Drexl, Andreas; Salewski, Frank

Distribution Requirements and Compactness Constraints in School Timetabling. Part I: Model

Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel, No. 383

No. 383

Distribution Requirements and Compactness Constraints in School Timetabling. Part I: Model

Drexl / Salewski

January 1996

Andreas Drexl, Frank Salewski
Institut für Betriebswirtschaftslehre, Christian-Albrechts-Universität zu Kiel,
Olshausenstr. 40, 24118 Kiel, Germany
Tel.&Fax ++49(0)431-880-1531
Email: Drexl@bwl.uni-kiel.de, Salewski@bwl.uni-kiel.de.

Abstract: Different constraints must be taken into account in school timetabling. Part I of this paper provides three major contributions: First, we show that several types of important constraints may be modelled using the unifying framework of partially renewable resources. Among these constraints are: No class, subject, room, and teacher overlaps; class, subject, room, and teacher unavailabilities; compactness constraints; preassignment constraints; lectures to be given simultaneously; lunch breaks, etc. Note that compactness constraints typically require no free time (other than lunch breaks or time to move from one building to another) between lessons for the pupils. Second, distribution requirements for lessons of different lengths are modelled in a novel way by the use of the so-called multiple mode concept with mode identity constraints. Third, we contribute to the distinction of school timetabling and course scheduling for universities. In part II of this paper parallel greedy randomized methods and genetic algorithms will be presented.

Keywords: Timetabling, distribution requirements, compactness constraints, partially renewable resources, multiple modes, mode identity

1. Introduction

The school timetabling problem requires to schedule a set of meetings between teachers and pupils over a set of time periods, where some resources must be available and several additional constraints have to be met. The timetabling problem includes a large variety of problems on different levels of an education system.

Among the constraints which have to be taken into account are: No class, subject, room, and teacher overlaps are allowed; one has to take care of class, subject, room, and teacher availabilities; compactness constraints have to be regarded; preassignment of lectures must be possible; some lectures have to be given simultaneously; lunch breaks have to be observed, etc. Note that compactness constraints typically require no free time (other than lunch breaks or time to move from one building to another) between lessons for the pupils. Most of the constraints mentioned will be modelled using the unifying framework of partially renewable resources. Moreover, it is necessary to distribute lessons of different lengths. These requirements are modelled by the use of the so-called multiple mode concept with mode identity constraints.

Timetabling models and methods known so far are closely related to one of the following concepts: *(i)* In graph colouring adjacent vertices must have different colours (cp. e.g. Cangalovic and Schreuder 1991, 1992, Kubale and Jackowski 1985); unfortunately, here it is difficult or even impossible to incorporate most of the constraints already mentioned. *(ii)* Multi-index transportation problems generalize two-dimensional assignment models in order to make more than two items such as classes and teachers tractable (cp. e.g. Junginger 1993). *(iii)* The impact of logic programming approaches is to express constraints via Prolog predicates (cp. e.g. Fahrion and Dollansky 1992, Kang and White 1992). *(iv)* Local search methods such as simulated annealing (cp. e.g. Abramson 1991, Dige, Lund, and Raun 1993) or tabu search (cp. e.g. Hertz 1991, 1992, Costa 1994) try to escape from local optima. Note that these approaches incorporate different constraints in different ways. The focus of this paper is to provide a unique framework for modelling different constraints via partially renewable resources.

The paper is organized as follows: Section 2 provides the problem setting. A mathematical programming formulation is introduced in Section 3 and explained by the use of examples. In Section 4 the differences between timetabling for schools and course scheduling for universities are analyzed. In addition, our framework is related to that of other authors. Finally, Section 5 gives a summary and an outline of future work.

2. Problem Setting

The outline of this section is as follows: First, we give some motivation and background. Second, we introduce the assumptions and the notation in such a way that all parameters are directly interpretable w.r.t the problem at hand. Third, we provide transformations such that some parameters vanish and others are simpler to state. Subsequently, the model can be formulated in Section 3 in a compact way.

Basically, we assume that teachers have already been assigned to subjects. Obviously, if this has to be done by an algorithm, an approach based on the *generalized assignment problem* would be appropriate (cp. e.g. Fisher, Jaikumar, and Van Wassenhove 1986, Cattrysse and Van Wassenhove 1992).

2.1 Background and Motivation

Before introducing the parameters which are necessary to derive the model subsequently, we first present the situation under consideration in an informal way in order to facilitate understanding of the technical details provided below.

School timetabling models must accommodate the characteristics and regulations of specific education systems. Therefore the problem under consideration varies from country to country. Consequently we find models and algorithms for schools in Australia (cf. Abramson 1991), Great Britain (cf. Lawrie 1969, Aust 1976, and Papoulias 1980), Denmark (cf. Dige, Lund, and Raun 1993), Germany (cf. Junginger 1986), Japan (cf. Kitagawa and Ikeda 1988), the Netherlands (cf. de Gans 1981), Spain (cf. Alvarez-Valdes, Martin, and Tamarit 1994), and Switzerland (cf. Osterman and de Werra 1982, Chahal and de Werra 1989), respectively. In addition, school timetabling problems are quite different in one country as well as depending on the level of the education system addressed.

More precisely, in Germany in primary schools we have classes in the traditional sense (classes 1 to 4), i.e groups of pupils at the same grading level with essentially the same teaching program. In secondary schools, we have classes in this sense (classes 5 to 10) and we have "classes" which do not have the same curriculum ("classes" 11 and 13). Thus, for secondary schools a model has to be developed which covers this issue as well. Moreover, we have to tackle the problem how to split large teaching units (e.g. six hours math per week) in

smaller ones and how to schedule them. In fact, the situation we are confronted with in secondary schools in Germany (and which is the topic of this paper) seems to be the most general one which has ever been addressed in the open literature.

To start with the informal description, we consider timetabling problems where we have two types of *pupils:*

- typ I pupils (pupils of one class having the same curriculum; class in the traditional sense; choose one pupil as the "representative")
- typ II pupils (pupils of one "class", i.e. with the same grading level, which do not have the same curriculum; each pupil is dealt with separately)

Consider the following instance: As typ I pupils we have the pupils of classes 5a, 5b, 6a, 6b. As typ II pupils we have pupils 1 and 2 of "class" 12 and pupil 1 of "class" 13, respectively. By relabeling we get the pupils 1 to 7.

Moreover, we have two types of subjects, too, i.e. there are main (e.g. german, math) and subsidiary (e.g. biology) subjects. Once more, by relabeling we get the subjects 1 to 3. Note that it is necessary to differentiate between main and subsidiary subjects, because timetables with e.g. six hours of (different) main subjects on half a day will not be acceptable. We have to formulate constraints on the amount of main subjects to be visited on a whole day or on half a day by the pupils (and to be given by the teachers).

In addition, there are Basic Teaching Units (BTUs). Consider the instance given in Table 1. Two pupils have to attend lectures in german, math, and biology, labeled as BTUs 1 to 4. Note that each BTU is characterized by the attributes length, required teacher, and requested typ of room (see below), respectively.

Table 1: Basic Teaching Units

pupil	subject	BTU
1	german	1
1	math	2
2	german	3
2	biology	4

BTUs in general are large teaching units (e.g. six hours math per week) where we have to decide how to split them into smaller ones (called blocks) and how to schedule them. This splitting and scheduling will be called "distributing". For the distribution of BTUs we introduce modes, blocks, and jobs as in the example of Table 2.

Table 2: Distribution of BTUs

BTU	length	block	job	mode	costs
1	2	ϕ ϕ	1 2	0^d	big
		□□ ϕ		1	0
		□ □		2	0
2	3	ϕ ϕ ϕ	3 4 5	0^d	big
		□□□ ϕ ϕ		1	5
		□□ □ ϕ		2	0
		□ □ □		3	1

For BTU 1 with a length of 2 hours we have two possible modes of distribution: First, two consecutive hours (□ ≃ 1 hour, ϕ ≃ 0 hours), i.e. one block of a two-hour lecture and one dummy block (i.e. no distribution), and two blocks with a duration of one hour each. Analogously, for BTU 2 with the length of 3 hours three modes of distribution are provided. Note that an artificial dummy mode 0 (cp. the superscript ^{d}) is introduced for technical reasons. Moreover, notice that blocks will be denoted as jobs and that costs are associated with each mode of processing the jobs of a specific BTU. Finally, it is noteworthy to mention that the definition of the dummy blocks secures an identical number of blocks for each BTU irrespective of the modes introduced. The details will be explained below.

