JOURNAL OF SCHEDULING
J. Sched. 2002; **5**:171–183 (DOI: 10.1002/jos.097)

Complexity of some special types of timetabling problems

D. de Werra[1,*,†], A. S. Asratian[2] and S. Durand[3]

[1]*Ecole Polytechnique Fédérale de Lausanne, CH 1015 Lausanne, Switzerland*
[2]*Lulea University of Technology, S 97187 Lulea, Sweden*
[3]*Laboratoire d'Informatique, Université d'Avignon, BP 1228, F 84911 Avignon, France*

SUMMARY

Starting from the simple class-teacher model of timetabling (where timetables correspond to edge colorings of a bipartite multigraph), we consider an extension defined as follows: we assume that the set of classes is partitioned into groups. In addition to the teachers giving lectures to individual classes, we have a collection of teachers who give all their lectures to groups of classes. We show that when there is one such teacher giving lectures to three groups of classes, the problem is NP-complete. We also examine the case where there are at most two groups of classes and we give a polynomial procedure based on network flows to find a timetable using at most t periods. Copyright © 2002 John Wiley & Sons, Ltd.

KEY WORDS: timetabling; edge coloring; chromatic index; school timetabling; chromatic scheduling

1. INTRODUCTION

Among the various models built for dealing with school timetabling problems, graph coloring have been extensively used (see References [1–11]). We shall consider here a model which extends the basic 'class-teacher model' and which corresponds to some situations which occur frequently in the basic training programmes of universities and of schools; such a model has been discussed in Reference [4]. Essentially it consists in introducing in addition to the so-called individual lectures (given by one teacher to one class) some group-lectures (given by one teacher to a group of classes). Here, we shall assume that the set of classes is partitioned into a collection of p groups; this means that each class belongs to exactly one group and we allow the possibility of having groups containing exactly one class.

After presenting the basic graph model, we shall discuss the special case where there is exactly one teacher whose lectures are all group-lectures (all the other teachers have only individual lectures). Section 3 will be devoted to the complexity of this problem. Connections with the open shop scheduling problems will be underlined and related complexity results will be derived.

*Correspondence to: D. de Werra, Département de Mathématiques, Chaire de Recherche Opérationnelle, EPFL, CH-1015 Lausanne, France
†E-mail: dewerra@dma.epfl.ch

In Section 4, the situation will be examined where the set of classes is partitioned into two groups and in addition some teachers give only group-lectures while the others give only individual lectures. A network flow algorithm will be designed for constructing a timetable using no more than a given number t of periods whenever such a timetable does exist.

All terms and notations of graph theory not defined here can be found in Reference [12]. The reader is referred to [13] for concepts related to complexity.

2. THE BASIC MODEL

The 'class-teacher' model of timetabling is defined as follows: we are given a set τ of m teachers $T_1, \ldots, T_m$ and a set $\mathscr{C}$ of n classes $C_1, \ldots, C_n$ (here a class is a group of students who follow exactly the same programme). A set $\mathscr{B}$ of lectures is given by an $(n \times m)$ *requirement matrix* $B = (b_{ij})$ where b_{ij} is the number of lectures to be given by teacher T_j to class C_i. We assume that all lectures have the same length (say, one period). A *timetable* (using t periods) is an assignment of each lecture to some period in $\{1, \ldots, t\}$ such that at each period no teacher (resp. no class) is involved in more than one lecture.

To this problem one may associate a bipartite multigraph $H = (\mathscr{C}, \tau, \mathscr{B})$: the 'left set' $\mathscr{C}$ of nodes corresponds to the classes, the 'right set' τ to the teachers. The set $\mathscr{B}$ of edges is obtained by introducing b_{ij} parallel edges between node C_i and node T_j for all i, j.

A *t-coloring* (of the edges) of H is an assignment of one color in $\{1, \ldots, t\}$ to each edge of H in such a way that no two adjacent edges have the same color. It is then clear that there is a one-to-one correspondence between timetables using t periods and t-colorings of H: an edge $[C_i, T_j]$ gets color k if and only if teacher T_j gives a lecture to class C_i at period k.

It is well-known [12] that a t-coloring of a bipartite multigraph H exists if and only if $t \geqslant \Delta(H)$ where $\Delta(H)$ is the maximum degree of the nodes in H, i.e.

$$\Delta(H) = \max\left(\max_i \sum_j b_{ij}, \max_j \sum_i b_{ij}\right)$$

All lectures given by one teacher to one class will be called *individual lectures* in the remainder of the paper.

In some educational institutions, it happens that some classes are grouped together in order to take some lectures given simultaneously to all classes of the group by one teacher; as discussed in Reference [4], this is often the case for some basic courses.

We shall assume here that a collection $\mathscr{G}$ of groups $G_1, \ldots, G_p$ of classes is given such that each class belongs to at most one group. If we allow groups containing exactly one class, then we may w.l.o.g consider that we have a partition $\mathscr{G}$ of the set $\mathscr{C}$ of classes into p (disjoint) groups $G_1, \ldots, G_p$.

