; notice that the supersource 0 and the supersink $n+1$ have not been introduced).

IF $rt < RT$ THEN

Determine the set of nodes P whose indegree can be increased:

$P := \left\{ i \in V \middle| \delta^{+}(i) < S_i^{max} \right\}$.

WHILE $rt < RT$ DO

Select randomly a node $i \in P$ where the probability for node i to be chosen is set to $p_i := \dfrac{\sum_{j\in V\setminus R} w_{ij}}{\sum_{k\in P}\sum_{j\in V\setminus R} w_{kj}}$.

Determine the set S_i of possible successors of node i:

$S_i := \left\{ j \in V \middle| \rho_{ij} = 0, r_{ji} = 0, \delta^{-}(j) < P_j^{max} \right\}$.

IF $S_i = \emptyset$ THEN

$P := P \setminus \{i\}$

ELSE

Select randomly a node $j \in S_i$.

Insert arc $< i, j >$ into $\dot{G}$: $E := E \cup \{< i, j >\}$.

Update sets $P, P, S, \overline{R}$, and R :

IF $\delta^{+}(i) = S_i^{max}$ THEN

$P := P \setminus \{i\}$.

END (* IF *).

$P(j) := P(j) \cup \{i\}$.

$S(i) := S(i) \cup \{j\}$.

$\overline{R}(l) := \overline{R}(l) \cup \{k\}\ \forall (k,l): k \in \overline{R}(i), l \in R(j)$.

$R(k) := R(k) \cup \{l\}\ \forall (k,l): k \in \overline{R}(i), l \in R(j)$.

Update matrices $\boldsymbol{A}$, $\boldsymbol{R}$, $\boldsymbol{R}^2$, and restrictiveness estimator rt:

$\boldsymbol{A}$: $a_{ij} := 1$.

rt: $rt := rt + \dfrac{2}{n(n-1)} \left| \left\{ (k,l) \in V \times V \middle| r_{kl} = 0, k \in \overline{R}(i), l \in R(j) \right\} \right|$.

$\boldsymbol{R}$: $r_{kl} := 1\ \forall (k,l): k \in \overline{R}(i), l \in R(j)$.

$\boldsymbol{R}^2$: $r_{kl}^{(2)} := \left| R(k) \cap \overline{R}(l) \right|\ \forall (k,l): k \in \overline{R}(i), l \in R(j)$.

END (* IF *).

END (* WHILE *).

END (* IF *). ❑

Algorithm 5. Insertion of redundant arcs into a redundancy-less digraph

Determine the number of redundant arcs to be inserted into $\dot{G}$: $m_{red} := \lfloor \rho\, m_{red}^{max} \rfloor$ (cf. Theorem 2).

$m_{nonRed} := |E|$.

$P := \{ i \in V \mid \delta^+(i) < |V \setminus R| \} \setminus S$.

WHILE $|E| < m_{nonRed} + m_{red}$ DO

Select randomly a node $i \in P$ where the probability for node *i* to be chosen is set to

$$p_i := \frac{\sum_{j \in V \setminus R} w_{ij}}{\sum_{k \in P} \sum_{j \in V \setminus R} w_{kj}}.$$

IF *i* has been selected for the first time THEN

Determine the set S_i of possible successors of node *i*:

$S_i := \{ j \in V \setminus R \mid r_{ij} = 1, a_{ij} = 0, r_{ji} = 0 \}$.

END (* IF *).

IF $S_i = \varnothing$ THEN

$P := P \setminus \{i\}$.

ELSE

Select randomly a node $j \in S_i$.

Insert arc $<i, j>$ into $\dot{G}$: $E := E \cup \{<i, j>\}$.

Update sets P, S_i, P, and S ($\bar{R}$ and R remain unchanged):

IF $\delta^+(i) = |V \setminus R|$ THEN

$P := P \setminus \{i\}$.

END (* IF *).

$S_i := S_i \setminus \{j\}$.

$P(j) := P(j) \cup \{i\}$.

$S(i) := S(i) \cup \{j\}$.

Update matrix $\boldsymbol{A}$ (matrices $\boldsymbol{R}$ and $\boldsymbol{R}^2$ remain unchanged):

$\boldsymbol{A}$: $a_{ij} := 1$.

END (* IF *).

END (* WHILE *). ❑

Remark 9.

The application of Algorithm A5 does not influence the restrictiveness of $\dot{G}$.

Remark 10.

The insertion of an arc $<i,j>$ in Algorithm A5 generates exactly one redundant arc, namely $<i,j>$. Replacing $r_{ij} = 1$ by $\rho_{ij} > 0$ for the determination of set S_i, we would not restrict ourselves to redundant arcs $<i,j>$ but could also insert redundancy generating arcs $<i,j>$, which are not redundant. In this case, however, the insertion of arc $<i,j>$ would increase the restrictiveness of $\dot{G}$.

3.3 Cycle Structures

Definition 17. Extension of a cycle structure $C(i)$ in a cyclic digraph $\dot{G}$

Let $\dot{G} = <V,E>$ be a cyclic digraph with set of cycle structures $\mathcal{C} \neq \emptyset$. By an extension of a cycle structure $C(i) \in \mathcal{C}$ we mean an operation

$$E: \mathcal{C} \to \mathcal{C}, C(i) \mapsto E(C(i)) = C'(i)$$

on a cycle structure such that $V(C'(i)) \supset V(C(i)), E(C'(i)) := \{<k,l> \in E \mid k,l \in V(C'(i))\}$.

Definition 18. Densification of a cycle structure $C(i)$ in a cyclic digraph $\dot{G}$

Let $\dot{G} = <V,E>$ be a cyclic digraph with set of cycle structures $\mathcal{C} \neq \emptyset$. By a densification of a cycle structure $C(i) \in \mathcal{C}$ we mean an operation

$$E: \mathcal{C} \to \mathcal{C}, C(i) \mapsto E(C(i)) = C'(i)$$

on a cycle structure such that $V(C'(i)) = V(C(i)), E(C'(i)) \supset E(C(i))$.

Let us be given the following input data for the generation of cycle structures in the acyclic digraph $\dot{G} = <V,E>$:

$MTL^{min}, MTL^{max} \in [0,1]$	minimum and maximum percentage of maximum time lags, respectively
CS^{min}, CS^{max}	minimum and maximum number of cycle structures, respectively
n_c^{min}, n_c^{max}	minimum and maximum cardinal number of a cycle structure, respectively
$\delta \in [0,1]$	percentage of arcs employed for cycle structure densification

Algorithm 6. Generation of the structure of a cyclic network: Direct method

Determine randomly the number t of backward arcs corresponding to maximal time lags which will be inserted into $\dot{G}$: $t \in \left\{\left\lceil |E| MTL^{min} \right\rceil, \dots, \left\lfloor |E| MTL^{max} \right\rfloor\right\}$.
Determine randomly the number CS of cycle structures which are to be generated in $\dot{G}$: $CS \in \left\{CS^{min}, \dots, \min\left\{t, CS^{max}\right\}\right\}$.
Initialize the set of (contracted) cycle structures $C := \varnothing$.

(1) *Generation of new cycle structures*

Set $P := V$.
WHILE $\Gamma := \sum_{\substack{i \in V \\ r_{ii}^{(2)} > 1}} \frac{1}{r_{ii}^{(2)}} < CS$ DO (cf. Corollary 2)

Select randomly a node $i \in P$.
Determine the set T_i of nodes j for which a maximal time lag T_{ij}^{max} (a corresponding backward arc $< j,i >$, respectively) can be introduced:
IF $t < CS - \Gamma$ THEN

$$T_i := \left\{ j \in V \setminus \{i\} \,\middle|\, r_{ij} = 1, \sum_{c \in C} r_{ic} r_{cj} = 0, r_{ij}^{(2)} \le n_c^{max} \right\} \text{ (cf. Theorem 4).}$$

ELSE

$$T_i := \left\{ j \in V \setminus \{i\} \,\middle|\, \sum_{c \in C} r_{ic} r_{cj} = 0, n_c^{min} \le r_{ij}^{(2)} \le n_c^{max} \right\}.$$

END (* IF *).
Select randomly a node $j \in T_i$.
Insert arc $< j,i >$ into $\dot{G}$: $E := E \cup \left\{< j,i >\right\}$.
$t := t - 1$.
Introduce the contracted cycle structure c of $C(i)$: $C := C \cup \{c\}$.
Update sets P, P, S, R, and $\overline{R}$:
$P := P \setminus (\overline{R}(j) \cap R(i))$
$P(i) := P(i) \cup \{j\}$.
$S(j) := S(j) \cup \{i\}$.
$R(h) := R(h) \cup \{g\} \; \forall (h,g): h \in \overline{R}(j), g \in R(i)$.
$\overline{R}(g) := \overline{R}(g) \cup \{h\} \; \forall (h,g): h \in \overline{R}(j), g \in R(i)$.