Each BTU requires a specific Type Of Room (TOR) as indicated by the following example. In Table 3 three TORs are given, # denotes the number of rooms available; the rooms are relabeled as indicated. In Table 4 the set of TORs required by each BTU is provided. I.e. BTU 2 may either be given in one of the three available classrooms of size 1 or in one of the two available classrooms of size 2. BTU 1 must be given in the language labaratory.

Table 3: Labeling of Rooms

TOR	#	label
1 (classroom, size 1)	3	1, 2, 3
2 (classroom, size 2)	2	4, 5
3 (lang. lab.)	1	6

Table 4: Room Requirements

BTU	set of TOR required
1	{3}
2	{1,2}

Pupils (types I and II), subjects, teachers, and rooms will be treated as resources. Relabeling the resources then allows to formulate capacity constraints for subsets of periods as outlined in the following. Consider the simple instance given in Table 5 where we have two days with four a.m.-hours and two p.m.-hours as periods:

Table 5: Days, Hours, and Periods

Mon	1	2	3	4	5	6
Tue	7	8	9	10	11	12

Now, we define, e.g., $\mathcal{P}_1 = \{1\}$, ... , $\mathcal{P}_{12} = \{12\}$, $\mathcal{P}_{13} = \{1,...,12\}$, $\mathcal{P}_{14} = \{1,2,3,4\}$, $\mathcal{P}_{15} = \{5,6\}$, $\mathcal{P}_{16} = \{7,8,9,10\}$, and $\mathcal{P}_{17} = \{11,12\}$ as Subsets Of Periods (SOPs). Clearly, the period subsets $\mathcal{P}_1$ to $\mathcal{P}_{12}$ facilitate to model renewable resources constraints, while $\mathcal{P}_{13}$ will do this for nonrenewable resources. Moreover, $\mathcal{P}_{14}$ to $\mathcal{P}_{17}$ allow to formulate so-called partially renewable resource constraints. Noteworthy to mention, that the period subsets must not be pairwise disjoint and that the partially renewable case covers the renewable and the nonrenewable one as special cases (cp. Böttcher, Drexl, Kolisch, and Salewski 1996).

2.2 Assumptions and Notation

In this subsection, the parameters which are necessary to formulate the model will be introduced. Let denote

T : the number of periods (hours) per week ($T \in \mathbb{N}$) indexed by t, $1 \leq t \leq T$.

Π : the number of subsets of periods ($\Pi \in \mathbb{N}$) indexed by π, $1 \leq \pi \leq \Pi$.

$\mathcal{P}_\pi$: a specific subset of periods ($1 \leq \pi \leq \Pi$); $\mathcal{P}_\pi$ contains periods for which certain constraints have to be observed.

$\mathcal{S}$: a set of period subsets; in $\mathcal{S}$ we have subsets of periods where minimal time lags have to be observed; consider in the example of Subsection 2.1 the situation, that $\mathcal{P}_{14}$ and $\mathcal{P}_{15}$ are elements of $\mathcal{S}$ (i.e. $\mathcal{S} = \{14,15\}$), then for some lectures to be given on Monday a.m. and p.m. minimal time lags may be regarded.

L : the number of teachers available ($L \in \mathbb{N}$) indexed by l, $1 \leq l \leq L$.

V : the number of different TORs ($V \in \mathbb{N}$) indexed by v, $1 \leq v \leq V$.

W : the number of sets of TORs ($W \in \mathbb{N}$) indexed by w, $1 \leq w \leq W$.

Z_w : a set of TORs; clearly, each Z_w, $1 \leq w \leq W$, must contain at least one room type, i.e. $Z_w \neq \phi$.

$\hat{Q}$: the number of grading levels having the same curriculum ($\hat{Q} \in \mathbb{N}_0$) indexed by $\hat{q}$, $0 \leq \hat{q} \leq \hat{Q}$.

$\tilde{Q}$: the number of grading levels which do not have the same curriculum ($\tilde{Q} \in \mathbb{N}_0$) indexed by $\tilde{q}$, $0 \leq \tilde{q} \leq \tilde{Q}$.

$G_{\hat{q}}$: ($1 \leq \hat{q} \leq \hat{Q}$) the number of classes with the same grading level $\hat{q}$ having the same curriculum, indexed by g, $1 \leq g \leq G_{\hat{q}}$.

H : the total number of classes, i.e. with and without the same curriculum ($H \in \mathbb{N}$) indexed by h, $1 \leq h \leq H$; clearly we have

$$H = \Sigma_{\hat{q}=1}^{\hat{Q}} G_{\hat{q}} + \tilde{Q}.$$

$S_{\tilde{q}}$: ($1 \leq \tilde{q} \leq \tilde{Q}$) the number of type II pupils ($S_{\tilde{q}} \in \mathbb{N}$) indexd by s, $1 \leq s \leq S_{\tilde{q}}$.

$\hat{F}_h$: ($1 \leq h \leq H$) the number of main subjects of class h ($\hat{F}_h \in \mathbb{N}_0$) indexd by $\hat{f}$, $0 \leq \hat{f} \leq \hat{F}_h$.

$\tilde{F}_h$: ($1 \leq h \leq H$) the number of subsidiary subjects of class h ($\tilde{F}_h \in \mathbb{N}_0$) indexd by $\tilde{f}$, $0 \leq \tilde{f} \leq \tilde{F}_h$.

F_h : ($1 \leq h \leq H$) the total number of subjects ($F_h \in \mathbb{N}$) indexed by f, $1 \leq f \leq F_h$; clearly, we have $F_h = \hat{F}_h + \tilde{F}_h$. Notice, that (h,f) denotes a BTU introduced above.

B_{hf} : ($1 \leq h \leq H$, $1 \leq f \leq F_h$) the number of blocks of BTU (h,f) ($B_{hf} \in \mathbb{N}$) indexed by b, $1 \leq b \leq B_{hf}$; note that B_{hf} must not be equal to the length of BTU (h,f).

M_{hf} : ($1 \leq h \leq H$, $1 \leq f \leq F_h$) the number of modes of BTU (h,f) ($M_{hf} \in \mathbb{N}$) indexed by m, $0 \leq m \leq M_{hf}$, i.e. including the dummy mode 0.

d_{hfbm} : ($1 \leq h \leq H$, $1 \leq f \leq F_h$, $1 \leq b \leq B_{hf}$, $0 \leq m \leq M_{hf}$) the length of block b of BTU (h,f) in mode m ($d_{hfbm} \in \mathbb{N}_0$).

c_{hfm} : ($1 \leq h \leq H$, $1 \leq f \leq F_h$, $0 \leq m \leq M_{hf}$) the costs of processing BTU (h,f) in mode m ($c_{hfm} \in \mathbb{R}_{\geq 0}$).

$\mathcal{N}_{hfbm}$: ($1 \leq h \leq H$, $1 \leq f \leq F_h$, $1 \leq b \leq B_{hf}$, $0 \leq m \leq M_{hf}$) the set of periods in which block b of BTU (h,f) to be processed in mode m must not be finished.

$\mathcal{G}_{hf}$: $(1 \le h \le H,\ 1 \le f \le F_h)$ the set of BTUs (θ,ζ) which must be processed in parallel with BTU (h,f); more precisely:

$\mathcal{G}_{hf} :=$ { (θ,ζ) | $1 \le \theta \le H,\ 1 \le \zeta \le F_\theta$ $\wedge$ all blocks of BTU (θ,ζ) must be processed in parallel with all blocks of BTU (h,f)}

Clearly, this definition requires, that all the BTUs (θ,ζ) must have equal length and that they have to be processed in the same mode.