In addition to the set $\mathscr{B}$ of individual lectures, we are given a set $\mathscr{A}$ of so-called *group-lectures* which is specified by a $(p \times m)$ requirement matrix $A = (a_{\ell j})$ where $a_{\ell j}$ is the number of group-lectures (lasting one period each) which teacher T_j must give to group G_ℓ.

A timetable using t periods is an assignment to some period in $\{1, \ldots, t\}$ of each lecture in $\mathscr{A} \cup \mathscr{B}$ such that:

no class (resp. no teacher) is involved in more than one lecture at a time

Such a model is called a *university timetable* model; clearly if $a_{\ell j} = 0$ for all ℓ, j or if all groups contain exactly one class, then we have the class-teacher model. It was shown in Reference [4] that deciding whether there exists a timetable in t periods in the university timetable model is NP-complete. It is furthermore the case even if $t = 3$ and $p = 4$.

We will in fact consider now a special case of the university timetable model. The set τ of teachers will be partitioned into two subsets $\mathscr{P} = (P_1, \ldots, P_r)$ and $L = (L_1, \ldots, L_s)$; each P_i has to give group-lectures but not individual lecture, while each L_i has to give only individual lectures, but no group-lecture. We shall call each P_i a *professor* and each L_i a *lecturer*, so that the special model of university timetable will be called the *professor–lecturer* model.

The above model is indeed much more general than it actually looks. One should first observe that it occurs whenever for practical reasons several classes are grouped together and attend the same lecture in the same classroom.

As mentioned above, this situation arises in many schools and universities where one has different programmes offered (each programme corresponds to one class) and for some common lectures, the students of several classes are brought together (it is in particular the case for courses in basic sciences, like mathematics, physics, etc.).

In our model, these courses are given by professors, the individual lectures corresponding to the specific programmes are given by the so-called lecturers.

But there is also a case where in each programme there are options; each student has to take one out of several topics. We group together the classes to which the same options are proposed and we split the resulting group of students into subgroups (new classes) corresponding to the different topics (which are offered simultaneously). This would correspond to a group lecture (involving several professors which can be merged into one fictitious professor since the options are always offered simultaneously).

Typically in a school or in a university the number of groups need not be very large; at EPFL for instance one may consider that the twelve orientations are organised in essentially 2 or 3 groups for the basic courses.

In addition, in some French autonomous university institutions (Instituts Universitaires de Technologie, engineering schools), the curriculum is often divided into 2 or 3 years: All the students of the same year have group-lectures and optional lectures.

We shall also mention later some connections with preemptive open shop scheduling (see Reference [14] for basic definitions).

3. A SPECIAL CASE OF THE PROFESSOR–LECTURER MODEL

We shall now discuss the complexity of the professor–lecturer model. It was shown in Reference [4] that it is NP-complete to decide whether a timetable using t periods exists for the professor–lecturer model.

Before stating a basic complexity result, let us recall some definitions and properties. In a bipartite multigraph $G = (V, W, E)$ a *V-sequential coloring* is an edge coloring of G such that for each node v in V the edges adjacent to v have colors $1, 2, \ldots, d_G(v)$. Let VS be the problem of existence of a V-sequential coloring in a bipartite multigraph G.

VS was shown to be NP-complete in Reference [15] even if $\Delta(G) = 3$.

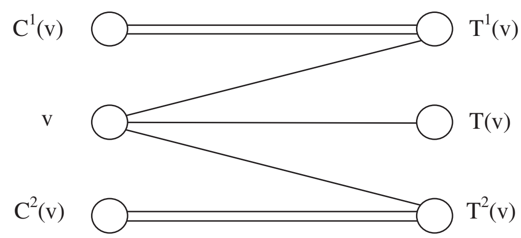

Figure 1. The construction of a PL model.

Proposition 3.1
It is NP-complete to decide whether there exists a timetable using t periods in the professor–lecturer model, even if there are 3 groups and $t=3$.

Proof
We will show that VS can be reduced to a professor–lecturer model with 3 groups and with $t=3$.

Let $G=(\mathscr{C},\mathscr{T},\mathscr{B})$ be a bipartite multigraph with $\Delta(G)=3$; for each node v in $\mathscr{C}$ with $d_G(v)=1$ let $T(v)$ be its neighbor in G; we introduce new nodes $T^1(v)$, $T^2(v)$, $C^1(v)$, $C^2(v)$ and edges $[v,T^1(v)]$, $[v,T^2(v)]$, $[C^1(v),T^1(v)]^1$, $[C^1(v),T^1(v)]^2$, $[C^2(v),T^2(v)]^1$, $[C^2(v),T^2(v)]^2$ (see Figure 1).