Update matrices $\boldsymbol{A}$, $\boldsymbol{R}$, and $\boldsymbol{R}^2$ (cf. Figure 2):

$\boldsymbol{A}$: $a_{ji} := 1$.

$\boldsymbol{R}$: $r_{hg} := 1 \ \forall\, (h,g){:}\ h \in \overline{R}\,(j), g \in R\,(i)$.

$\boldsymbol{R}^2$: $r_{hg}^{(2)} := \left|\overline{R}\,(g) \cap R\,(h)\right| \ \forall (h,g){:}\ h \in \overline{R}\,(j), g \in R\,(i)$.

Insert new column c and new row c into matrices $\boldsymbol{R}$ and $\boldsymbol{R}^2$:

$r_{hc} := 1 \ \forall h \in \overline{R}\,(j)$.

$r_{cg} := 1 \ \forall g \in R\,(i)$.

END (* WHILE *).

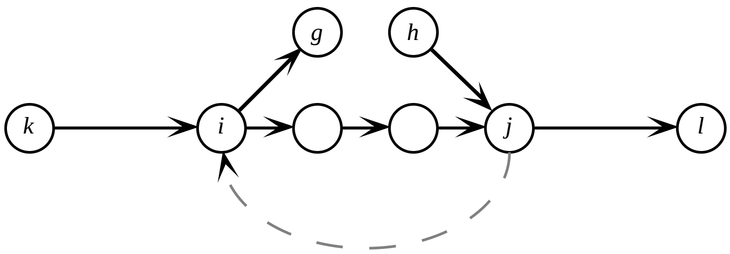

Fig. 2. Insertion of arc $< j,i >$

(2) *Extension of cycle structures*

Let $C' \subseteq C$ be the set of cycle structures $C(i)$ which have not attained the minimal node set cardinal number.

Determine the number t_e of arcs to be used for the extension of cycle structures: $t_e := |C'| + \lfloor (1-\delta)(t - |C'|) \rfloor$.

Set $P := V$.

WHILE $t_e > 0$ DO

IF $C \neq \emptyset$ THEN

$P := C$.

END (* IF *).

Select randomly a node $i \in P$.

Determine the set T_i of nodes j for which a maximal time lag T_{ij}^{max} (a corresponding backward arc $< j,i >$, respectively) can be introduced:

$$T_i := \left\{ j \in V \setminus \{i\} \,\middle|\, r_{ij} = 1, \sum_{c \in C} r_{ic} r_{cj} = 1, n_c^{min} \le r_{ij}^{(2)} \le n_c^{max} \right\}.$$

IF $T_i = \emptyset$ THEN

$P := P \setminus \{i\}$.

ELSE

Select randomly a node $j \in T_i$.

Insert arc $<j,i>$ into $\dot{G}$: $E := E \cup \{<j,i>\}$.

$t := t - 1$.

$t_e := t_e - 1$.

Update sets P, P, S, R, and $\bar{R}$:

$P(i) := P(i) \cup \{j\}$.

$S(j) := S(j) \cup \{i\}$.

$R(h) := R(h) \cup \{g\}\ \forall (h,g): h \in \bar{R}(j), g \in R(i)$.

$\bar{R}(g) := \bar{R}(g) \cup \{h\}\ \forall (h,g): h \in \bar{R}(j), g \in R(i)$.

Update matrices $\boldsymbol{A}$, $\boldsymbol{R}$, and $\boldsymbol{R}^2$:

$\boldsymbol{A}$: $a_{ji} := 1$.

$\boldsymbol{R}$: $r_{hg} := 1\ \forall (h,g): h \in \bar{R}(j), g \in R(i)$,

$\boldsymbol{R}^2$: $r_{hg}^{(2)} := |\bar{R}(g) \cap R(h)|\ \forall (h,g): h \in \bar{R}(j), g \in R(i)$.

Let c be the contracted cycle structure C with $\{i,j\} \cap V(C) \neq \emptyset$.

$r_{hc} := 1\ \forall h \in \bar{R}(j)$.

$r_{cg} := 1\ \forall g \in R(i)$.

END (* IF *).

END (* WHILE *).

(3) *Densification of cycle structures*

Set $P := V$.

WHILE $t > 0$ DO

Select randomly a node $i \in P$.

IF i has been selected for the first time THEN

Determine the set T_i of nodes j for which a maximal time lag T_{ij}^{max} (a corresponding backward arc $<j,i>$, respectively) can be introduced:

$T_i := \{j \in V \setminus \{i\} | a_{ji} = 0, r_{ij} = 1, r_{ji} = 1\}$.

END (* IF *).

IF $T_i = \emptyset$ THEN

$P := P \setminus \{i\}$.

ELSE

Select randomly a node $j \in T_i$.

Insert arc $<j,i>$ into $\dot{G}$: $E := E \cup \{<j,i>\}$.

$t := t - 1$.

Introduce the contracted cycle structure c of $C(i)$: $C := C \cup \{c\}$.

Update sets T_i, P, and S (R and $\bar{R}$ remain unchanged):

$T_i := T_i \setminus \{j\}$.
$P(i) := P(i) \cup \{j\}$.
$S(j) := S(j) \cup \{i\}$.
Update matrix $\boldsymbol{A}$ ($\boldsymbol{R}$ and $\boldsymbol{R}^2$ remain unchanged):
$\boldsymbol{A}$: $a_{ji} := 1$.
END (* IF *).
END (* WHILE *). ❑

Let $\dot{G} = < V, E >$ be a digraph. The following algorithm introduces a supersource 0 and a supersink $n+1$, thus transforming $\dot{G}$ into a weakly connected digraph.

Algorithm A7. Introduction of supersource 0 and supersink n+1

$V := V \cup \{0, n+1\}$.
$E := E \cup \{< 0, j > | j \in R\} \cup \{< i, n+1 > | i \in S\}$.
$a_{0,j} := 1 \, \forall j \in R$, $a_{i,n+1} := 1 \, \forall i \in S$.
$P(j) := P(j) \cup \{0\} \, \forall j \in R$, $S(i) := S(i) \cup \{n+1\} \, \forall i \in S$
$r_{0,j} := 1 \, \forall j \in V$, $r_{i,n+1} := 1 \, \forall i \in V$.
$R(0) := V, \overline{R}(n+1) := V$. ❑

The direct method for the generation of cyclic networks consists of the application of Algorithms A5, A6, and A7. Since cycle structures are constructed by the subsequent insertion of backward arcs in the acyclic digraph, it can happen that a (feasible) number of cycle structures cannot be generated.
In the following, we develop another algorithm for network structure generation. The contraction method (Algorithm A8) first constructs cycle structures which are then built in an acyclic network. Thus, any feasible number of cycle structures can be generated. On the other hand, the control of the restrictiveness must be limited to the generation of the acyclic skeleton of the isolated cycle structures in Step 1 and the construction of the aggregated network (including nodes corresponding to contracted cycle strucututres) in Step 4.

Algorithm A8. Generation of the structure of a cyclic network: Contraction method

Determine randomly the number t of backward arcs corresponding to maximal time lags which will be inserted into $\dot{G}$: $t \in \left\{ \left\lceil |E| MTL^{min} \right\rceil, \ldots, \left\lfloor |E| MTL^{max} \right\rfloor \right\}$.
Determine randomly the number CS of cycle structures which are to be generated in $\dot{G}$: $CS \in \left\{ CS^{min}, \ldots, \min\left\{ t, CS^{max} \right\} \right\}$.

(1) *Generation of CS acyclic digraphs*

Determine a random partitioning $X := \left\{ V_\nu \middle| \nu = 1, \ldots, CS \right\}$ of a subset V' of the set V of nodes ($|V'| \in \left\{ n_c^{min} CS, \ldots, n_c^{max} CS \right\}$) such that $|X| = CS$ and $|V_\nu| \in \left\{ n_c^{min}, \ldots, n_c^{max} \right\}$ $\forall \nu = 1, \ldots, CS$.
Construct CS acyclic digraphs $\dot{G}_\nu =< V_\nu, E_\nu >$ by performing Steps 1, 2, and 3 of Algorithm A4. Let R_ν be the set of sources and let S_ν be the set of sinks of digraph $\dot{G}_\nu$.

(2) *Transformation of the CS acyclic digraphs into strongly connected digraphs*

Compute the number t^{min} of arcs required for the transformation of any digraph $\dot{G}_\nu =< V_\nu, E_\nu >$ into a strongly connected digraph $\dot{G}'_\nu =< V_\nu, E'_\nu >$ ($\nu = 1, \ldots, CS$):
$t^{min} := \sum_{\nu=1}^{CS} \max\left\{ |R_\nu|, |S_\nu| \right\}$ (cf. Theorem 9).
$t := \max\left\{ t, t^{min} \right\}$. Determine a random vector $\boldsymbol{t} := (t_1, t_2, \ldots, t_{CS})$ with $t_\nu \geq \max\left\{ |R_\nu|, |S_\nu| \right\}$ $\forall \nu = 1, \ldots, \mathrm{CS}$ and $\sum_{\nu=1}^{CS} t_\nu = t$.