$\mathcal{E}_{hf}$: $(1 \le h \le H,\ 1 \le f \le F_h)$ the set of BTUs (θ,ζ) to which a minimum time lag must be regarded; more precisely:

$\mathcal{E}_{hf} :=$ { (θ,ζ) | $1 \le \theta \le H,\ 1 \le \zeta \le F_\theta$ $\wedge$ for all blocks β of BTU (θ,ζ) it holds: block β of BTU (θ,ζ) can only be scheduled after block b of BTU (h,f) in the SOP $\mathcal{P}_\pi$ $(\pi \in \mathcal{P})$ if a minimum time lag is regarded}

Clearly, if the minimal time lag has to be regarded in both "directions" then $\mathcal{E}_{\theta\zeta}$ must contain BTU (h,f) as well. In addition, if there is no time lag to any of the of the BTUs $(\theta,\zeta) \neq$ (h,f) then we have $\mathcal{E}_{hf} = \phi$.

$\sigma_{hf\theta\zeta}$: $(1 \le h,\ \theta \le H,\ 1 \le f,\ \zeta \le F_h)$ minimal time lag between BTU $(\theta,\zeta) \in \mathcal{E}_{hf}$ and BTU (h,f) in periods $(\sigma_{hf\theta\zeta} \in \mathbb{N}_0)$.

For the purpose of a compact formulation of the model some of the parameters introduced so far will be considered as resources in the sequel. This is done for the (real) resources teacher, type I and II pupils, and rooms, respectively. In addition, subjects will be considered as (abstract) resources as well.

Resource usages will be defined for block b of BTU (h,f) processed in mode m $(1 \le h \le H,\ 1 \le f \le F_h,\ 1 \le b \le B_{hf},\ 0 \le m \le M_{hf})$, resource usage is defined in units per period. For the sake of shortness, (h,f,b) will denote block b of BTU (h,f). Note that "k" is used as denotator for resource usage, while "K" will denote resource availability.

kL_{hfbml} : usage of resource teacher l $(1 \le l \le L)$; more precisely:
$kL_{hfbml} :=$ 1, if (h,f,b) is given by teacher l in mode $m \neq 0$ ($kL_{hfbml} :=$ 0, otherwise)

$kK_{hfbm\hat{q}g}$: usage of resource class $(\hat{q},g)$ $(1 \le \hat{q} \le \hat{Q},\ 1 \le g \le G_{\hat{q}})$; more precisely:
$kK_{hfbm\hat{q}g} :=$ 1, if (h,f,b) is given for class $(\hat{q},g)$ in mode $m \neq 0$ ($kK_{hfbm\hat{q}g} :=$ 0, otherwise)

$kS_{hfbm\tilde{q}s}$: usage of resource pupil $(\tilde{q},s)$ $(1 \le \tilde{q} \le \tilde{Q},\ 1 \le s \le S_{\tilde{q}})$; more precisely:
$kS_{hfbm\tilde{q}s} :=$ 1, if (h,f,b) is given for pupil $(\tilde{q},s)$ in mode $m \neq 0$ ($kS_{hfbm\tilde{q}s} :=$ 0, otherwise)

kR_{hfbmw} : usage of resource TOR w ($1 \le w \le W$); more precisely:
$kR_{hfbmw} := 1$, if (h,f,b) is given in room type w in mode $m \neq 0$ ($kR_{hfbmw} := 0$, otherwise)

$kF_{hfbm\theta\zeta}$: usage of resource BTU (θ,ζ) ($1 \le \theta \le H$, $1 \le \zeta \le F_\theta$); more precisely:
$kF_{hfbm\theta\zeta} := 1/d_{hfbm}$, if $h = \theta \wedge f = \zeta \wedge m \neq 0$ ($kF_{hfbm\theta\zeta} := 0$, otherwise)

$kU_{hfbm\hat{q}g}$: usage of resource class $(\hat{q},g)$ ($1 \le \hat{q} \le \hat{Q}$, $1 \le g \le G_{\hat{q}}$); more precisely:
$kU_{hfbm\hat{q}g} := 1$, if (h,f,b) of the main subject (h,f) is given for class $(\hat{q},g)$ in mode $m \neq 0$ ($kU_{hfbm\hat{q}g} := 0$, otherwise)

$kO_{hfbm\tilde{q}s}$: usage of resource pupil $(\tilde{q},s)$ ($1 \le \tilde{q} \le \tilde{Q}$, $1 \le s \le S_{\tilde{q}}$); more precisely:
$kO_{hfbm\tilde{q}s} := 1$, if (h,f,b) of the main subject (h,f) is given for class $(\tilde{q},g)$ in mode $m \neq 0$ ($kO_{hfbm\tilde{q}s} := 0$, otherwise)

Resource availabilities will be defined for SOPs $\mathcal{P}_\pi$ ($1 \le \pi \le \Pi$). They are given in resource units (hours, rooms, etc.) $\in \mathbb{N}_0$.

$KL_{l\pi}$: availability of resource teacher l ($1 \le l \le L$); more precisely:
$KL_{l\pi}$:= maximum number of hours given by teacher l in SOP $\mathcal{P}_\pi$

$KK_{\hat{q}g\pi}$: availability of resource class $(\hat{q},g)$ ($1 \le \hat{q} \le \hat{Q}$, $1 \le g \le G_{\hat{q}}$); more precisely:
$KK_{\hat{q}g\pi}$:= maximum number of hours given for class $(\hat{q},g)$ in SOP $\mathcal{P}_\pi$

$KS_{\tilde{q}s\pi}$: availability of resource pupil $(\tilde{q},s)$ ($1 \le \tilde{q} \le \tilde{Q}$, $1 \le s \le S_{\tilde{q}}$); more precisely:
$KS_{\tilde{q}s\pi}$:= maximum number of hours given for pupil $(\tilde{q},s)$ in SOP $\mathcal{P}_\pi$

$KR_{w\pi}$: availability of resource TOR w ($1 \le w \le W$); more precisely:
$KR_{w\pi}$:= maximum number of hours TOR w is available in SOP $\mathcal{P}_\pi$

$KF_{hf\pi}$: availability of BTU (h,f) ($1 \le h \le H$, $1 \le f \le F_h$); more precisely:
$KF_{hf\pi}$:= maximum number of blocks of BTU (h,f) in SOP $\mathcal{P}_\pi$

$KU_{\hat{q}g\pi}$: availability of the main subjects of class $(\hat{q},g)$ ($1 \le \hat{q} \le \hat{Q}$, $1 \le g \le G_{\hat{q}}$); more precisely: $KU_{\hat{q}g\pi}$:= maximum number of hours of the main subjects for class $(\hat{q},g)$ in SOP $\mathcal{P}_\pi$

$KO_{\tilde{q}s\pi}$: availability of the main subjects of pupil $(\tilde{q},s)$ $(1 \le \tilde{q} \le \tilde{Q},\ 1 \le s \le S_{\tilde{q}})$; more precisely: $KO_{\tilde{q}g\pi}$:= maximum number of hours of the main subjects for pupils $(\tilde{q},s)$ in SOP $\mathcal{P}_\pi$

Note that $kU_{hfbm\hat{q}g}$ in addition to $kK_{hfbm\hat{q}g}$ is necessary in order to model constraints for the number of main subject hours to be given within specific SOPs (especially for half days as outlined in Subsection 2.1). The same is true for $kO_{hfbm\tilde{q}g}$ / $kS_{hfbm\tilde{q}g}$, for $KU_{\hat{q}g\pi}$ / $KK_{\hat{q}g\pi}$, and for $KO_{\tilde{q}s\pi}$ / $KS_{\tilde{q}s\pi}$, respectively.

Based on these parameters now a mathematical programming model could be stated. But, on account of the large number of parameters and indices it would be difficult to get insight into the structure of the model. Moreover, simple transformations cut down the large set of parameters and allow then to state a quite compact model.

2.3 Transformations

In the following we describe how to transform the parameters introduced in the previous subsection. Although the basic idea of these transformations is simple, their formal description is burdensome. Therefore, we refrain from the tedious task of describing the transformations precisely. Here we only give a flavour of what to do. (For the example of Subsection 3.2 the transformations are given in the Appendix.) We use the operator "$\leftarrow$" in order to point to a specific transformation.