Let $K=(\mathscr{M},\mathscr{N},\mathscr{O})$ be the resulting multigraph; it is still bipartite and $\Delta(K)=3$.

$$
\begin{aligned}
\text{Let } G_1 &= \{C^1(v)|v\in\mathscr{C} \text{ and } d_G(v)=1\}\cup\{v|v\in\mathscr{C} \text{ and } d_G(v)=2\}\\
G_2 &= \{C^2(v)|v\in\mathscr{C} \text{ and } d_G(v)=1\}\\
G_3 &= \{v|v\in\mathscr{C} \text{ and } d_G(v)=1 \text{ or } 3\}
\end{aligned}
$$

Fact 1
G has a $\mathscr{C}$-sequential coloring if and only if K has an edge 3-coloring such that for each node x in G_1, the edges adjacent to x have colors 1 and 2, and for each node x in G_2 the edges adjacent to x have colors 1 and 3.

Proof

(A) If G has a $\mathscr{C}$-sequential coloring, then each edge $[v,T(v)]$ (where v has $d_G(v)=1$) has color 1; so we may color $[v,T^1(v)]$ with 3 and $[v,T^2(v)]$ with 2. Then the two parallel edges $[C^1(v,T^1(v))]$ are given colors 1 and 2 and the two parallel edges $[C^2(v,T^2(v))]$ are given colors 1 and 3. Repeating this for each v with $d_G(v)=1$ gives the required coloring of K (since in addition for each v with $d_G(v)=2$, the colors on adjacent edges to v are 1 and 2).

(B) Conversely assume we have an edge 3-coloring of K which satisfies the conditions on G_1, G_2; we show that after removal of all edges introduced to transform G into K, the resulting coloring of G is $\mathscr{C}$-sequential coloring. Clearly all nodes v with $d_G(v)=2$ are in G_1, so the colors on edges adjacent to v are 1 and 2.

Consider node v with $d_G(v)=1$; since the edges between $C^1(v)$ and $T^1(v)$ have colors 1 and 2, the edges $[v, T^1(v)]$ has color 3. Similarly the edges between $C^2(v)$ and $T^2(v)$ have colors 1 and 3, so $[v, T^2(v)]$ has color 2; this forces $[v, T(v)]$ to have color 1.

So we have a $\mathscr{C}$-sequential coloring of G. □

Consider now the professor–lecturer model defined from K as follows: $\mathscr{M}$ is the set of classes which is divided into 3 groups G_1, G_2, G_3 as defined above. $\mathscr{N}$ is the set of lecturers and the collection of individual lectures is given by the set $\mathscr{O}$ of edges of K.

In addition, define a (3×1)—matrix $A=(a_{lj})$ giving the number a_{lj} of group lectures which professor P_j must give to group G_1 by $a_{11}=a_{21}=1$; $a_{31}=0$.

Fact 2

For the above professor–lecturer model there is a timetable in $t=3$ periods if and only if K has an edge 3-coloring such that for each node v in G_1 (resp. G_2) the colors on edges adjacent to v are 1 and 2 (resp. 1 and 3).

Proof

(A) If there is a timetable for the professor–lecturer model then by permuting the periods if necessary, we have a timetable where G_1 has its group lecture at period 3 and G_2 at period 2.
So the individual lectures of this timetable give an edge coloring of K where for each node v in G_1 (resp. G_2) the edges adjacent to v have colors 1 and 2 (resp. 1 and 3).

(B) Conversely assume we have an edge 3-coloring of K satisfying the requirements for all nodes in G_1 and G_2.
Then this coloring gives a timetable for the individual lectures; the group lecture of G_1 (resp. G_2) is then scheduled at period 3 (resp. 2). □

So we have reduced VS to a professor–lecturer model with $t=3$ periods and 3 groups and Proposition 3.1 is proved. □

We should also observe that the above model can be viewed differently; it is a special case of restricted timetable problem (RTT) which was shown to be NP-complete in Reference [16].

For sake of simplicity we interchange the role of classes and teachers in the formulation of [16]: sets $\mathscr{C}$ of classes and $\mathscr{T}$ of teachers are given. We have a set $\mathscr{B}$ of lectures given by a requirement matrix $B=(b_{ij})$ with $b_{ij}=0$ or 1 for all i, j.

The number of periods is $t=3$ and in addition for each class C_i we are given the set $C_i^*\subseteq\{1,2,3\}$ of periods where C_i is available for lectures, we may even assume $2\leqslant|C_i^*|\leqslant 3$ and $|C_i^*|=\sum_j b_{ij}$.

One may observe that since b_{ij} is 0 or 1, no class ever meets the same teacher at different periods. So we may interchange the role of periods and of teachers: C_i^* will be the set of teachers who must give a lecture do C_i.