FOR $\nu = 1, \ldots, CS$ DO

$t_{\nu,d} := t_\nu - \max\left\{ |R_\nu|, |S_\nu| \right\}$

Transform $\dot{G}_\nu =< V_\nu, E_\nu >$ into a strongly connected digraph $\dot{G}'_\nu =< V_\nu, E'_\nu >$ by applying Algorithm A9.

Perform Step 4 of Algorithm A4 and Algorithm A5.

Insert $t_{\nu,d}$ further arcs by densifying $\dot{G}'_\nu$ using Step 3 of Algorithm A6. Instead of V, set P has to be initialized with $P := V_\nu$ and the formula for the determination of set T_i has to be replaced by $T_i := \left\{ j \in V_\nu \middle| a_{ji} = 0 \right\}$.

END (* FOR *).

(3) *Contraction of all cycle structures*

Set $C := \bigcup_{\nu=1,\ldots,CS} \dot{G}'_\nu$. Replace the nodes of all cycle structures $\dot{G}'_\nu$ by the corresponding contracted cycle structure c_ν: $V := V \setminus \bigcup_{\nu=1,\ldots,CS} V(\dot{G}'_\nu) \cup \bigcup_{\nu=1,\ldots,CS} \{c_\nu\}$.

(4) *Construction of an acyclic multidigraph*

Construct an acyclic digraph $\dot{G} = < V, E >$ based on the (new) node set V using algorithms A4 and A5. Since the contracted cycle structures actually consist of several nodes, an arc $< i, j >$ being incident with a node c_ν can be in parallel $\left|V(\dot{G}'_\nu)\right|$ times.

(5) *Expansion of cycle structures*

FOR $\nu = 1, \ldots, CS$ DO
    Set $\bar{E}_\nu := \{< i, j > \in E \mid i = c_\nu \vee j = c_\nu\}$.
    Expand cycle structure $\dot{G}'_\nu$ by replacing node c_ν in the node set V of $\dot{G}$ by $V(\dot{G}'_\nu)$.
    Update E: $E := E \setminus \bar{E}_\nu \cup E'_\nu$.
    Determine a random assignment of arcs $< i, j > \in \bar{E}_\nu$ to nodes $i \in V(\dot{G}'_\nu)$ or $j \in V(\dot{G}'_\nu)$, respectively:
    WHILE $\bar{E}_\nu \neq \emptyset$ DO
        Select an arc $< i, j > \in \bar{E}_\nu$.
        $\bar{E}_\nu := \bar{E}_\nu \setminus \{< i, j >\}$.
        IF $i = c_\nu$ THEN
            Select randomly a node $k \in V(\dot{G}'_\nu)$.
            Insert arc $< k, j >$ into $\dot{G}$: $E := E \cup \{< k, j >\}$.
            $a_{kj} := 1$.
        ELSE
            Select randomly a node $l \in V(\dot{G}'_\nu)$.
            Insert arc $< i, l >$ into $\dot{G}$: $E := E \cup \{< i, l >\}$.
            $a_{il} := 1$.
        END (* IF *).
    END (* WHILE *).
END (* FOR *).

(6) *Weakly connection*

Introduce supersource 0 and supersink n+1 applying Algorithm A7. ❑

The following algorithm which is used in Step 2 of the contraction method transforms a digraph into a strong component inserting a minimal number of additional arcs.

Algorithm A9. Generation of a strong component $\vec{G}'=<V,E'>$ based on a given subdigraph $\vec{G}=<V,E>$

(1) *Generation of an acyclic weak component*

$\vec{G}':=\vec{G}$.
Compute the set C of all cycle structures of $\vec{G}'$ with the algorithm of Even: $C:=\{C(i)|i\in V\}$.
Contract all cycle structures $C(i)\in C$.
Let R be the set of sources of $\vec{G}'$ and S be the set of sinks of $\vec{G}'$.
Let SC and SG be empty stacks.
Let $\{\vec{G}_1,\vec{G}_2,...,\vec{G}_{WC}\}$ be the set of weak components of $\vec{G}'$.
FOR $\nu=1,...,WC-1$ DO
    Let s be a sink of $\vec{G}_\nu$ and let r be a source of $\vec{G}_{\nu+1}$.
    Insert arc $<s,r>$.
    $R:=R\setminus\{r\},S:=S\setminus\{s\}$.
END (* FOR *).

(2) *Elimination of sources and sinks*

WHILE $|S|>0$ and $R\cap S=\emptyset$ DO
    IF $|R|=1$
        Select an arbitrary sink $s\in S$ and the unique source $r\in R$.
        Insert arc $<s,r>$.
        $R:=R\setminus\{r\}\cup\{C(r)\},S:=S\setminus\{s\}$.
        IF $|S|=0$
            $S:=S\cup\{C(r)\}$.
        END (*IF*).
    ELSIF $|S|=1$
        Select the unique sink $s\in S$ and an arbitrary source $r\in R$.
        Insert arc $<s,r>$.
        $R:=R\setminus\{r\},S:=S\setminus\{s\}\cup\{C(r)\}$.
    ELSE
        Select a sink $s\in S$ with $|\overline{R}(s)\cap R|\geq 2$.
        Determine a source $r\in R$ with $s\in R(r)$ and $|R(r)\cap S|\geq 2$.
        Insert arc $<s,r>$.
        $R:=R\setminus\{r\},S:=S\setminus\{s\}$.
    END (* IF *).
    $SC:=SC\cup C(r)$. $SG:=SG\cup\{\vec{G}'\}$.
    Contract cycle structure $C(r)$.
END (* WHILE *).

(3) *Expansion of the cycle structures*

WHILE $SC \neq \emptyset$ DO

$C(r)$:= Head(SC).

$\dot{G}''$:= Head(SG).

Expand $C(r)$ in $\dot{G}'$ with respect to $\dot{G}''$ in analogy to step 5 of Algorithm A8.

$SC := SC \setminus \{C(r)\}$.

$SG := SG \setminus \{\dot{G}''\}$.

END (* WHILE *). ❑

Remark 11.

Applying Algorithm A9 in the contraction method, the algorithm of Even can be skipped since $\dot{G}$ is acyclic.

Theorem 9.

(a) Algorithm A9 is correct (that is, it generates a strong component $\dot{G}'$ with subdigraph $\dot{G}$ within a finite number of steps).

(b) Among all strong components with subdigraph $\dot{G}$ the generated strongly connected digraph $\dot{G}'$ contains the minimum number of arcs.

Proof.

(a) The contraction of the cycle structures in Step 1 yields an acyclic digraph $\dot{G}'$ consisting of WC weak components, each having at least one source and one sink (cf. Theorem 7). The insertion of WC–1 arcs which weakly connect subdigraphs $\dot{G}_\nu$ and $\dot{G}_{\nu+1}$ $(\nu = 1,\ldots,WC-1)$ obviously makes digraph $\dot{G}'$ a weak component. Like any acyclic digraph, $\dot{G}'$ has at least one source and one sink.
If $R \cap S \neq \emptyset$, $\dot{G}'$ consists of only one node and we skip to Step 3.
Since $s \in R\,(s)$ for each of the three cases, we generate a new cycle structure in every pass of Step 2 which is contracted in $\dot{G}'$, that is, each digraph $\dot{G}'$ obtained at the end of Step 2 is acyclic and weakly connected, too. Due to Theorem 5, there can always be selected a source $r \in R$ and a sink $s \in S$ satisfying the conditions specified in Step 2.
At any pass of Step 2, the number of nodes is reduced by at least one. The algorithm passes to Step 3, if there remains only one sink which represents a source at the same time. This is, due to the weak connectivity of all intermediate digraphs $\dot{G}'$, true exactly if there remains only one node. Since V is finite, this will be achieved after a finite number of passes of Step 2.
The successive expansion of contracted cycle structures in Step 3 finally yields a strong component which constitutes a subdigraph of the underlying digraph $\dot{G}$.

(b) The contraction of all cycle structures of digraph $\dot{G}$ leads to an acyclic digraph $\dot{G}'$ with set of sources R and set of sinks S. The number of arcs required for the transformation of $\dot{G}'$ into a strong component will be the same than the number of arcs required for the transformation of $\dot{G}$ into a strong component.
The insertion of an arc $<i,j> (i,j \in V)$ can at most eliminate one source and one sink. Obviously, in the node set of a strong component there are neither sources nor sinks. Therefore, the minimum number of arcs which are required to strongly connect $\dot{G}$ is $\max\{|R|,|S|\}$.
In Step 1, at each insertion of an arc $<s,r>$ exactly one source and one sink are eliminated without creating a new cycle structure in $\dot{G}'$ ($s \notin R(r)$, since r and s are nodes which initially belong to two different weak components of $\dot{G}'$).
Considering Step 2, we have to distinguish between four cases.