U $\leftarrow$ $(1 \le h \le H,\ 1 \le f \le F_h)$ the number of BTUs indexed by u, $1 \le u \le U$; then, (h,f) is replaced by u any time it occurs.
Now, BTUs u and θ with $\theta \in \mathcal{G}_u$ which have to scheduled in parallel are unified; consequently, e.g. $\mathcal{N}_{ubm}$, $\mathcal{E}_u$, c_{um}, and the resource usages have to be adjusted.

J $\leftarrow$ $(1 \le u \le U,\ 1 \le b \le B_u)$ the number of jobs indexed by j, $1 \le j \le J$; then, (u,b) is replaced by j any time it occurs.
Now, the parameters a_u and e_u have to be introduced; a_u (e_u) points to the first (last) job of BTU u.

R $\leftarrow$ $(1 \le l \le L,\ 1 \le b \le B_u, 1 \le \hat{q} \le \hat{Q},\ 1 \le g \le G_{\hat{q}},\ 1 \le \tilde{q} \le \tilde{Q},\ 1 \le s \le S_{\tilde{q}},\ 1 \le w \le W)$ the number of resources indexed by r, $1 \le r \le R$.

k_{jmr} $\leftarrow$ the number of units of resource r used by job j being processed in mode m $(1 \le j \le J,\ 0 \le m \le M_j, 1 \le r \le R)$; clearly, without citing the indices completely, k_{jmr} substitutes kL, kK, kS, kR, kF, kU, and kO appropriately.

$K_{r\pi}$ ← the availability level of resource r in SOP π ($1 \le r \le R$, $1 \le \pi \le \Pi$); clearly, without citing the indices completely, $K_{r\pi}$ substitutes KL, KK, KS, KR, KF, KU, and KO appropriately.

Now, having defined all the parameters in a convenient way, the mathematical programming model can be stated easily.

3. Model

The outline of this section is as follows: First, we provide the mathematical programming formulation. Second, we show by examples what we are able to model with the formulation provided.

3.1 Mathematical Programming Formulation

In the following the model is stated for two reasons: *(i)* First of all it precisely describes what we want to do in terms of optimization under constraints. *(ii)* Second and even more important, the model will be used in part II of this paper to provide benchmark solutions for small instances in order to evaluate the performance of heuristics.

The model is formulated in terms of binary optimization. The decision variables couple jobs, modes, and periods, respectively; they are defined as follows: $x_{jmt} = 1$, if job j is assigned to mode m and finished in period t ($x_{jmt} = 0$, otherwise).

Now, the equations (1)-(7) represent the model.

$$\min \sum_{u=1}^{U} \sum_{m=0}^{M_u} \sum_{t=1}^{T} c_{um}\, x_{a_u mt} \tag{1}$$

$$\text{s.t.} \quad \sum_{m=0}^{M_u} \sum_{t \in \mathcal{N}_{jm}} x_{jmt} = 0 \qquad (1 \le u \le U;\ a_u \le j \le e_u) \tag{2}$$

$$\sum_{m=0}^{M_u} \sum_{t=1}^{T} x_{a_u mt} = 1 \qquad (1 \le u \le U) \tag{3}$$

$$\sum_{t=1}^{T} x_{a_u mt} = \sum_{t=1}^{T} x_{jmt} \qquad (1 \le u \le U;\ a_u + 1 \le j \le e_u;\ 0 \le m \le M_u) \tag{4}$$

$$\sum_{u=1}^{U} \sum_{m=0}^{M_u} \sum_{j=a_u}^{e_u} k_{jmr} \sum_{t \in \mathcal{P}_\pi} \sum_{p=t}^{t+d_{jm}-1} x_{jmp} \le K_{r\pi} \qquad (1 \le r \le R;\ 1 \le \pi \le \Pi) \tag{5}$$

$$\sum_{m=1}^{M_u} x_{jmt} + \sum_{n=1}^{M_y} \sum_{\substack{p=t \\ p \in \mathcal{P}_\tau}}^{t+\sigma_{uy}+d_{hm}-1} x_{hnp} \leq 1$$

$$(1 \leq u \leq U; a_u \leq j \leq e_u; \forall y \in \mathcal{E}_u; a_y \leq h \leq e_y; \forall \tau \in \mathcal{P}; \forall t \in \mathcal{P}_\tau) \quad (6)$$

$$x_{jmt} \in \{0,1\} \qquad (1 \leq u \leq U; a_u \leq j \leq e_u; 0 \leq m \leq M_u; 1 \leq t \leq T) \quad (7)$$

Some explanations shall be given as follows: Equation (1) minimizes the sum of costs of processing the first block of each BTU in the chosen mode m. To take the first job w.l.o.g. is valid in conjunction with the mode identity-constraints (4). Equations (2) take care of the set of forbidden periods. Equations (3) force the first block of each BTU to be chosen in one of the available modes. (3) and (4) together secure that all blocks of each BTU are processed exactly once in the same mode. Resource constraints are covered by (5). Constraints (6) have a look on the minimal time lags; clearly, it is sufficient to disregard mode m = 0.

While many details of the model so far have not been discussed thoroughly, we refer the reader to Subsection 3.2 where the modelling capabilities will be illustrated to some extent by examples.

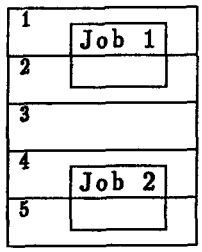
We finish Subsection 3.1 with two remarks: *(i)* The model (1)-(7) closely relates to multi-mode resource-constrained project scheduling. More specific, it generalizes the multi-mode concept by the introduction of the mode identity-constraints. In addition, the constraints (2) and (6) are entirely new, because usually they are not covered by project scheduling models. *(ii)* Disregarding the mode m = 0 the feasibility variant of the model (1)-(7) is NP-complete while the optimization problem is NP-hard (cf. the complexity results given in Salewski, Schirmer, and Drexl 1996 as well as the related work provided in Even, Itai, and Shamir 1976, Garey and Johnson 1979). In fact, the only reason for introducing the dummy mode m = 0 is to penalize infeasibility which clearly cannot be excluded a priori.

3.2 Examples and Illustrations

Clearly, one large instance could serve to illustrate all the relationships covered by the model (1)-(7). We prefer, however, to provide insight into local mechanisms by giving some pieces of data.

To start with, first we will show that constraints (5) for resources r with $0 < k_{jmr} < 1$ do not cover the case where only a subset of jobs is scheduled in one SOP. Consider the five period instance given in Figure 1. Assume that (blocks) jobs 1 and 2 with $k_{1mr} = k_{2mr} = 1/2$ belong to BTU 1. Moreover, assume that at most one of the two blocks must be processed in SOP $\mathcal{P}_1 = \{2,3,4\}$. Clearly, then constraints (5) only take into account one of the two hours per job and thus the schedule depicted in Figure 1 would become feasible.

Figure 1: Illustration of Constraints (5)



This case could be covered by constraints (5)'. The operator $\lceil \alpha \rceil$ denotes the smallest integer greater than or equal α. Awfully this operator makes the constraints (5)' nonlinear and this would prevent to solve instances by the use of commercial MIP-solvers.

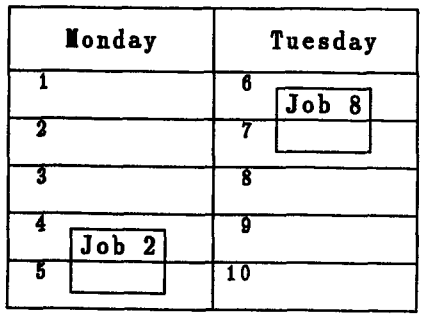
$$\sum_{u=1}^{U} \sum_{m=0}^{M_u} \sum_{j=a_u}^{e_u} \left\lceil k_{jmr} \sum_{t \in \mathcal{P}_\pi} \sum_{p=t}^{t+d_{jm}-1} x_{jmp} \right\rceil \leq K_{r\pi} \qquad (1 \leq r \leq R;\ 1 \leq \pi \leq \Pi) \quad (5)'$$

Therefore, we proceed as follows: The instance generator described in part II of this paper only generates instances in which the situation described in Figure 1 might not occur. Then, the instances remain tractable with MIP-solvers. Moreover, the approximation methods presented in part II of this paper cover constraints (5)' as well.