The periods where the lectures of C_i can occur is given by $T_i^*=\{j\,|\,b_{ij}=1\}$. We obtain a problem P* which has a solution if and only the initial problem RTT has a solution: assume we have a selection of RTT and consider that G_i meets teacher T_j in T_i^* at period k in C_i^*; for P* we will get a timetable by setting:

C_i meets teacher k in C_i^* at period T_j in T_i^*. It is a timetable, because there is no G_s $(s \neq i)$ meeting k at the same period T_j (this would mean that in RTT there are two classes C_s, C_i meeting the same teacher T_j at the same period k) and there is no teacher $1 \neq k$ meeting C_i at the same period T_j (this would mean that T_j gives lectures to C_i at periods 1 and k and this is impossible since $b_{ij} = 0$ or 1).

Conversely if we have a solution to P*, it is easy to derive by a similar argument a solution to RTT.

Hence we can conclude that problem P* is NP-complete: so we can state

Proposition 3.2
The class-teacher timetable problem where each class C_i has a set C_i^* of available periods in which its lectures have to be scheduled is NP-complete even if there are 3 teachers.

We may in addition interpret Proposition 3.1 in terms of open shop scheduling: $\mathscr{T}$ is a set of processors and $\mathscr{C}$ a set of jobs. Each job C_i consists of operations O_{ij} to be processed on processor T_j with an (integer) processing time b_i. Preemptions (interruptions) are allowed (it is sufficient to allow them at integral times).

There is a one-to-one correspondence between edge t-coloring of the associated multigraph $G = (\mathscr{C}, \mathscr{T}, \mathscr{B})$ and schedules in t time units: a schedule is an assignment of each operation O_{ij} to b_{ij} time periods in $\{1, \ldots, t\}$ such that no two operations of the same job are assigned to the same period and no processor is working on operations of different jobs at the same time.

Let Q be the preemptive open shop problem with $t = 3$ where we introduce for each job C_i a release time r_i and a deadline d_i; let $G_1 = \{\text{jobs } C_i \text{ with } r_i = 0,\ d_i = 2\}$, $G_2 = \{\text{jobs } C_i \text{ with } r_i = 1,\ d_i = 3\}$ and $G_3 = \{\text{jobs } C_i \text{ with } r_i = 0,\ d_i = 3\}$. We may in addition assume that for each job C_i we have $d_i - r_i = \sum_j b_{ij}$.

Then there is a one-to-one correspondence between the feasible schedules for Q (in $t = 3$ periods) and the edge 3-colorings of the associated multigraph $G = (\mathscr{C}, \mathscr{T}, \mathscr{B})$ where for each node $v \in G_1$ (resp. G_2) the colors on the edges adjacent to v are 1 and 2 (resp. 2 and 3).

The existence of such a coloring in G is NP-complete from Proposition 3.1 (after interchanging colors 2 and 3). So Q is NP-complete and we have

Corollary 3.3
The preemptive open shop scheduling problem with release times and deadlines is NP-complete even when $t = 3$ periods and there is one group of jobs C_i with $r_i = 1$, one group of jobs C_i with $d_i = 2$ and all remaining $r_i = 0$ and $d_i = 3$.

Note: Related results concerning the complexity of open shop scheduling problems where the processing terms b_{ij} are 0 or 1 are given in Reference [9].

We shall now consider another case of professor–lecturer model; in the example used above for establishing the NP-completeness of the model three groups were needed in the proof. We may ask how limitations on the groups can reduce the difficulty of the problem.

Trivially if each group consists of one class, then we are in the classical class-teacher model. On the other hand, what happens when the number of groups is $p = 2$?

We shall show that when $p = 2$ the problem can be solved by means of network flow techniques.

4. A PROFESSOR–LECTURER MODEL WITH TWO GROUPS

As before we are given a professor–lecturer model with sets $\mathscr{C}$ of classes, $\mathscr{G}$ of groups, $\tau=\mathscr{P}\cup\mathscr{L}$ of teachers (partitioned into professors and lecturers), $\mathscr{A}$ of group-lectures and $\mathscr{B}$ of individual lectures. We assume that $\mathscr{G}=\{G_1,G_2\}$, so $p=2$.

The requirement matrix A of group-lectures is a $(2\times r)$-matrix where $r=|\mathscr{P}|$ is the number of professors.

Let us define $u=\sum_j a_{1j}$, $v=\sum_j a_{2j}$ the number of group-lectures for G_1 and for G_2.

Proposition 4.1

If there is a timetable in t periods for the professor–lecturer model with two groups G_1,G_2, there is a timetable in t periods where the lectures of G_1 are scheduled in the first u periods and the lectures of G_2 in the last v periods.