(i) $|R| = 1, |S| \geq 2$

By the insertion of $<s,r>$ and the subsequent contraction of $C(r)$ to node c, we eliminate source r and sink s, obtaining an additional source c.

(ii) $|R| \geq 2, |S| = 1$

By the insertion of $<s,r>$ and the subsequent contraction of $C(r)$ to node c, we eliminate source r and sink s, obtaining an additional sink c.

(iii) $|R| \geq 2, |S| \geq 2$

Applying Theorem 6, we obtain that the insertion of $<s,r>$ and the subsequent contraction of $C(r)$ to node c reduces both the number of sources and the number of sinks by one.

(iv) $\quad |R| = 1, |S| = 1$

By the insertion of $<s,r>$ and the subsequent contraction of $C(r)$ to node c, we eliminate source r and sink s, obtaining an additional source c which constitutes a sink at the same time, that is, $|R| = 1, |S| = 1$ and $R \cap S \neq \emptyset$. $\dot{G}'$ consists of the single node c.

All in all, the algorithm inserts

$$\underbrace{WC-1}_{\text{Step 1}} + \underbrace{\underbrace{\big||R|-|S|\big| + \min\{|R|,|S|\}}_{\text{Case 1, 2}} - \underbrace{(WC-1)-1}_{\text{Case 3}} + \underbrace{1}_{\text{Case 4}}}_{\text{Step 2}} = \max\{|R|,|S|\}$$

arcs into $\dot{G}$, which represents, as seen above, the minimum number required to achieve the strong connectivity of $\dot{G}$. ❏

Remark 12. Time complexity of Algorithm A9

The time complexity of Step 1 of Algorithm A9 is $O(|E|)$, since $O(|E|)$ represents the time complexity of the algorithm of Even, and each arc $<i,j> \in E$ is at most concerned by one contraction.
Let R be the set of sources and S be the set of sinks in the acyclic digraph $\dot{G}'$ obtained at the end of Step 1. Then, the time complexity of Step 2 is $O(\min\{|R|,|S|\}(|R|+|S|))$.
The time complexity of Step 3 is $O(|E|)$, since each arc $<i,j> \in E$ is concerned by at most one expansion.
Hence, the time complexity of Algorithm A9 is $O(|E| + \min\{|R|,|S|\}(|R|+|S|))$.

3.4 Arc weights

Let $\dot{G} = <V,E>$ be the weakly connected digraph generated in Subsections 3.2 and 3.3. (Forward) arcs generated in Subsection 3.2 belong to minimal time lags, (backward) arcs generated in Subsection 3.3 belong to maximal time lags. For the representation of precedence and time constraints by the project network $\dot{N} = <V,E;c>$ we have to determine a weight c_{ij} for any arc $<i,j> \in E$.
In this subsection we generate activity durations D_{jm} $(j \in V, m \in M_j)$. Based on these durations, minimal and maximal time lags (arc weights c_{ij}) are calculated for $<i,j> \in E$.

The following input data has to be specified by the user of ProGen / max:

D^{min}, D^{max}	integer-valued minimal and maximal duration of an activity
$\varepsilon_d \in [0,\infty)$	maximal relative deviation of minimal time lags from activity durations
$CST \in [0,1]$	cycle structure tightness
$SF \in [0,\infty)$	slack factor
$PDT \in [0,1]$	project duration tightness

$c^{\rho,min}, c^{\rho,max}$	minimal and maximal costs for the period availability of one unit of a renewable or doubly-constrained resources
$c^{\nu,min}, c^{\nu,max}$	minimal and maximal costs for the total availability of one unit of a non-renewable or doubly-constrained resources

Activity durations D_{jm} $(j \in V \setminus \{0, n+1\}, m \in M_j)$ are drawn randomly out of the set $\{D^{min}, \dots, D^{max}\}$, each element having the same probability to be selected.

In contrast to the problem generator of Kolisch et al. (1992), the minimal time lags between two nodes (or corresponding activities, respectively) may be different from the duration of the first activity. This way, overlappings and waiting times between activities can be modeled. We introduce an index $\varepsilon_d \in [0, \infty)$ such that the minimal time lag T_{ijm}^{min} between the start of activity i and the start of activity j depending on the execution mode $m \in M_i$ of activity i is in the interval $[(1-\varepsilon_d)D_{im}, (1+\varepsilon_d)D_{im}]$.

The maximal time lag T_{ij}^{max} between the start of activity i and the start of activity j is determined randomly in interval (cf. Figure 3)

(3.1) $\left[a(i,j) + (b(i,j) - a(i,j))CST^2, [a(i,j) + 2(b(i,j) - a(i,j))CST - (b(i,j) - a(i,j))CST^2](1+SF)\right].$

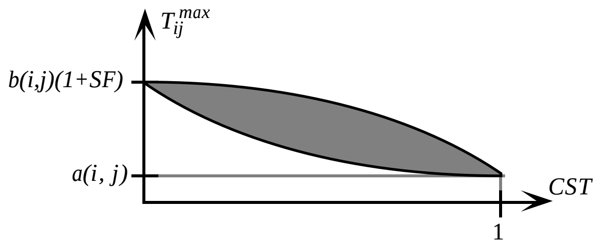

Fig. 3. Interval for maximal time lags depending on the cycle structure tightness *CST*

$a(i,j)$ represents the minimum time lag between the start of activity i and the start of activity j induced by the precedence constraints. $b(i,j)$ corresponds to a maximal time lag which always can be met (observing precedence and resource constraints).

Let $\dot{G}' = \langle V, E'; c' \rangle$ be the weakly connected acyclic network with $c'_{kl} := \max_{m \in M_k} \{T_{klm}^{min}\}$ $(\langle k,l \rangle \in E')$, E' being the subset of E which contains all arcs corresponding to minimal time lags. $S'(i)$ denotes the set of direct successors of node i in $\dot{G}'$. $a(i,j)$ and $b(i,j)$ can be computed as follows:

(3.2) $a(i,j) := L_{G'}^{r}(i,j)$:= length of a longest (directed) path from i to j in $\dot{G}'$.

(3.3) $$b(i,j) := \sum_{k \in R(i) \cap \overline{R}(j) \setminus \{j\}} \max_{m \in M_k} \max\left\{ D_{km}, \max_{l \in S'(k) \cap \overline{R}(j)} T_{klm}^{min} \right\}.$$

Note that $R(i)$ and $\overline{R}(j)$ denote the corresponding sets in (the cyclic network) G'.
Let $N' := < V, E, c >$ be the resulting project network with arc weights corresponding to minimal time lags and negative maximal time lags.

If the problem instances to be generated are of type MRLP / max or MRIP / max, we have to determine an appropriate value for the project duration T. A lower bound T^{min} on the project duration is given by $T^{min} := L_{\vec{G}'}(0, n+1)$.
For each cycle structure $C \in \mathcal{C}$ of N' we determine earliest start times EST_j and latest finish times LFT_j of activities $j \in V(C)$. An upper bound T^{max} on the project duration is then given by $T^{max} := \sum_{C \in \mathcal{C}} \max_{j \in V(C)} LFT_j + \sum_{j \in V \setminus \bigcup_{C \in \mathcal{C}} V(C)} \max_{m \in M_j} T_{ijm}^{min}$.

The project duration T is determined as follows: $T := T^{min} + \text{int}(PDT(T^{max} - T^{min}))$.

4. Resource Demand and Availability Generation

The third kind of restrictions besides precedence and time constraints given by the project network are limitations due to scarce resources. The relation of resource demand and resource availability strongly influences, apart from the case of a series project network, the set of feasible subsets (cf. Mingozzi at al. 1994) where a feasible subset F is defined to be a subset of activity set V such that all activities of F can be executed at the same time taking precedence, time, and resource constraints into account (evidently, F depends on the modes in which the activities are performed).
Numerous resource characteristics for resource-constrained scheduling problems can be found in literature, for example in Kurtulus and Davis (1975), Patterson (1976), Davis (1982), Kurtulus and Narula (1985), and Kolisch et al. (1992).
Generalizing and normalizing measures which have been used before, Kolisch et al. (1992) developed a new set of control parameters which have a strong impact on the hardness of problem instances. ProGen / max employs the same set of resource measures for the problem generation. In the following, we briefly sketch the generation of resource requirements and resource availbaility proposed by Kolisch et al.

4.1 Resource demand generation

The processing of an activity consumes or uses a certain amount of one or several nonrenewable, renewable or doubly-constrained resources, respectively. After the generation of a certain number of resources, the generation of resource consumption and resource usage is performed in two steps: First, for any given activity-mode combination (j,m) with $j \in V, m \in M_j$ we have to determine a set R_{jm} of resources required for the processing of activity j in mode m. Then, for all resources $i \in R_{jm}$ we fix the (integer-valued) number of units which will be consumed or used for the processing of activity j in mode m $(j \in V, m \in M_j)$.