Note that compactness constraints may be taken into account by constraints (5)' also. One has to define unavailabilities for the resources teacher, class, and pupil on SOPs appropriately. Suppose e.g. one class with a total of 28 hours of lessons per week. Now, compactness can be enforced by allowing lessons for this class only in the first six periods of each day from Monday to Friday.

Let us illustrate now constraints (6). Consider the example given in Figure 2. Assume $\mathcal{P}_1 = \{1,2,3,4,5\}$, $\mathcal{P}_2 = \{6,7,8,9,10\}$, $\mathcal{P} = \{1,2\}$, BTU 1 and BTU 3 with three jobs each, i.e. $a_1 = 1$, $e_1 = 3$, $a_3 = 6$, $e_3 = 8$, respectively. Moreover, assume that the minimal time lag $\sigma_{13} = 2$ between jobs of BTU 1 and jobs of BTU 3 has to be regarded. Now, although the distance between jobs 2 and 8 is 0 periods only, constraints (6) are preserved, because $\mathcal{P} = \{1,2\}$ restricts the application of the distance constraints to SOPs belonging to different days.

Figure 2: Illustration of Constraints (6)



We now consider a small but nontrivial instance which covers some of the constraints simultaneously. A timetable for two days, say Monday and Tuesday, each with four periods, has to be constructed. The following situation arises:

- First, there is one class having the same curriculum with two BTUs, i.e. the three hour main subject German given by teacher 2 in a classroom and the two hour subsidiary subject Biology given by teacher 1 either in a classroom or in a biology room. If both Biology and German will be given on one day and Biology preceedes German then at least an one hour break must be scheduled. Preferably German should be split into a two hour and an one hour block (mode 1), but three one hour blocks (mode 2) would be acceptable also. Biology should be split into two one hour blocks (mode 2), but one two hour block (mode 1) would be acceptable also.
- Second, there is one class without the same curriculum consisting of four pupils. Pupils 1 and 2 have two hours English given by teacher 1 in a classroom, while pupils 3 and 4 have two hours Mathematics given by teacher 2 in a classroom also. Both English and Mathematics are subsidiary courses and have to be scheduled in parallel. Both English and Mathematics should be split into two one hour blocks (mode 2), but one two hour block (mode 1) would be acceptable also.
- The following constraints have to be regarded: Only one block of each BTU is allowed to be scheduled per day, disregarding the length of the block. For the class at most two hours of main subjects are allowed per day, on Tuesday at most three hours in total. For the pupils, at most three main hours are allowed per day, but neither on Monday nor on Tuesday the fourth hour is available, i.e. periods 4 and 8 are excluded. Teacher 1 (2) is not available for the first Tuesday (Monday) period, i.e. in period 5 (1). There are two classrooms and one biology room.

W.r.t. the notation introduced in Section 2, this timetabling situation can be expressed by the following parameter instantiations:

- Periods
  $T = 8$, $\Pi = 11$, $\mathcal{P}_1 = \{1\}$, ..., $\mathcal{P}_8 = \{8\}$, $\mathcal{P}_9 = \{1,...,8\}$, $\mathcal{P}_{10} = \{1,...,4\}$, $\mathcal{P}_{11} = \{5,...,8\}$, $\mathcal{P} = \{10,11\}$.
- Teachers, classes and subjects
  $L = 2$, $\hat{Q} = 1$, $G_1 = 1$, $\tilde{Q} = 1$, $S_1 = 4$, $H = 2$;
  $\hat{F}_1 = 1$, $\tilde{F}_1 = 1$, $F_1 = 2$; $B_{1,1} = 3$, $M_{1,1} = 2$, $B_{1,2} = 2$, $M_{1,2} = 2$;
  $\hat{F}_2 = 0$, $\tilde{F}_2 = 2$, $F_2 = 2$; $B_{2,1} = B_{2,2} = 2$, $M_{2,1} = M_{2,2} = 2$.
- Types of rooms
  $V = 2$ (classroom and biology room), $W = 2$, $Z_1 = \{1\}$ (i.e. classroom), $Z_2 = \{1,2\}$ (i.e. classroom and biology room).

- Forbidden periods, minimal time lags, and parallel BTUs
  $\mathcal{N}_{1,1,1,1} = \mathcal{N}_{1,2,1,1} = \mathcal{N}_{2,1,1,1} = \mathcal{N}_{2,2,1,1} = \{1,5\}$, $\mathcal{E}_{1,2} = \{(1,1)\}$, $\sigma_{1,2,1,1} = 1$, $\mathcal{E}_{1,1} = \mathcal{E}_{2,1} = \mathcal{E}_{2,2} = \phi$; $\mathcal{G}_{2,1} = \{(2,2)\}$, $\mathcal{G}_{2,2} = \{(2,1)\}$, $\mathcal{G}_{1,1} = \mathcal{G}_{1,2} = \phi$.

- Duration of blocks (jobs) and costs (cf. Table 6)

Table 6: Durations

$d_{1,1,b,m}$		b		
		1	2	3
m	0	0	0	0
	1	2	1	0
	2	1	1	1

$d_{1,2,b,m}$ $d_{2,1,b,m}$ $d_{2,2,b,m}$		b	
		1	2
m	0	0	0
	1	2	0
	2	1	1

For the costs we assume $c_{1,1,m} = (100,1,3)$, $c_{1,2,m} = c_{2,1,m} = c_{2,2,m} = (100,3,1)$, m = 0,1,2.

- Resource usages (cf. Table 7; modes m = 0 and entries equal 0 are omitted)

Table 7: Resource Usages

	h	1					2			
	f	1			2		1		2	
	b	1	2	3	1	2	1	2	1	2
	m	1 2	1 2	1 2	1 2	1 2	1 2	1 2	1 2	1 2
kL	1				1 1	1	1 1	1		
	2	1 1	1 1	1					1 1	1
kK	(1,1)	1 1	1 1	1	1 1	1				
kS	(1,1)						1 1	1		
	(1,2)						1 1	1		
	(1,3)								1 1	1
	(1,4)								1 1	1
kR	1	1 1	1 1	1			1 1	1	1 1	1
	2	1 1	1 1	1	1 1	1	1 1	1	1 1	1
kF	(1,1)	½ 1	1 1	1						
	(1,2)				½ 1	1				
	(2,1)						½ 1	1		
	(2,2)								½ 1	1
kU	(1,1)	1 1	1 1	1						
kO	(1,1)									
	(1,2)									
	(1,3)									
	(1,4)									

kR equals 1 whenever a block uses a TOR. Thus, each block of BTUs German, English and Mathematics to be given in a classroom requests TOR 1 as well as TOR 2. While the usage of TOR 1 lowers the number of classrooms available in a period, the consideration of TOR 2 guarantees that the total number of biology rooms and classrooms is corrected as well.

Note that kF equals $\frac{1}{2}$ if the duration of the block in the corresponding mode is 2. Neither English nor Mathematics are main subjects and thus there is no request for the resource kO.

- Resource availabilities (cf. Table 8)

Table 8: Resource Availabilities

		$\lvert\mathcal{P}_\pi\rvert = 1$								$\lvert\mathcal{P}_\pi\rvert = 8$	$\lvert\mathcal{P}_\pi\rvert = 4$	
	π	1	2	3	4	5	6	7	8	9	10	11
KL	1					0						
	2	0										
KK	(1,1)											3
KS	(1,1)				0				0			
	(1,2)				0				0			
	(1,3)				0				0			
	(1,4)				0				0			
KR	1	2	2	2	2	2	2	2	2	16	8	8
	2	3	3	3	3	3	3	3	3	24	12	12
KF	(1,1)										1	1
	(1,2)										1	1
	(2,1)										1	1
	(2,2)										1	1
KU	(1,1)										2	2
KO	(1,1)										3	3
	(1,2)										3	3
	(1,3)										3	3
	(1,4)										3	3

Note that in Table 8 only those entries are reproduced which are unequal $\lvert\mathcal{P}_\pi\rvert$. Each subject may be given at most once per day; thus KF = 1 in the SOPs 10 and 11. The entries for KO are redundant since pupils 1 to 4 have no main subjects (cp. kO in Table 7).