Proof

Suppose we are given a timetable in t periods; we may partition the set of periods as follows:

$$\begin{aligned} H_i &= \{\text{periods in which only group } G_i \text{ is involved in a group-lecture}\} \text{ for } i=1,2\\ H_{12} &= \{\text{periods where } G_1 \text{ and } G_2 \text{ have a group-lecture}\}\\ H_0 &= \{\text{periods where neither } G_1 \text{ nor } G_2 \text{ is involved in a group-lecture}\}\end{aligned}$$

We construct a new timetable by placing consecutively the lectures scheduled at periods in H_1,H_{12},H_0,H_2; notice that if $H_0=\emptyset$ or $H_{12}=\emptyset$ we are done. Suppose $H_0\neq\emptyset$ and $H_{12}\neq\emptyset$. We now have to move the $\alpha=\min(|H_{12}|,|H_0|)>0$ group-lectures given to G_2 during the first periods of H_{12}; we exchange them with the individual lectures given to the classes contained in G_2 during the α last periods of H_0.

Notice that since we have a professor–lecturer model, this exchange is always possible: no lecturer will be involved in more than one (individual) lecture at a time and the same holds for professors.

So we have constructed a timetable with the required property. □

Essentially the timetables satisfying the property of Proposition 4.1 are such that the number of periods where only individual lectures are scheduled is minimum.

This number is given by

$$\max(t-(u+v),0)$$

Let us now define w by $w=t-(u+v)$. The two cases ($w\leqslant 0$, $w>0$) are illustrated in Figure 2. In case (a). $H_0=\emptyset$ while in case (b). $H_{12}=\emptyset$.

Let us now construct a network $N=(X,U)$ with capacities and lower bounds of flow on each arc. Although the algorithm will have to be used only in the case where $w>0$, according to Proposition 4.3, we will formulate it here in a general form without assumption on the sign of w.

We introduce the following nodes: a source s_0, a sink t_0, a node C_i for each class in $\mathscr{C}$; for each lecturer L_j in $\mathscr{L}$ we introduce three nodes L_j,L_j^1,L_j^2.

We link s_0 to each node C_i by an arc (s_0,C_i); also we link L_j^1 and L_j^2 to L_j by arcs (L_j^1,L_j) and (L_j^2,L_j). Furthermore we link each L_j to the sink t_0 by an arc (L_j,t_0). In addition for each

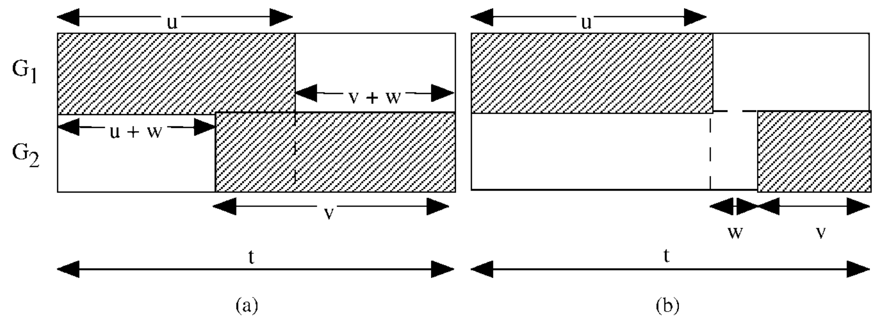

Figure 2. The form of timetables for the two-group case; (a) Case $w \leqslant 0$; and (b) case $W > 0$.

Table I. The network N for the two-group case.

Arcs (x, y)	Lower bound $l(x, y)$	Capacity $\bar{c}(x, y)$
(s_0, C_i)	$\sum_j b_{ij} - \max(0, w)$	$\min(0, w) + \begin{Bmatrix} v & \text{for } C_i \in G_1 \\ u & \text{for } C_i \in G_2 \end{Bmatrix}$
(C_i, L_j^k) $k = 1, 2$	0	1
(L_j^1, L_j)	0	$v + \min(0, w)$
(L_j^2, L_j)	0	$u + \min(0, w)$
(L_j, t_0)	$\sum_i b_{ij} - \max(0, w)$	∞
	$w = t - (u + v)$	

individual lecture involving class C_i and teacher L_j we introduce an arc (C_i, L_j^k) if $C_i \in G_k$. The capacities $\bar{c}(x, y)$ and the lower bounds of flow $l(x, y)$ for each arc (x, y) of N are given in Table I. The construction is illustrated in Figure 3.

The reader is referred to [17] for definitions and properties of flows in networks. We can now state:

Proposition 4.2
There exists a timetable in t periods for the two-group case of the professor–lecturer model if and only if there exists a compatible flow from s_0 to t_0 in the associated network N.

Proof

(A) Assume first that we have obtained a compatible flow f in N and let us show how to obtain a timetable in t periods.
The arcs (C_i, L_j^1) with $f(C_i, L_j^1) = 1$ define a bipartite partial graph H' of H with maximum degree at most v; similarly the arcs (C_i, L_j^2) with $f(C_i, L_j^2) = 1$ define a bipartite partial graph H'' of H with maximum degree at most u.
The edges of H'' represent the individual lectures which must be scheduled in the first u for the classes not included in G_1. These lectures occur during the first u group-lectures of G_1.