The generalized resource factor RF for multi-mode problems which has been introduced by Kolisch et al. (1992) indicates the mean percentage of resources which are affected by the execution of an activity:

(4.1) $$RF_\rho := \frac{1}{|V|-2} \frac{1}{|R^\rho \setminus R^\nu|} \sum_{j \in V \setminus \{0,n+1\}} \frac{1}{|M_j|} \sum_{m \in M_j} \sum_{i \in R^\rho \setminus R^\nu} \delta(r^\rho_{ijm}) \text{ for renewable resources,}$$

$$RF_\nu := \frac{1}{|V|-2} \frac{1}{|R^\nu \setminus R^\rho|} \sum_{j \in V \setminus \{0,n+1\}} \frac{1}{|M_j|} \sum_{m \in M_j} \sum_{i \in R^\nu \setminus R^\rho} \delta(r^\nu_{ijm}) \text{ for nonrenewable resources,}$$

$$RF_\delta := \frac{1}{|V|-2} \frac{1}{|R^\rho \cap R^\nu|} \sum_{j \in V \setminus \{0,n+1\}} \frac{1}{|M_j|} \sum_{m \in M_j} \sum_{i \in R^\rho \cap R^\nu} \delta(r^\rho_{ijm}) \text{ for doubly-constrained resources}$$

with $$\delta(x) := \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{otherwise} \end{cases}.$$

ProGen / max employs the following input data for the resource demand generation:

$Q_\tau^{min}, Q_\tau^{max}$ $(\tau \in \{\rho, \nu, \delta\})$	minimal and maximal number of renewable, nonrenewable, and doubly-constrained resources used by an activity ("request")
$RF_\tau^{min}, RF_\tau^{max}$ $(\tau \in \{\rho, \nu, \delta\})$	minimal and maximal resource factor of renewable, nonrenewable, and doubly-constrained resources
$U_\tau^{min}, U_\tau^{max}$ $(\tau \in \{\rho, \nu\})$	minimal and maximal number of units of renewable, nonrenewable, and doubly-constrained resources required for the processing of an activity ("level of demand")

For the algorithmic generation of the three-dimensional resource-activity-mode-array $RQ := \left(\delta(r_{ijm}^\tau)\right)_{i \in R^\rho \cup R^\nu, j \in V, m \in M_j}$ based on $Q_\tau^{min}, Q_\tau^{max}$ and $RF_\tau^{min}, RF_\tau^{max}$ $(\tau \in \{\rho, \nu, \delta\})$ we refer to Kolisch et al. (1992).
In the second step, we assign a demand level to any triplet (i, j, m) with $\delta(r_{ijm}^\tau) = 1$: $r_{ijm}^\tau := rand\{U_\tau^{min}, \ldots, U_\tau^{max}\}$ $(i \in R^\rho \cup R^\nu, j \in V, m \in M_j, \delta(r_{ijm}^\tau) = 1)$. In contrast to ProGen, for a given activity j and a given resource i the resource requirements r_{ijm}^τ can vary with modes $m \in M_j$, if the respective option (mode-varying resource demand levels) has been selected. The generation of inefficient modes in the second step (that is, modes $\overline{m} \in M_j$: $(\exists\, m \in M_j : D_{jm} \le D_{j\overline{m}}, r_{ijm}^\rho \le r_{ij\overline{m}}^\rho \;\forall i \in R^\rho, r_{ijm}^\nu \le r_{ij\overline{m}}^\nu \;\forall i \in R^\nu))$ may necessitate several passes of the algorithm.

4.2 Resource Availability Generation

Let $R_{i,\rho}^{min} := \max_{j \in V} \min_{m \in M_j} r_{ijm}^\rho$ be the minimal availability of renewable or doubly-constrained resource $i \in R^\rho$ required to perform all activities of the project and let $R_{i,\nu}^{min} := \sum_{j \in V} \min_{m \in M_j} r_{ijm}^\nu$ be the minimal availability of nonrenewable or doubly-constrained resource $i \in R^\nu$ required to perform all activities of the project.
In case of resources $i \in R^\nu$, an upper bound on the maximal availability required to perform all activities of the project can be calculated as follows: $R_{i,\nu}^{max} := \sum_{j \in V} \max_{m \in M_j} r_{ijm}^\nu$. For resources $i \in R^\rho$ we perform a resource-unconstrained temporal analysis in network $G'(i) := \langle V, E', c'(i) \rangle$ with E' being the subset of the project network arc set E including all (forward) arcs which correspond to minimal time lags. Arcs $\langle j, l \rangle \in E'$ are weighted with $c'_{jl}(i) := \min\left\{ T_{jlm_{ij}^*}^{min} \,\middle|\, m_{ij}^* = \arg\max_{m \in M_j} r_{ijm}^\rho \right\}$. Let $V_i(t) := \left\{ j \in V \,\middle|\, t - D_{jm_{ij}^*} < EST_j \le t \right\}$ be the set of activities in progress at time t based on earliest start times EST_j $(j \in V)$ determined by the

temporal analysis in $\dot{G}'(i)$. Then, $R_{i,\rho}^{max} := \max_{t=0,...,\overline{T}-1} \sum_{j \in V(t)} r_{ijm_{ij}^*}^{\rho}$ represents an upper bound on the maximal availability of a resource $i \in R^{\rho}$ required to perform all activities of the project.

The [0,1]-normalized resource strength $RS_{i,\tau}$ of resource i introduced by Kolisch et al. (1992) is defined as follows:

$$(4.2) \quad RS_{i,\tau} := \frac{R_i^{\tau} - R_{i,\tau}^{min}}{R_{i,\tau}^{max} - R_{i,\tau}^{min}} \; (i \in R^{\tau}, \tau \in \{\rho, \nu\}).$$

The following input data is required for the generation of resource availability:

$RS_{\tau}^{min}, RS_{\tau}^{max}$ $(\tau \in \{\rho, \nu\})$	minimal and maximal resource strength of renewable, nonrenewable, and doubly-constrained resources.

The resource strength is randomly determined as follows: $RS_{i,\tau} := rand[RS_{\tau}^{min}, RS_{\tau}^{max}]$. Hence, in opposite to ProGen, the resource strength of resources belonging to the same type $\tau \in \{\rho, \nu\}$ may be different if $RS_{\tau}^{min} < RS_{\tau}^{max}$.
With $R_i^{\tau} := R_{i,\tau}^{min} + \mathrm{int}(RS_{i,\tau}(R_{i,\tau}^{max} - R_{i,\tau}^{min}))\,(i \in R^{\tau}, \tau \in \{\rho, \nu\})$ we obtain the availability of renewable, nonrenewable, and doubly-constrained resources.

5. Conclusions

ProGen / max, a problem generator for the resource-constrained minimum project duration problem MRCPSP / max, the resource leveling problem MRLP / max, and the resource investment problem MRIP / max has been developed. The emphasis was put on the efficient and parameter-driven generation of cyclic network structures which occur if maximal time lags have to be taken into consideration.
Two different approaches for the network generation have been proposed: the direct and the contraction method. These will be tested competitively with respect to computation time and the ability to generate special-shaped digraphs such as networks with a high percentage of maximal time lags and a large number of cycle structures.
The impact of network restrictiveness introduced by Thesen on the hardness of problem instances in resource-constrained scheduling will be evaluated in comparison with other measures of network complexity.
A large testset of problem instances will be generated allowing full factorial design evaluation of exact and heuristic procedures. Thus, the specific suitability of algorithms depending on values for network and resource measures will be investigated.

References

Agrawal, M., Elmaghraby, S., Herroelen, W. (1994): DAGEN: A Generator of Testsets for Project Activity Nets; *Working Paper*, North Carolina State University, USA

Bein, W., Kamburowski, J., Stallmann, M. (1992): Optimal reduction of two-terminal directed acyclic graphs; *SIAM J. Comput. 21*, 1112 – 1129

Berge, C. (1985): *Graphs*; 2nd rev. ed., North-Holland, Amsterdam

Bondy, J.A., Murty, U.S.R. (1976): Graph Theory with Applications; The MacMillan Press, London

Davis, E.W. (1975): Project Network Summary Measures Constrained-Resource Scheduling; *AIIE Trans. 7*, 132 – 142

Demeulemeester, E.L. (1992): Optimal Algorithms for Various Classes of Multiple Resource-Constrained Project Scheduling Problems; *Ph.D. Thesis*, University of Leuven

Demeulemeester, E.L., Dodin, B., Herroelen, W.S. (1993): A random activity network generator; *Oper. Res. 41*, 972 – 980

Elmaghraby, E., Herroelen, W.S. (1980): On the measurement of complexity in activity networks; *EJOR 5*, 223 – 234

Even, S. (1979): *Graph Algorithms*. Pitman, London

Franck, B., Schwindt, C. (1995): Different Resource-Constrained Project Scheduling Problems - Models and Practical Applications; *Report WIOR-450*, Institut für Wirtschaftstheorie und Operations Research, University of Karlsruhe

Kaimann, R.A. (1974): Coefficient of network complexity; *Mgmt Sci. 21*, 172 – 177