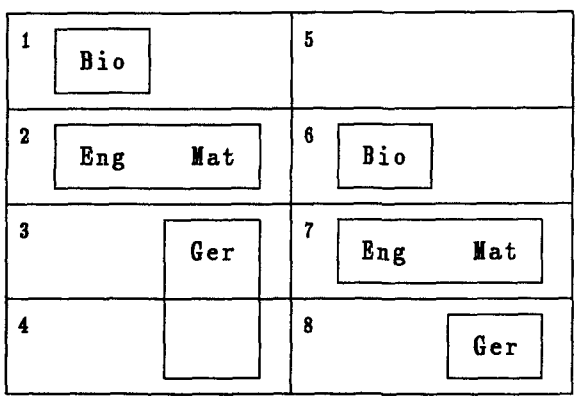
Now, all the parameters of the example are instantiated. Clearly, in the following the transformations outlined in Subsection 2.3 have to be performed. For the sake of completeness the transformations are reproduced in the Appendix.

Figure 3 presents a feasible and optimal timetable for the example. The small numbers denote the periods 1,...,4 (Monday) and 5,...,8 (Tuesday). The abbreviations in the boxes

represent the blocks to be given. Obviously, there is no need for explaining the abbreviations used.

Biology is processed in mode 2, English and Mathematics are processed in mode 2 as well, and Germany is processed in mode 1. It is easy to verify that all the constraints are satisfied. Note that the jobs scheduled in the "left column" of each day require teacher 1, while the "right" ones require teacher 2, respectively. The subsidiary subjects English and Mathematics are scheduled in parallel and given by both teachers simultaneously. Biology is scheduled to be given in a classroom, because on Tuesday no other room is available.

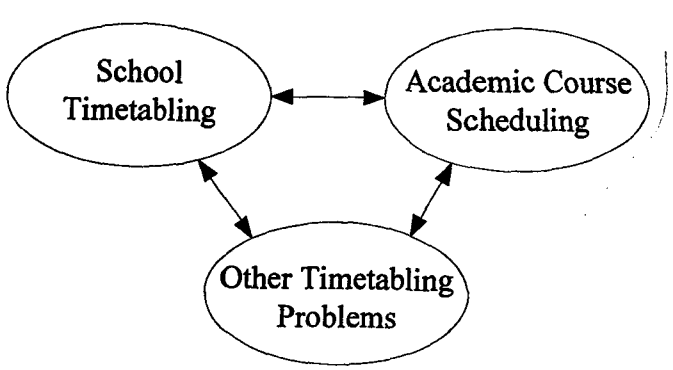
Figure 3: Timetable for the Instance



4. Course Scheduling vs. Timetabling

In literature, usually no clear distinction is made between timetabling and course scheduling. There seems to be confusion (or at least ambiguity) on the different topics of the former and the latter. Clearly, it is really not sufficient to define the borderline between both categories as a function of the problem size (as some authors do). This section aims in pointing out the differences between three important problem categories, i.e. school timetabling, academic course scheduling, and other closely related timetabling problems, respectively (cf. Figure 4).

Figure 4: Three Important Problem Categories



To start with, we compile the issues which are a "must" for any of the three problem categories, at least from our point of view. Then, we give a list of issues, which are optional. Note, that it is sufficient to use quite general terms like "job" (for teaching units, courses, etc.) and "item" (for classes, teachers, rooms, etc.) at this level of abstraction. Moreover, it does not matter whether the respective issue is dealt with as part of the constraints or as part of the objective function in a penalty approach, because this is usually done w.r.t. a specific methodology.

- *Necessary Issues*

  Job completion, no overlaps of items, item availability, preassignments.

- *Optional Issues*

  Room categories, parallelism of jobs, precedence constraints between jobs, changeover or setup times, workload constraints, room preparing constraints, breaks, preferences for rooms and/or hours, distribution requirements, compactness constraints, etc.

Now, by definition the distinction between the problem categories mentioned above must be done w.r.t. the presence/absence of the optional issues only. In addition to the (small) list of necessary issues our experience requires a few optional issues only as a prerequisite for the clustering of a specific approach. To wit:

- *School timetabling*

  The timetable must satisfy compactness constraints (rigorously) for pupils and (modestly) for teachers. In addition, the large teaching units have to be distributed over the week in order to get acceptable timetables. Other optional issues might be accommodated in specific situations.

- *Academic course scheduling*

  In contrast, at universities there do not exist compactness constraints, neither for students nor for professors. In addition, the teaching units per week are not that large that they have to be split into smaller peaces of work. But, a timetable for universities will only be acceptable if and only if it takes care of the professors' preferences for rooms and/or hours (at least in Germany).

- *Other timetabling problems*

  No general guideline can be established. It depends on the specific timetabling problem under consideration which optional issues become necessary.

Clearly, this distinction is not the same which has been provided by other authors. It is stated e.g. by de Werra 1985 that we are confronted with academic course scheduling when there is no fixed curriculum for the students. The problem setting provided in this paper definitely shows that there must be other issues which allow to discriminate between school timetabling and academic course scheduling.

Based on this classification and in addition to the references already cited in Section 1, a short survey of related work will be given as follows (cp. the bibliography provided by Schmidt and Ströhlein 1980):

- *School timetabling*

  Aubin and Ferland 1989 deal with a "large scale" timetabling problem. Abramson 1991 considers pupils, teachers, subjects, and rooms as relevant items, yet in a more restricted form than we do. For the solution of the model he suggests a simulated annealing procedure. Cangalovic and Schreuder 1991 consider the case that lectures of different lenghts have to be scheduled and they provide an exact graph colouring algorithm.

- *Academic course scheduling*

  Tripathy 1980 considers a course scheduling problem, which does not allow to take care of rooms of different sizes, but which considers the professors' preferences for periods. Ferland and Roy 1985 consider two subproblems; roughly speaking, one coordinates the temporal requirements, whereas the other looks at the requirements for rooms. An interactive man-machine approach is described in Mulvey 1982. Dyer and Mulvey 1976 and McClure and Wells 1984 consider the problem of assigning professors to subjects, wheras Shih and Sullivan 1977 and Dinkel, Mote and Venkataramanan 1989 in addition compute a timetable, yet in a rather restricted way. A general model is described in Heinrich 1984 and Bettin 1986. Approaches based on graph colouring may be found in de Werra 1985, Hertz 1991, 1992, and Kiaer and Yellen 1992. Dowsland 1990 models and solves a problem where the disappointments of the students have to be minimized.

- *Other timetabling models*

  In this category the farmost relevant problem is the examination scheduling problem which has been dealt with in, e.g., White and Chan 1979, Mehta 1981, Carter 1986, and Balakrishnan, Lucena, and Wong 1992, respectively. A college timetabling problem is considerd in Tripathy 1980.

5. Summary and Outline of Future Work

In this paper we propose a new model for timetabling which addresses most of the items which are relevant for applications: No class, subject, room, and teacher overlaps, are allowed; one has to take care of class, subject, room, and teacher availability; compactness constraints have to be regarded; preassignment of lectures must be possible; some lectures have to be given simultaneously; lunch breaks have to be observed, etc. Most of the constraints mentioned have been modelled using the unifying framework of partially renewable resources. Lessons of different lengths are distributed by the use of the so-called multiple mode concept with mode identity constraints.