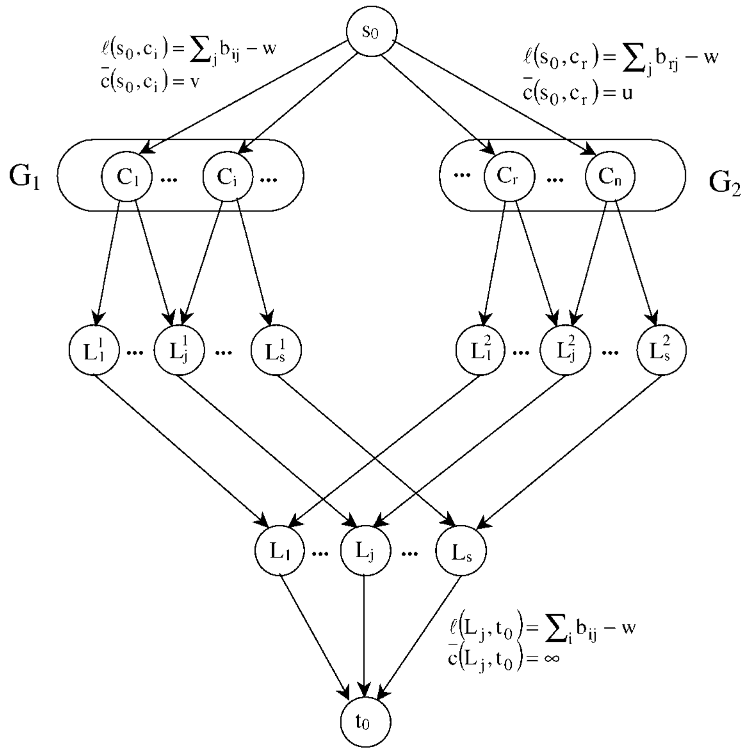

Figure 3. The network associated to the 2-group case.

Similarly the edges of H' represent the individual lectures to be scheduled during the last v periods for the classes not included in G_2. Those lectures are scheduled during the last v group-lectures of G_2.

Since the flow f is compatible the arcs (C_i, L_j^k) with $f(C_i, L_j^k) = 0$ and $1 \leqslant k \leqslant 2$ form a bipartite partial subgraph H^* with maximum degree at most w.

So if $w > 0$, these meetings can be scheduled in the $w = t - (u + v)$ periods where no group-lectures are scheduled. If $w \leqslant 0$, H^* has no edges at all and the case can be handled in a similar way.

(B) If a timetable in t periods exists, then we may according to Proposition 4.1 transform it in such a way that it has the required form; it is then easy to define a flow f which is compatible in the network N.

To every individual lecture (C_i, L_j) given to a class C_i of group G_2 in period $1 \leqslant u + \min(0, w)$ we associate a flow $f(C_i, L_j^2) = 1$ and to every individual lecture (C_i, L_j) given to a class C_i of group G_1 in period $1 > u + \max(0, w)$ we associate a flow $f(C_i, L_j^1) = 1$. It is easy to verify that this will define (by extending the flow on arcs (s_0, C_i), (L_j^k, L_j), (L_j, t_0)) a compatible flow f from s_0 to t_0 in N. □

In fact we can give in a more explicit way the minimum number $t_{\min}$ of periods which are needed for a timetable in the two-group case of the professor–lecturer model.

Let us first define B_i as the submatrix of B generated by the rows corresponding to the classes in group G_i $(i=1,2)$. Let $r(D)$ (resp. $c(D)$) be the maximum row sum (resp. column sum) of a matrix D. Finally let $q(D)=\max\{r(D),c(D)\}$; these values will be needed for $D=A,B,B_1$ and B_2.

Now we observe that $t_{\min}\geqslant\max\{u+q(B_1),v+q(B_2),q(A)\}$.

Proposition 4.3
The minimum number $t_{\min}$ of periods in a timetable for the two-group case of the professor–lecturer model can be determined as follows:

(1) if $q(A)\geqslant\max\{u+q(B_1),\ v+q(B_2)\}$, then $t_{\min}=q(A)$
(2) if $q(A)<\max\{u+q(B_1),\ v+q(B_2)\}$ and we have either $v\geqslant q(B_1)$ or $u\geqslant q(B_2)$, then $t_{\min}=\max\{u+q(B_1),\ v+q(B_2)\}$
(3) if $q(A)<\max\{u+q(B_1),v+q(B_2)\}$, $v<q(B_1)$ and $u<q(B_2)$, then $t_{\min}>u+v$; it is the smallest value of t for which a feasible flow can be found as in Proposition 4.2 (case $w>0$).

Proof
We may assume w.l.o.g that $u+q(B_1)\geqslant v+q(B_2)$. We construct first a timetable of length $q(A)$ for professors (the group-lectures of G_1 are in the first u periods, the ones of G_2 in the last v periods) and we construct a timetable for classes in G_1 using periods $u+1,\ldots,u+q(B_1)$.