Kolisch, R., Sprecher, A., Drexl, A. (1992): Characterization and generation of a general class of resource-constrained project scheduling problems: Easy and hard instances; Manuskript aus den Instituten für Betriebswirtschaftslehre der Universität Kiel Nr. 301, University of Kiel

Kurtulus, I., Davis, E.W. (1982): Multi-project scheduling: Categorization of heuristic rules performance; *Mgmt Sci. 28*, 161 – 172

Kurtulus, I.S., Narula, S. (1985): Multi-project scheduling: Analysis of project performance; *IIE Trans. 17*, 58 – 65

Mingozzi, A., Maniezzo, V., Ricciardelli, S., Bianco, L. (1994): An Exact Algorithm for Project Scheduling with Resource Constraints Based on a New Mathematical Formulation; *Working Paper*, University of Bologna, Italy

Neumann, K., Morlock, M. (1993): *Operations Research*; Hanser, München

Neumann, K., Schwindt, C. (1995): Projects with Minimal and Maximal Time Lags: Construction of Activity-on-Node Networks and Applications; *Report WIOR-447*, Institut für Wirtschaftstheorie und Operations Research, University of Karlsruhe

Patterson, J.H. (1976): Project scheduling: The effects of problem structure on heuristic performance; *Nav. Res. Log. Quart. 23*; 95 – 123

Patterson, J.H. (1984): A comparison of exact approaches for solving the multiple constrained resource, project scheduling problem; *Mgmt Sci. 30*, 854 – 867

de Reyck, B., Herroelen, W. (1994): On the use of the Complexity index as a Measure of complexity in activity Networks; *Working Paper*, University of Leuven, Belgium

Schrage, L. (1979): A more portable FORTRAN random number generator; *ACM Transact. Math. Software 5*, 132 – 138

Thesen, A. (1977): Measures of the restrictiveness of project networks; *Networks 7*, 193 – 208

DISCUSSION PAPERS DES INSTITUTS FÜR WIRTSCHAFTSTHEORIE UND OPERATIONS RESEARCH

275. **Neumann, K.:** Einführung in das Operations Research I. Korrigierter Nachdruck. *Dezember 1986.*

276. **Neumann, K.:** Einführung in das Operations Research I I. *Juli 1986.*

277. **Hofmann, H. und A. Lamatsch:** Programmsystem Projektplanung - Vorstufe eines Expertensystems. *Juli 1986.*

278. **Pfingsten, A.:** New Concepts of Lorenz Domination and Risk Aversion. *März 1986, erscheint in:* Methods of Operations Research.

279. **Pfingsten, A.:** Generalized Concepts of Tax Progression and Inequality Reduction. *März 1986.*

280. **Neumann, K.:** Stochastic Project Networks I. *Mai 1986.*

281. **Neumann, K.:** Stochastic Project Networks I I. *Juli 1986.*

282. **Lamatsch, A.:** Einsatz des Savingverfahrens zur Wagenumlaufplanung im öffentlichen Personennahverkehr. *April 1986.*

283. **Buhl, H. U.:** Besprechung zu " Stochastische Optimierung bei partieller Information", Autor: P. Abel; *erschienen in:* Journal of Economics / Zeitschrift für Nationalökonomie *1986.*

284. **Buhl, H. U.:** Generalization and Applications of a Class of Dynamic Programming Problems, EJOR 31 (1987).

285. **Bossert, W. und A. Pfingsten:** The Circular and Time Reversal Tests Reconsidered in Economic Price Index Theory. *1986*; *erschienen in:* Statistical Papers 28 (1987), 271-284.

286. **Neumann, K. und A. Lamatsch:** Mehrgüterflüsse in Graphen zur Beschreibung des Verkehrsablaufs auf einer Strecke bei Verkehrsmischung. *Oktober 1986.*

287. **Pfingsten, A.:** Progressive Taxation and Redistributive Taxation: Different Labels for the Same Product? *Erschienen in:* Social Choice and Welfare 5, 1988, 235-246, *und in* Gaertner, W. und P. K. Pattanaik (Hrsg.), Distributive Justice and Inequality, Springer- Verlag, 1988, 147-158.

288. **Neumann, K.:** Klausuraufgaben OR I + I I mit Lösungen. Neuauflage *Mai 1989.*

289. **Bossert, W.:** A Note on Intermediate Inequality Indices which are Quasilinear Means. *1986.*

290. **Buhl, H. U.:** Theorie und Anwendungen zur Optimierung von Verschrottungserscheinungen; *erschienen in:* Isermann et al. (Hrsg.): Tagungsband der 15. DGOR Jahrestagung. OR- Proceeding of the 15. DGOR Conference 1986. Springer- Verlag, Berlin 1987.

291. **Buhl, H. U. und A. Pfingsten:** On the Distribution of Public Funds.

292. **Fuchs-Seliger, S.:** Applications of Income Compensation Functions to Social Welfare Theory.

293. Buhl, H. U.: Optimization of Scrapping Decisions: Theory and Applications; *erschienen in:* ZOR B, 1988; *erscheint in:* Radermacher, F. J. (Hrsg.), Methods of Operations Research, Athenäum- Hein, Meisenheim 1989.

294. Lamatsch, A., M. Morlock, K. Neumann und Th. Rubach: SCHEDULE - An Expert System for Scheduling. *Oktober 1986.*

295. Egle, K. und S. Fenyi: Zweistufige Disaggregation und baryzentrisches Kalkül.

296. Eichhorn, W.: On a Class of Inequality Measures; *erschienen in:* Social Choice and Welfare, 5 (1988), 171-177.

297. Lamatsch, A.: Progammbibliothek Operations Research für die Rechner HP 1000 - Siemens 7881. *Februar 1987.*

298. Buhl, H. U.: Optimale Verschrottungsentscheidungen in der Lagerhaltung; *erschienen in:* Schmalenbachs Zeitschrift für betriebswirtschaftliche Forschung, 5 (1987).

299. Buhl, H. U. und W. Bossert: More on Sufficiency Conditions for Interior Location in the Triangle Space. 1987; *erschienen in:* Journal of Regional Science (1988).

300. Buhl, H. U.: Ein finanzwissenschaftlich-methodischer Diskussionsbeitrag zur Neuordnung des Länderfinanzausgleichs in der Bundesrepublik Deutschland; *erschienen in:* Finanzarchiv, N. F. 44 (1986)3.

301. Fuchs-Seliger, S.: A note on non-inferior demand functions.

302. Eichhorn, W. und F. Stehling: Eine Bemerkung zur Verteilungsneutralität der produktivitätsorientierten Lohnpolitik. *Dezember 1986*; *erschienen in:* O. Opitz, R. Rauhut (Hrsg.): Ökonomie und Mathematik. Springer- Verlag, Berlin- Heidelberg 1987, 523-532.

303. Pfingsten, A.: Incentives to Forecast Honestly; *erschienen in:* Agency Theory, Information, and Incemtires. G. Bamberg, K. Spremann et al. (eds.). Springer- Verlag, Heidelberg 1987, 117-134.

304. Bossert, W. und A. Pfingsten: Relations between the true cost of living index and statistical price index numbers: A survey and some extensions, *1986.*

305. Fuchs-Seliger, S.: Money- Metric Utility Functions in the Theory of Revealed Preference; *erscheint in:* Mathematical Social Sciences.

306. Eichhorn, W. und M. Hellwig: Versicherungsmärkte: Theorie A. Versicherungsmärkte mit vollständiger Information; *erschienen in:* Handwörterbuch der Versicherung, HdV, herausgegeben von D. Farny, E. Helten, P. Koch und R. Schmidt, Verlag Versicherungswirtschaft, Karlsruhe 1988, 1055-1064.

307. Eichhorn, W. (editor in coorporation with W. E. Diewert, S. Fuchs- Seliger, W. Gehrig, A. Pfingsten, K. Spremann, F. Stehling and J. Voeller): Measurement in Economics. Theory and Applications of Economic Indices. Physica- Verlag, Heidelberg 1988, 830 Seiten.

308. Eichhorn, W.: Mikroelektronik - Wurzel der dritten industriellen Revolution; *erschienen in:* WÜBA- Gazette, Jubiläumsausgabe WÜBA, 150 Jahre Dienst am Kunden, 1837-1987. Heilbronn 1987, 47-51.

309. Fuchs-Seliger, S.: Measuring by Money- Metric Utility Functions.

310. Pfingsten, A.: Scaling Income Distributions.

311. Bossert, W.: The Impossibility of an Intermediate Solution in a One-dimensional Location Model: A General Result. *1987*; *erscheint in:* Regional Science and Urban Economics.

312. Weinhardt, C.: On inequality measurement when population sizes differ; *erschienen in:* Henn et al. (Hrsg.): Methods of Operations Research, 59, Athenäum Verlag (1989), S.111-124

313. Eichhorn, W. und Gleissner, W.: The Equation of Measurement; *erschienen in:* vgl. Discussion Paper No. 307, 19-27.

314. Eichhorn, W. und Gleissner W.: The Solution of Important Special Cases of the Equation of Measurement; *erschienen in:* vgl. Discussion Paper No. 307, 29-37.