Figure 5: Systems Survey and Synthesis

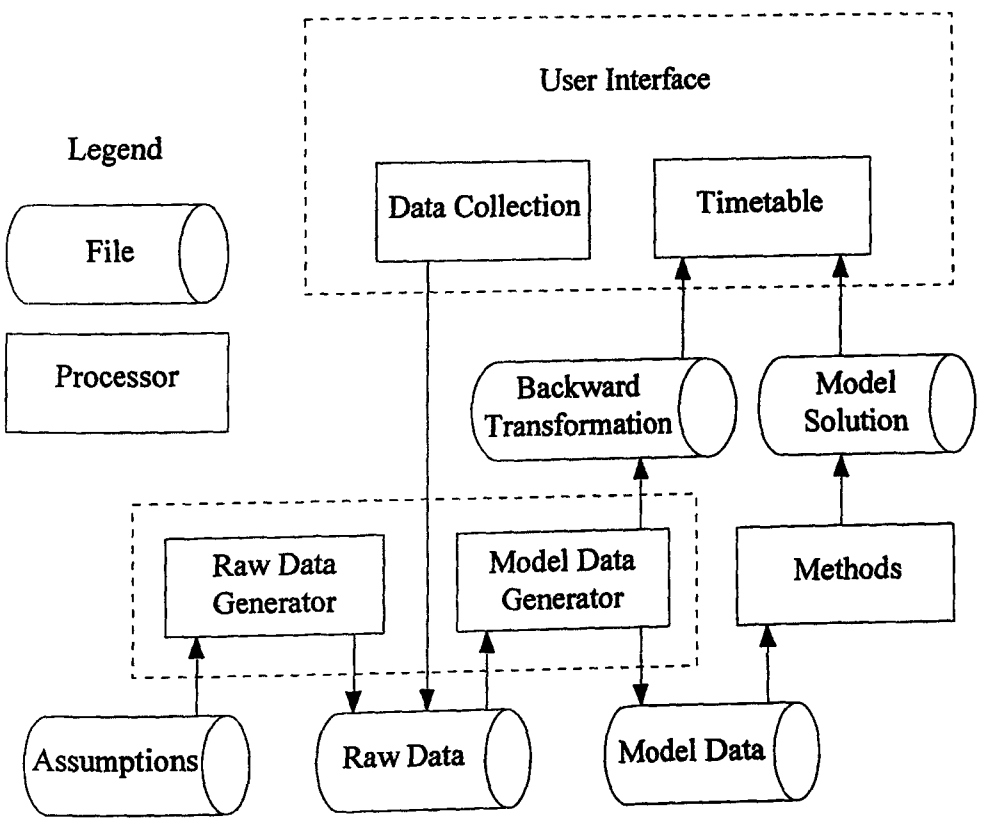


Figure 5 gives an overview of the system built so far, i.e. it shows what we already did and what we still have to do. The instance generator reads the assumptions from an input file. The raw data generator which is part of the instance generator creates an output file in terms of Subsection 2.2 which is directly interpretable by the user. Then, the raw data are transformed into the model data. They are used as input for methods, i.e. a modelling language which creates the MPS-file for a general purpose MIP-solver, as well as for special purpose methods. Besides the model data file, the model data generator creates an information file (for backward transformation). This file and the model solution file allow to calculate a timetable which presents all the information not in terms of the (internal) model,

but in terms of the specific application under consideration. So far, we did not code the user interface, although this clearly is a dominating prerequisite for the systems use.

In part II of this paper, we present greedy randomized and genetic algorithms. We provide an instance generator for the generation of a representative set of instances. The generator along with a statistical model is used for a thorough experimental evaluation of the methods.

Appendix: Transformations for the Example of Subsection 3.2

First, we relabel the BTUs $\{(h,f)\} = \{(1,1),(1,2),(2,1),(2,2)\}$ to the $U = 4$ BTUs 1, 2, 3, and 4, respectively. Thus, we get $\mathcal{G}_3 = \{4\}$ and $\mathcal{G}_4 = \{3\}$. BTUs 3 and 4, which have to be processed in parallel, establish the new BTU 3 with $c_{3m} = (200,6,2)$ for $m = 0,1,2$. Table A.1 reproduces part of the resource usages after the transformations (new entries are underlined).

Table A.1: Resource Usages with Parallel BTUs

	u	1			2		3	
	b	1	2	3	1	2	1	2
	m	1 2	1 2	1 2	1 2	1 2	1 2	1 2
kL	1				1 1	1	1 1	1
	2	1 1	1 1	1			1 1	1
	⋮							
kR	1	1 1	1 1	1			2 2	2
	2	1 1	1 1	1	1 1	1	2 2	2
kF	1	½ 1	1 1	1				
	2				½ 1	1		
	3						½ 1	1
	⋮							

Now, the blocks of the BTUs are transformed to seven jobs. Finally, the resources are transformed to $R = 17$ resources. (Note, that we now have 3 instead of 4 resources KF.) For the sake of completeness in the following all transformed parameters are reproduced.

$T = 8$ number of periods

$\Pi = 11$ number of period subsets

$\mathcal{P}_i$ period subsets

$\mathcal{P}_1 = \{1\}, \ldots, \mathcal{P}_8 = \{8\}, \mathcal{P}_{10} = \{1,\ldots,4\}, \mathcal{P}_{11} = \{5,\ldots 8\}$

$\mathcal{P} = \{10,11\}$ set of SOPs

U = 3 number of BTUs

J = 7 number of jobs

a_u, e_u first, last job of BTU u

$a_1 = 1 \quad a_2 = 4 \quad a_3 = 6$ (first job)

$e_1 = 3 \quad e_2 = 5 \quad e_3 = 7$ (last job)

R = 17 number of resources

M_u number of modes of BTU u

$M_1 = 3, M_2 = 2, M_3 = 2$

d_{jm} duration of job j in mode m (cf. Table A.2)

Table A.2: Mode Dependent Job Durations d_{jm}

d_{jm}	j	1	2	3	4	5	6	7
m	0	0	0	0	0	0	0	0
	1	2	1	0	2	0	2	0
	2	1	1	1	1	1	1	1

c_{um} costs of BTU u in mode m

$c_{1m} = (100,1,3), c_{2m} = (100,3,1), c_{3m} = (200,6,2)$

$\mathcal{M}_{jm}$ set of forbidden periods

$\mathcal{M}_{1,1} = \{1,5\}, \mathcal{M}_{4,1} = \{1,5\}, \mathcal{M}_{6,1} = \{1,5\}$

$\mathcal{E}_u$ set of BTUs with minimum time lags

$\mathcal{E}_2 = \{1\}$

σ_{uv} minimum time lags

$\sigma_{2,1} = 1$

The BTUs, jobs, modes, and requested resources are collected in Table A.3. Note that the mode 0 has been omitted, because no resource is required at all.

Table A.3: BTUs u, Jobs j, Modes m, and Requested Resources r

u	1			2		3	
j	1	2	3	4	5	6	7
m	1,2	1,2	1,2	1,2	1,2	1,2	1,2
r	2,3,8, 9,10,13	2,3,8, 9,10,13	2,3,8, 9,10,13	1,3,9, 11	1,3,9, 11	1,2,4, 5,6,7, 8,9,12	1,2,4, 5,6,7, 8,9,12

The transformed resource usages are presented in Table A.4, while Table A.5 provides the transformed resource availabilities.

Table A.4: Resource Usages k_{jmr} after Transformations

	j	1	2	3	4	5	6	7
	m	1 2	1 2	1 2	1 2	1 2	1 2	1 2
	1				1 1	1	1 1	1
	2	1 1	1 1	1			1 1	1
	3	1 1	1 1	1	1 1	1		
	4						1 1	1
	5						1 1	1
	6						1 1	1
	7						1 1	1
r	8	1 1	1 1	1			2 2	2
	9	1 1	1 1	1	1 1	1	2 2	2
	10	½ 1	1 1	1				
	11				½ 1	1		
	12						½ 1	1
	13	1 1	1 1	1				
	14							
	15							
	16							
	17							

Table A.5: Resource Availabilities $K_{r\pi}$ after Transformations

		$\lvert\mathcal{P}_\pi\rvert = 1$								$\lvert\mathcal{P}_\pi\rvert = 8$	$\lvert\mathcal{P}_\pi\rvert = 4$	
	π	1	2	3	4	5	6	7	8	9	10	11
	1					0						
	2	0										
	3											3
	4				0				0			
	5				0				0			
	6				0				0			
	7				0				0			
r	8	2	2	2	2	2	2	2	2	16	8	8
	9	3	3	3	3	3	3	3	3	24	12	12
	10										1	1
	11										1	1
	12										1	1
	13										2	2
	14										3	3
	15										3	3
	16										3	3
	17										3	3

Table A.6 provides a feasible solution in terms of the binary variables x_{jmt}. For the sake of readibility variables $x_{jmt} = 0$ are omitted.