If we are in case (1), then $q(A)\geqslant u+q(B_1)$. We construct a timetable for classes in G_2 using periods $1,2,\ldots,q(B_2)$.

Since $q(A)\geqslant\max\{u+q(B_1),v+q(B_2)\}$, we have a timetable in $q(A)$ periods.

If we are in case (2), then $q(A)<u+q(B_1)$; we construct a timetable in $u+q(B_1)$ periods; the group-lectures of G_2 are scheduled in the last v periods, i.e. periods $u+q(B_1)-v+1,\ldots,u+q(B_1)$.

Then we construct a timetable for classes in G_2 using periods $1,2,\ldots,q(B_2)$; since we have either $u\geqslant q(B_2)$ or $v\geqslant q(B_1)$, we have in both situations a timetable in $u+q(B_1)$ periods.

Finally in case (3), we have $q(A)<\max\{u+q(B_1),v+q(B_2)\},v<q(B_1),u<q(B_2)$, then clearly $t_{\min}>u+v$ and we are in the case $w=t-(u+v)>0$ of the network flow algorithm for deciding whether there exists a timetable in t periods. □

Remark 4.4
The network flow algorithm devised for constructing (if it exists) a timetable using t periods may be used to determine the smallest t for which a timetable exists. Starting from a lower bound of the minimum t, we may increase the value of t stepwise until a compatible flow exists in N; changing the value of t will simply modify lower bounds of flow on some arcs; the (incompatible) flow constructed before can be used in an algorithm as a starting solution.

Remark 4.5
When G_1 and G_2 are not disjoint, the same algorithm may be used. In that case, since the classes in $G_\cap=G_1\cap G_2$ must follow all the group-lectures, then $w\geqslant 0$. All we have to do is to check also whether for all C_i in $G_\cap,\Sigma_j b_{ij}\leqslant w$. If there is a feasible flow, then the corresponding lectures can be scheduled during the w periods without group-lectures.

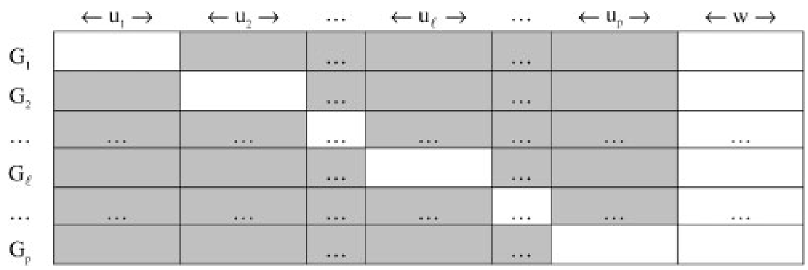

Figure 4. General case solvable with the network flow algorithm.

Remark 4.6
The network flow algorithm described above can be used in more general situations where we may have more than two groups. Suppose that we have a timetable which has the structure presented in Figure 4. The shadowed cells may correspond to group-lectures or periods not available for the classes or the lecturers. All the individual lectures have to be scheduled during the other periods (the set of u_ℓ periods where only the classes of group G_ℓ can have individual lectures (for $\ell = 1, \ldots, p$) and the set of w periods where all classes are available for individual lectures).

Such a situation occurs for example in some institutions where part time students have to follow some given programmes. Besides the weekends (where w periods are available to schedule individual lectures of all classes), each class belongs to some group G_ℓ and so it can be involved in lectures (at most u_ℓ) given on the evening of day ℓ of the week ($\ell \leqslant 5$). Due to the limitation on the number of classrooms available during the week, only one group of classes can get lectures on each day ℓ.

In that case, the network flow model can easily be adapted: We just add p nodes $L_j^1, L_j^2, \ldots, L_j^p$ for each lecturer L_j instead of two. The lower and upper bounds on each arc are accordingly adapted. □

Remark 4.7
The example given in Figure 5 shows that the smallest number of periods for a timetable may be strictly larger than the maximum number of lectures involving a class, a professor or a lecturer. Here the maximum load is 3, but a timetable has at least 4 periods: if we try to construct a timetable in 3 periods, we may w.l.o.g. schedule the group-lecture of $G_1 = C_1$ at period 1 and the group-lectures of $G_2 = \{C_2, C_3, C_4\}$ at periods 2 and 3.

Then lecturer L_1 (resp. L_2) meets C_1 at period 2 (resp. 3); but then at period 1 one cannot schedule the two individual lectures of L_1 to C_2 and C_3.

Remark 4.8
The following example of university timetable problem shows that unless the group-lectures are spread there may be no timetable in t periods; in other words it is not always possible to construct a timetable in the minimum number t of periods if the group-lectures are themselves scheduled within a minimum number of periods.