315. Bossert, W. und A. Pfingsten: Intermediate Inequality: Concepts, Indices, and Welfare Implications. *1987, erschienen in:* Mathematical Social Sciences.

316. Fuchs-Seliger, S.: Non-inferior Demand Functions Revisted.

317. Buhl, H. U. und A. Pfingsten: 10 Gebote für Finanzausgleichsverfahren und deren Implikationen. *September 1987.*

318. Buhl, H. U.: Die " Zitronen- Kette". *Oktober 1987.*

319. Buhl, H. U.: Axiomatic Considerations in Multi- Objective Location Theory. *Oktober 1987*; *erschienen in:* European Journal of Operations Research, 1988.

320. Fuchs-Seliger, S. und M. Krtscha: An Alternative Approach to Joint Continuity in Economics.

321. Wenzelburger, D.: Vergleich verschiedener umweltpolitischer Instrumentarien anhand eines Optimiermodells.

322. Neumann, K.: Die Netzwerk- Simplexmethode zur Lösung des Umlade- und des Transportproblems. *Dezember 1987.*

323. Neumann, K.: Kürzeste Wege und Matchings. *Januar 1988.*

324. Buhl, H. U.: Besprechung zu: Anders/ Borglin " Optimality in Infinite Horizon Models"; *erschienen in:* Journal of Institutional and Theoretical Economics / Zeitschrift für die gesamte Staatswissenschaft, 1988.

325. Buhl, H. U.: Dauerarbeitslosigkeit: Wachstumstheoretische und Verteilungstheoretische Aspekte.

326. Eichhorn, W.: Risiko und Versicherung; *erschienen in:* Das Risiko und seine Akzeptanz, Hoechst-Gespräch 1988, herausgegeben von der Hoechst AG, Schütze Verlag, Bonn-Frankfurt 1989, 95-120.

327. Morlock, M.: Wissensbasierte Systeme im ORI (Heuristiken). *März 1988.*

328. Pfingsten, A.: Empirical Investigation of Inquality Concepts: A Method and First Results.

329. Neumann, K.: Das Briefträgerproblem in Graphen, Digraphen und gemischten Graphen. *März 1988.*

330. Neumann, K.: Handlungsreisendenproblem und Tourenplanung. *Juni 1988.*

331. Egle K. und S. Fenyi: Lösung des D/W Input- Output- Systems durch stochastische Inversion.

332. Pfingsten, A.: Surplus Sharing Methods.

333. Schnelle, M.: Kundenzeitschranken in der Tourenplanung. *September 1988*.

334. Zhan, J.: Kalendrierung der Terminplanung in MPM-Netzplänen. *Januar 1988*.

335. Ott, V. und J. Weber: Entscheidungsmethoden für Umwelttechnik.

336. Eichhorn, W.: Unabhängigkeit der Shephardschen Axiome; *erschienen in:* Statistik, Informatik und Ökonomie, W. Janko (Hrsg.), Springer-Verlag, Berlin-Heidelberg 1988, 49-54.

337. Fuchs-Seliger, S.: An Axiomatic Approach to Compensated Demand; *erscheint in:* Journal of Economic Theory.

338. Bossert, W.: Social Evaluation with Variable Population Size: An Alternative Concept.

339. Bossert, W.: On the Extension of Preferences over a Set to the Power Set: An Axiomatic Characterization of a Quasi- Ordering (Revised Version); *erscheint in:* Journal of Economic Theory.

340. Eichhorn, W. und U. Leopold: Logical Aspects Concerning Shephard's Axioms of Production Theory.

341. Bossert, W.: Rawlsian Welfare Orderings with Variable Population Size; *erscheint als:* " Maximin Welfare Orderings with Variable Population Size" in Social Choice and Welfare.

342. Fuchs-Seliger, S.: On the Continuity of Income Compensation Functions.

343. Bossert, W.: Generalized Gini Social Evaluation Functions and Low Income Group Aggregation; *erscheint als:* " An Axiomatization of the single-series Ginis" *in* Journal of Economic Theory.

344. Weinhardt, C.: Currency-Independence of Inequality Measures; *erschienen in:* Henn et al. (Hrsg.): Methods of Operations Research, 60, Anton Hain Verlag, (1990), S.525-536.

345. Eichhorn, W.: Vom magischen Viereck zum ökolomagischen Neuneck.

346. Buhl, H. U.: Eine Finanzanalyse des Hersteller- Leasings; *erschienen in*: Zeitschrift für Betriebswirtschaft, 4, 1989.

347. Bossert, W.: The Location of a Monopolistic Firm.

348. Bossert, W.: Population Replications and Ethical Poverty Measurement.

349. Fuchs- Seliger, S.: Non- Saturated Preferences and Compensated Demand.

350. Pfingsten, A.: Der Einsatz von monetären Anreizsystemen in der Planung.

351. Buhl, H. U.: Ein Finanzierungs- Expertensystem zur Unterstützung der Unternehmens-Strategischen Vorteile des Hersteller-Leasings; *erschienen in:* Spremann, K. (Hrsg.): Informationstechnologie und strategische Führung, Gabler, Wiesbaden 1989.

352. Pfingsten, A.: Sparten gut, alles gut? - Zur Notwendigkeit ergänzender zentraler Steuerung; *erschienen in:* Die Bank, 1989, 139-141.

353. Bossert, W. und A. Pfingsten: Nonhomothetic Preferences and Exact Price Index Numbers.

354. **Hofmann, H., A. Lamatsch und M. Bücker:** Programmbibliothek. *April 1988.*

355. **Eichhorn, W.:** Inequalities in the Theory of Economic Inequality.

356. **Fuchs- Seliger, S.:** Dual Models in the Theory of Demand.

357. **Eichhorn, W.:** Generalized Convexity in Economics: Some Examples.

358. **Fuchs-Seliger, S.:** Compensated and Direct Demand without Transitive and Complete Preferences.

359. **Stehling, F.:** Umweltschutz in der wirtschaftswissenschaftlichen Ausbildung.

360. **Stehling, F.:** Ökonomische Aspekte des Umweltschutzes: Ökonomie und Ökologie im Konflikt ?

361. **Bücker, M. und K. Neumann:** Stochastic Single-Machine Scheduling to Minimize the Weighted Expected Flow Time and Maximum Expected Lateness Subject to Precedence Constraints Given by an OR Network. *März 1989.*

362. **Pfingsten, A.:** Fiscal Competition and Equalization of Public Funds.

363. **Geidel, J.:** Richtlinien zur Dokumentation und Archivierung von Software. *März 1989*

364. **Bossert, W und F. Stehling:** On the Uniqueness of Cardinally Interpreted Utility Functions.

365. **Eichhorn, W. und H. Funke:** Prices Before and After the Vertical Merging of Firms.

366. **Weinhardt, C:** The Trade Off between Real Equity and Real Efficiency in Welfare Measurement Theory. Vortragsmanuskript zur Tagung der GMÖR, Ulm 1989.

367. **Buhl, H. U.:** Much Ado About Leasing? *Erschienen in:* Zeitschrift für Betriebswirtschaft, 8, 1989.

368. **Derr, Ph.:** Bericht über das Praktikum OR auf Kleinrechnern im WS 87/88, SS 88. *Juni 1989*

369. **Eichhorn, W. und A. Vogt:** Gemeinsames bei der Messung von Ungleichheit, Streuung, Risiko und Information.

370. **Eichhorn, W.:** Volkswirtschaftliche Auswirkungen der Mikroelektronik; *erschienen in:* Informationstechnologie und strategische Führung, Klaus Spremann und Eberhard Zur (Hrsg.), Gabler, Wiesbaden 1989, 367-377.

371. **Eichhorn, W.:** Equations and Inequalities in the Theory of Measurement.

372. **Buhl, H. U. und G. Satzger:** " Principals" und " Agents" in der Unternehmensplanung.

373. **Buhl, H. U. und N. Erhard:** Steuerlich linearisiertes Leasing: Kalkulation und Steuerparadoxon.

374. **Geidel, J., Hoffmann D. und M. Lachmann:** Teachware in Operations Research. *Dezember 1989*

375. **Fuchs- Seliger, S.:** Reformulation of the Theory of Demand by Compensated Demand Functions.

376. **Chakravarty, S. R. und W. Eichhorn:** The Optimum Size Distribution of Income.

377. Chakravarty, S. R. und W. Eichhorn: An Axiomatic Characterization of a Generalized Index of Concentration.

378. Eichhorn, W.: How not to Lie with Statistics in Regional Analysis.

379. Chakravarty, S. R.: The Optimum Size Distribution of Firms.

380. Chakravarty, S. R.: On Quasi-Orderings of Income Profiles.

381. Chakravarty, S. R.: Ethical Social Index Numbers; *erschienen als Buch:* Springer-Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong, (1990).

