An Exact Branch–and–Price Algorithm for Workforce Scheduling

Christoph Stark and Jürgen Zimmermann

Clausthal University of Technology, Department for Operations Research, Julius-Albert-Str. 2, D–38678 Clausthal-Zellerfeld

Abstract. We consider a generic workforce scheduling problem, where employees are characterized by qualifications. Given a set of shifts for each day, we have to determine for each employee his working days as well as a specific shift for each working day. The overall objective is to find a set of feasible schedules with respect to hard and soft restrictions. We propose an integer multi–commodity network flow formulation for the problem under consideration and show how branch–and–price can be used to solve this problem.

1 Introduction

Workforce scheduling is the process of generating work schedules for a number of employees such that an organization can satisfy the demand for its goods or services. We refer to [3] for a recent survey on workforce scheduling. In this paper, we consider a problem which decidedly extends the so-called days-off scheduling problem, where the number of employees to be assigned to different working patterns has to be determined. Given a set of predetermined shifts for each day of the planning horizon, we have to find a schedule for each employee that specifies his days off, his working days as well as a specific shift for each working day. In particular, we consider employees with different qualifications and individual preferences. Moreover, we distinguish between hard restrictions that must be met and soft restrictions which should be obeyed.

Hard restrictions:

1. No employee may be assigned to a shift he is not qualified for.
2. In between two shifts, there must be at least 11 hours of idle time.
3. No employee may work longer than 10 days in a row.
4. Each employee must receive two days off for every 5 days on duty.

Soft restrictions:

1. At least two days off should be scheduled in a row.
2. A day off should be preceded by another day off or an early shift and should be succeeded by another day off or a late shift.
3. A vacation should be preceded by an early shift and succeeded by a late shift.

Our objective is to find a solution which is feasible with respect to the hard constraints and minimizes violations of the soft constraints. Moreover, working at night or on weekends is awarded with significant extra pay. Thus, a long–term objective is to balance gross wages among employees.

2 Problem Formulation

We consider a planning horizon of $t = 1, \ldots, 14$ days. Let J_t be the set of all shifts that take place on day t and define $J := \bigcup_{t=1}^{14} J_t$. J_t contains exactly one dummy shift which is used to admit assignments of superfluous employees to shifts on day t. With each shift $i \in J$, we associate two events s_i and e_i representing the start and end of shift i, respectively. Let set $J^d \subset J$ contain all dummy shifts. A transition between two shifts $i \in J_t$ and $j \in J_{t'}$, $t < t'$, is associated with the events e_i and s_j. For $t' - t > 1$, a transition represents one or more days off. Transitions violating hard restriction 2 are not considered.

2.1 Multi–Commodity Flow Network

For each employee $k \in K$, we consider a network $G^k = (N^k, A^k)$ (cf. Fig. 1). The set of nodes N^k consists of two nodes s_i and e_i for each $i \in J$ that employee k is suitably skilled for as well as a source α and a sink ω. All employees are sufficiently skilled for the dummy shifts. Nodes s_i and e_i associated to already granted holidays or unacceptable shifts for employee k are omitted in N^k. For two nodes $s_i, e_i \in N^k$, A^k contains a directed arc (s_i, e_i). Moreover, A^k contains the arcs (α, s_i) for all $s_i \in N^k$ and (e_i, ω) for all $e_i \in N^k$. Finally, there is an arc (e_i, s_j) for each transition. For each arc $(i, j) \in A^k$, we introduce an arc weight (cost) c_{ij}^k, which serves to penalize violations of the soft restrictions and takes into consideration employee k's preference and eligibility for individual shifts. Observe that a path p_k from α to ω corresponds to a feasible work sequence with respect to hard restrictions 1 and 2. Overlapping networks G^k, $k \in K$, results in a multi–commodity flow network. While the proposed network is structurally similar to the network formulation in [4], we

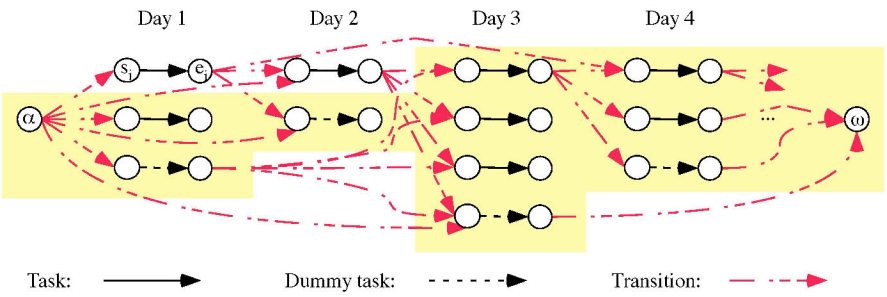

Fig. 1. Sketch of the underlying multi–commodity flow network