Here a timetable in $t = 3$ periods exists as shown in Figure 6 but if the group-lecture $G_2 - P_1$ is scheduled at period 2 instead of 3, the individual lectures $C_3 - L_1$ and $C_4 - L_2$ cannot be moved from period 2 to period 3.

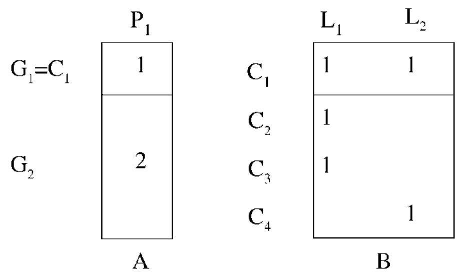

Figure 5. A professor–lecturer model which needs at least 4 periods.

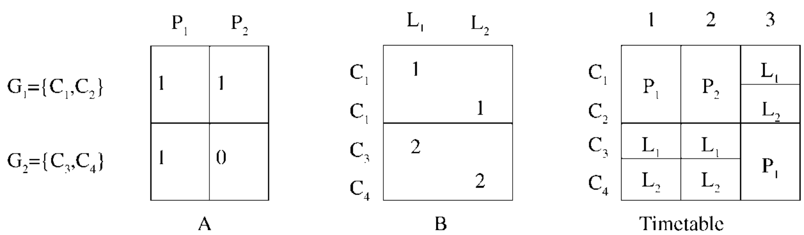

Figure 6. A professor–lecturer model where no timetable in $t = 3$ periods exists in which group-lectures are scheduled in 2 periods.

5. CONCLUSIONS

In this note we have only considered a very simple model for timetabling problems. We should as well include constraints of unavailability; it has been done in some models (see References [10, 11, 18, 19] for instance) but the combination of all these constraints gives difficult problems even in rather simple contexts. Therefore the need to develop efficient heuristic procedures is still obvious. Our purpose here was simply to explore the border between easy and difficult problems.

ACKNOWLEDGEMENTS

This note was written while A.S. Asratian (resp. S. Durand) was visiting the Department of Mathematics of the Ecole Polytechnique Fédérale de Lausanne in March 2000 (resp. in Fall 2000); the support of EPFL is gratefully acknowledged.

The authors would like to thank Tamas Kis for having pointed out some useful references.

REFERENCES

1. Asratian AS, Miruman AN. Transformation of edge colorings of a bipartite multigraph and their applications. *Soviet Mathematics Doklady* 1991; **43**(1):1–3.
2. Asratian AS, Kamalian RR. Investigation on interval edge colorings of graphs. *Journal of Combinatorial Theory Series B* 1994; **62**:34–43.
3. Asratian AS, Denley T, Häggkvist R. Bipartite graphs and their applications. *Cambridge Tracts in Mathematics*, Vol. 131. Cambridge University Press: Cambridge, 1998.
4. Asratian AS, de Werra D. A university timetable problem: complexity and existence theorems, to appear in *European Journal of Operational Research*.

5. Blazewicz J, de Werra D. Some preemptive open shop scheduling problems with a renewable or a non-renewable resource. *Discrete Applied Mathematics* 1992; **35**:205–219.
6. Burke E, Ross P (eds). *Practice and Theory of Automated Timetabling*. Springer: Berlin, 1996.
7. Burke E, Carter M (eds). *Practice and Theory of Automated Timetabling II*. Springer: Berlin, 1998.
8. Hilton AJW. School of Timetables. *Annals of Discrete Mathematics* 1981; **11**:177–188.
9. Kubale M. Open shop problem with zero-one time operations and integer release data/deadline intervals. *Discrete Applied Mathematics* 1997; **76**:213–223.
10. de Werra D. Restricted coloring models for timetabling. *Discrete Mathematics* 1997; **165**/**166**:161–170.
11. de Werra D. The combinatorics of timetabling. *European Journal of Operational Research* 1997; **96**:504–513.
12. Berge C. *Graphes*. Gauthier-Villars: Paris, 1983.
13. Garey MR, Johnson DS. *Computers and Intractability*: *a Guide to the Theory of NP-Completeness*. Freeman: New York, 1979.
14. Blazewicz J, Ecker KH, Pesch E, Schmidt G, Weglarz J. *Scheduling Computer and Manufacturing Processes*. Springer: Berlin, 1996.
15. Asratian AS, Kamalian RR. Interval edge colouring of multigraphs. *Applied Mathematics Yerevan University* 1987; **5**:25–34 (in Russian).
16. Even S, Itai A, Shamir A. On the complexity of timetable and multicommodity flow problems. *SIAM Journal of Computers* 1976; **5**:691–703.
17. Ford LR, Fulkerson DR. *Flows in Networks*. Princeton University Press: Princeton, NJ, 1962.
18. de Werra D. Extensions of coloring models for scheduling purposes. *European Journal of Operations Research* 1996; **92**:474–492.
19. de Werra D. An Introduction to Timetabling. *European Journal of Operations Research* 1985; **19**:151–162.