382. Chakravarty, S. R. and B. Dutta: Migration and Welfare.

383. Chakravarty, S. R. and A. Majumder: Personal Income Distribution: Development of a New Model and Its Application to U. S. Income Data.

384. Pfingsten, A.: Profit-Based Payment Schemes in the Banking Sector.

385. Derr, Ph.: Projektplanungssoftware. *Januar 1990.*

386. Weinhardt, C.: The Central Role of Efficiency in Inequality and Welfare Measurement Theory.

387. Weinhardt, C.: The Efficiency of Price Income Situations, the Real Average Income - A Characterization -.

388. Eichhorn, W.: Das magische Neuneck. Umwelt und Sicherheit in einer Volkswirtschaft.

389. Bossert, W. und A. Pfingsten: Ordinal Utility and Economic Price Indices.

390. Bossert, W. und F. Stehling: Admissable Transformations for Interpersonally Comparable Utilities : A Rigorous Derivation.

391. Bossert, W. und F. Stehling: Social Preferences as Optimal Compromises.

392. Foulds, L. R., Hoffmann D. und K. Neumann: Stochastic Identical Parallel Machine Scheduling with OR Precedence Constraints. *March 1990.*

393. Fuchs-Seliger, S.: Non-Saturated Preferences and Compensated Demand - A Reexamination.

394. Geidel, J. und M. Lachmann: Projekt" Entscheidungsunterstützung in der Projektplanung" - Zwischenbericht. *Mai 1990.*

395. Brinkmann, K.: Bericht über das Praktikum OR auf Kleinrechnern im WS 88/89, SS 89. *Mai 1990.*

396. Weinhardt, C.: How to measure Price Progression

397. Morlock, M.: Dynamic Programming

398. Neumann, K.: Einführung in die Maschinenbelegungsplanung. *Juli 1990.*

399. Christmann, A. und W. Jorasz: Verfahren zur Aufteilung von Fertigungsgemeinkosten bei Verwendung der Bezugsgrößen Fertigungslohn und Maschinenzeiten

400. Christmann, A.: Synergetics in Kaldor`s Business Cycle Model

401. Christmann, A.: Synergetik in der Ökonomie

402. Chakravarty, S. R. and W. Eichhorn: Measurement of Income Inequalities : True versus Observed Data

403. Fuchs-Seliger, S.: A Reconsideration of Income Compensation

404. Beck, T.: Integrated Capacity and Lot Size Planning in Decentralized Production Planning. *Dezember 1990* (vergriffen).

405. Pfingsten, A. und J. Schneider: Retrieving Inequality Concepts and Progressivity Objectives From Tax Functions via Approximations

406. Geidel, J.: Praktikum " OR auf Kleinrechnern" im WS 89/90, SS 90. *Januar 1991.*

407. Lachmann, M.: Programmbibliothek Operations Research. *Januar 1991.*

408. Bücker, M., Neumann, K., Rubach, T.: Algorithms for Single-Machine Scheduling with Stochastic Outtree Precedence Relations to Minimize Expected Weighted Flow Time or Maximum Expected Lateness. *February 1991.*

409. Geidel, J., Lachmann, M.: Konzept eines modellbasierten Entscheidungsunterstützungssystems. *Februar 1991.*

410. Eichhorn, W.: Uneasy Polygons: Environment and Security Within the System of Aims of an Economy.

411. Eichhorn, W.: Produktionskorrespondenzen

412. Fuchs-Seliger: On the Evaluation of Budget Situations

413. Lachmann, M.: Begriffsgraphen. *April 1991.*

414. Bol, Morlock, Neumann, Pallaschke, Waldmann: Operations Research studieren an der Universität Karlsruhe. *September 1991.*

415. Lachmann, M.: Modelle und Verfahren in entscheidungsunterstützenden Systemen. *Dezember 1991.*

416. Bücker, M., Neumann, K.: Algorithms for Single-Machine Scheduling with Stochastic Outtree Precedence Relations to Minimize Expected Weighted Flow Time or Maximum Expected Lateness. *February 1991.*

417. Egle, K., Fenyi, S.: Eigenvalue Estimations in Input-Output and Growth Models by Monte Carlo Techniques

418. Gerhards, T.: Purchasing Power Parity and Cointegration

419. Bücker, M.: Ein Programmpaket zur Folgeplanung mit stochastischen Anordnungsbeziehungen. *Juli 1992.*

420. Fuchs- Seliger: Order Extensions and Budget Correspondences

421. Lachmann, M., Neumann, K.: A Heuristic for Multi-Product, Multi-Period, Single-Level Batch Production. *September 1992.*

422. Bücker, M.: Object Oriented Operations Research. *October 1992.*

423. Lachmann, M.: Genetische Algorithmen in der Optimierung, Bericht zum Rechnerpraktikum 1990/91. *Oktober 1992.*

424. Neumann, K.: Dynamic Programming - Basic Concepts and Applications. *October 1992.*

425. Neumann, K.: Production and Operations Management. *November 1992.*

426. Geidel, J., Lachmann, M., Präger, R. : Modellierung und Methodenauswahl in Entscheidungsunterstützungssystemen. *März 1993.*

427. Neumann, K.: Produktions- und Operations-Management I. *September 1993.*

428. Neumann, K.: Produktions- und Operations-Management II, 2. Auflage. *April 1994.*

429. Eichhorn, W., Krtscha, M.: Informationsmessung und Beziehungen zur Messung von Steuerung, Risiko, Entropie, Konzentration und Ungleichheit, *erschienen in*: Walter Frisch und Alfred Taudes (Hrsg.): Informationswirtschaft - Aktuelle Entwicklungen und Perspektiven, Physica-Verlag, Heidelberg 1993, 3-20.

430. Gerhards, T.: Strukturelle Wechselkursbeziehungen auf den Internationalen Devisenmärkten. *1993.*

431. Gerhards, T.: Zwanzig Jahre Flexible Wechselkurse - Eine Empirische Bewertung des Monetären Modells. 1993.

432. Fuchs-Seliger, S.: On Competitive Equilibria in Models of Consumer Behaviour. *1994.*

433. LS Neumann, K.: Klausuraufgaben OR I + II mit Lösungen. *September 1994.*

434. LS Neumann, K.: Klausuraufgaben POM I + II mit Lösungen. *Juni 1994.*

435. LS Neumann, K.: Klausuraufgaben Graphen und Netzwerke I + II mit Lösungen. *Juni 1994.*

436. Schneider, W.: Programmbibliothek Operations Research. *April 1994.*

437. Schwindt, C.: Vergleichende Beurteilung mehrerer Varianten der Heuristik von Lambrecht & Vanderveken zur sukzessiven Lösung des integrierten Losgrößen- und Ablaufplanungsproblems. *Mai 1994.*

438. Schnur, B.: Ein objektorientiertes Architekturmodell zur Außendienstberatung.

439. Fuchs-Seliger, S.: On Shephard's Lemma and the Continuity of Income Compensation Functions.

440. Geidel, J.: Eine graphische Modellierungssprache. *April 1994.*

441. Lachmann, M.: Ein Entscheidungsunterstützungssystem für die Standortplanung. Bericht zum Praktikum "Software-Entwicklung im Operations Research" 1992/93. *August 1994.*

442 Neumann, K., Zhan, Ji.: Heuristics for the Minimum Project-Duration Problem with Minimal and Maximal Time Lags under Fixed Resource Constraints. *August 1994.*

443 Brinkmann, K., Neumann, K.: Heuristic Procedures for Resource-Constrained Project Scheduling with Minimal and Maximal Time Lags: The Minimum Project-Duration and Resource-Levelling Problems. *November 1994.*

444 Olt, B.: Indices of Structural Changes *1995*

445 Schnur, B.: Operationalisierung von unscharfem Wissen mittels eines objektorientierten Ansatzes. *1995*

446 Schnur, B.: Realisierung eines regelbasierten Fuzzy-Systems mit der objektorientierten Entwicklungsumgebung ADS und der relationalen Datenbank MS-Access. *1995*

447 **Neumann, K., Schwindt, C.:** Projects with Minimal and Maximal Time Lags: Construction of Activity-on-Node Networks and Applications. *June 1995*

448 **Schneider, W.:** Job Shop Scheduling with Stochastic Precedence Constraints. *February 1995*

449 **Schwindt, C.:** ProGen / max: A New Problem Generator for Different Resource-Constrained Project Scheduling Problems with Minimal and Maximal Time Lags. *July 1995*

450 **Franck, B.; Schwindt, C.:** Different Resource-Constrained Project Scheduling Problems with Minimal and Maximal Time Lags - Models and Practical Applications. *1995*

451 **Redin, J.; Schneider, W.:** Programmbibliothek Operations Research - WIOR Gopher. *März 1995*

452 **Zimpelmann, M.:** Folgen des Versicherungsbinnenmarktes für die Schadenversicherung unter besonderer Berücksichtigung der Kraftfahrtversicherung. *1995*