Table A.6: Feasible Solution x_{jmt} ($\forall$ j, $\forall$ m, $\forall$ t)

	j	1		2		3		4		5		6		7	
	m	0	1 2	0	1 2	0	1 2	0	1 2	0	1 2	0	1 2	0	1 2
t	1					1		1							
	2											1			
	3														
	4	1													
	5														
	6									1					
	7													1	
	8			1											

Acknowledgements: The authors are indepted to Stefan Carstens and Harald Hoffmann for providing the illustrative examples and for helpful comments. Andreas Schirmer improved the phrasing substantially.

References

Abramson, D., "Constructing school timetables using simulated annealing: Sequential and parallel algorithms", *Management Science*, Vol. 37 (1991), pp. 98–113.

Aust, R.J., "Improvement algorithms for school timetabling", *Computing J.*, Vol. 19 (1976), pp. 339-343.

Alvarez-Valdes, R., Martin, G., Tamarit, J.M., "Constructing good solutions for a school timetabling problem", Working Paper, University of Valencia, Valencia/Spain 1994.

Aubin, J., Ferland, J.A., "A large scale timetabling problem", *Computers and Operations Research*, Vol. 16 (1989), pp. 67–77.

Balakrishnan, N., Lucena, A., Wong, R.T., "Scheduling examinations to reduce second-order conflicts", *Computers and Operations Research*, Vol. 19 (1992), pp. 353-361.

Bettin, M., "A heuristic algorithm for constructing university timetables on microcomputers", in: *OR Models on Microcomputers*, Coelho, J.D., Tavares, L.V. (eds.), Elsevier, Amsterdam 1986, pp. 105–110.

Böttcher, J., Drexl, A., Kolisch, R., Salewski, F., "Project scheduling under partially renewable resource constraints", Universität Kiel 1996 (in preparation).

Cangalovic, M., Schreuder, J.A.M., "Exact colouring algorithm for weighted graphs applied to timetabling problems with lectures of different lengths", *European J. of Operational Research*, Vol. 51 (1991), pp. 248-258.

Cangalovic, M., Schreuder, J.A.M., "Modelling and solving an acyclic multi-period timetabling problem", *Discrete Applied Mathematics*, Vol. 35 (1992), pp. 177-195.

Carter, M.W., "A survey of practical applications of examination timetabling algorithms", *Operations Research*, Vol. 34 (1986), pp. 193-202.

Cattrysse, D.G., Van Wassenhove, L.N., "A survey of algorithms for the generalized assignment problem", *European J. of Operational Research*, Vol. 60 (1992), pp. 260-272.

Chahal, N, de Werra, D., "An interactive system for constructing timetables on a PC", *European J. of Operational Research*, Vol. 40 (1989), pp. 32-37.

Costa, D., "A tabu search algorithm for computing an operational timetable", *European J. of Operational Research*, Vol. 76 (1994), pp. 98-110.

Dige, P., Lund, C., Raun, H.C., "Timetabling by simulated annealing", in: *Applied simulated annealing.* Lecture Notes in Economics and Mathematical Sciences, Vol. 396, Berlin, Springer 1993, pp. 151-174.

Dinkel, J.J., Mote, J., Venkataramanan, M.A., "An efficient decision support system for academic course scheduling", *Operations Research,* Vol. 37 (1989), pp. 853-864.

Dowsland, K.A., "A timetabling problem in which clashes are inevitable", *J. of the Operational Research Society,* Vol. 41 (1990), pp. 907-918.

Dyer, J.S., Mulvey, J.M., "An integrated optimization/information system for academic departmental planning", *Management Science,* Vol. 22 (1976), pp. 1332-1341.

Even, S., Itai, A., Shamir, A., "On the complexity of timetable and multicommodity flow problems", *SIAM J. on Computing,* Vol. 5 (1976), pp. 691-703.

Fahrion, R., Dollansky, G., "Construction of university faculty timetables using logic programming techniques", *Discrete Applied Mathematics,* Vol. 35 (1992), pp. 221-236.

Ferland, J.A., Roy, S., "Timetabling problem for university as assignment of activities to resources", *Computers and Operations Research,* Vol. 12 (1985), pp. 207-218.

Fisher, M.L., Jaikumar, R., Van Wassenhove, L.N., "A multiplier adjustment method for the generalized assignment problem", *Management Science,* Vol. 32 (1986), pp. 1095-1103.

de Gans, O.-B., "A computer timetabling system for secondary schools in the Netherlands", *European J. of Operational Research,* Vol. 7 (1981), pp. 175-182.

Garey, M.R., Johnson, D.S., *Computers and intractability: a guide to the theory of NP-completeness.* Freeman, San Francisco 1979.

Heinrich, H., "Ein heuristisches Lösungsverfahren zur Stundenplanung und Raumbelegung an Hochschulen", *Angewandte Informatik,* Vol. 26 (1984), pp. 379-388.

Hertz, A., "Tabu search for large scale timetabling problems", *European J. of Operational Research,* Vol. 54 (1991), pp. 39-47.

Hertz, A., "Finding a feasible course schedule using tabu search", *Discrete Applied Mathematics,* Vol. 35 (1992), pp. 255-270.

Junginger, W., "Timetabling in Germany - A survey", *Interfaces,* Vol. 16:4 (1986), pp. 66-74.

Junginger, W., "On representatives of multi-index transportation problems", *European J. of Operational Research,* Vol. 66 (1993), pp. 353-371.

Kang, L., White, G.M., "A logic approach to the resolution of constraints in timetabling", *European J. of Operational Research,* Vol. 61 (1992), pp. 306-317.

Kiaer, L., Yellen, J., "Weighted graphs and university course timetabling", *Computers and Operations Research,* Vol. 19 (1992), pp. 59-67.

Kitagawa, F., Ikeda, H., "An existential problem of a weight-controlled subset and its application to school timetable construction", *Discrete Mathematics,* Vol. 72 (1988), pp. 195-211.

Kubale, M., Jackowski, B., "A generalized implicit enumeration algorithm for graph colouring", *Communications of the ACM,* Vol. 28 (1985), pp. 412-418.

Lawrie, N.L., "An integer linear programming model of a school timetabling problem", *Computing J.,* Vol. 12 (1969), pp. 307-316.

McClure, R.H., Wells, C.E., "A mathematical programming model for faculty course assignments", *Decision Sciences,* Vol. 15 (1984), pp. 409-420.

Mehta, N.K., "The application of a graph coloring method to an examination scheduling problem", *Interfaces,* Vol. 11:5 (1981), pp. 57-64.

Mulvey, J.M., "A classroom/time assignment model", *European J. of Operational Research,* Vol. 9 (1982), pp. 64-70.

Osterman, R., de Werra, D., "Some experiments with a timetabling system", *OR Spektrum,* Vol. 3 (1982), pp. 199-204.

Papoulias, D.B., "The assignment-to-days problem in a school timetable - a heuristic approach", *European J. of Operational Research,* Vol. 4 (1980), pp. 31-41.

Salewski, F., Schirmer, A., Drexl, A., "Project scheduling under resource and mode identity constraints", Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel, No. 387 (Part I: Model, complexity status, and methods), No. 388 (Part II: An application to audit-staff scheduling) (1996).

Schmidt, G., Ströhlein, T., "Timetable construction - An annotated bibliography", *The Computer J.,* Vol. 23 (1980), pp. 307-316.

Shih, W., Sullivan, J.A., "Dynamic course scheduling for college faculty via zero-one programming", *Decision Sciences,* Vol. 8 (1977), pp. 711-721.

Tripathy, A., "A Lagrangean relaxation approach to course timetabling", *J. of* the *Operational Research Society,* Vol. 31 (1980), pp. 599-603.

White, G.M., Chan, P.W., "Towards the construction of optimal examination schedules", *INFOR,* Vol. 17 (1979), pp. 219-229.

de Werra, D., "An introduction to timetabling", *European J. of Operational Research,* Vol. 19 (1985), pp. 151-162.