A New MIP Model for Parallel-Batch Scheduling with Non-identical Job Sizes

Sebastian Kosch and J. Christopher Beck

Department of Mechanical & Industrial Engineering
University of Toronto, Toronto, Ontario M5S 3G8, Canada
{skosch,jcb}@mie.utoronto.ca

Abstract. Parallel-batch machine problems arise in numerous manufacturing settings from semiconductor manufacturing to printing. They have recently been addressed in constraint programming (CP) via the combination of the novel `sequenceEDD` global constraint with the existing `pack` constraint to form the current state-of-the-art approach. In this paper, we present a detailed analysis of the problem and derivation of a number of properties that are exploited in a novel mixed integer programming (MIP) model for the problem. Our empirical results demonstrate that the new model is able to outperform the CP model across a range of standard benchmark problems. Further investigation shows that the new MIP formulation improves on the existing formulation primarily by producing a much smaller model and enabling high quality primal solutions to be found very quickly.

1 Introduction

Despite the widespread application of mixed integer programming (MIP) technology to optimization problems in general and scheduling problems specifically,[1] there is a significant body of work that demonstrates the superiority of constraint programming (CP) and hybrid approaches for a number of classes of scheduling problems [1–5]. While the superiority is often a result of strong inference techniques embedded in global constraints [6–8], it is sometimes due to problem-specific implementation in the form of specialized global constraints [4] or instantiations of decomposition techniques [1–3]. The flexibility of CP and decomposition approaches which facilitates such implementations is undoubtedly positive from the perspective of solving specific problems better. However, the ability to create problem-specific approaches is in some ways in opposition to the compositionality and model-and-solve "holy grail" of CP [9]: to enable users to model and solve problems without *implementing* anything new at all.

Our overarching thesis is that, in fact, MIP technology is closer to this goal than CP, at least in the context of combinatorial optimization problems. In our investigation of this thesis, we are developing MIP models for scheduling

[1] For example, of the 58 papers published in the *Journal of Scheduling* in 2012, 19 use MIP, more than any other single approach.

H. Simonis (Ed.): CPAIOR 2014, LNCS 8451, pp. 55–70, 2014.

problems where the current state of the art is customized CP or hybrid approaches. Heinz et al. [10] showed that on a class of resource allocation and scheduling problems, a MIP model could be designed that was competitive with the state-of-the-art logic-based Benders decomposition. This paper represents a similar contribution in different scheduling problem: a parallel-batch processing problem which has previously been attacked by MIP, branch-and-price [11], and CP [4] with the latter representing the state of the art.

We propose a MIP model inspired by the idea of modifying a canonical feasible solution. The definition of our objective function in this novel context is not intuitive until we reason algorithmically about how constraints and assignments interact – a strategy often used in designing metaheuristics. Indeed, we suggest that the analogy between branching on independent decision variables and making moves between neighbouring schedules should be explored in more detail for a range of combinatorial problems.

In the next section we present the formal problem definition and discuss existing approaches. In Section 3 we prove a number of propositions that allow us to formally propose a novel MIP model for the problem in Section 4. Section 5 presents our empirical results, demonstrating that the performance of the new model is superior to the existing CP model, both in terms of mean time to find optimal solutions and in terms of solution quality when optimal solutions could not be found within the time limit.

2 Background

Batch machines with limited capacity exist in many manufacturing settings in forms such as ovens [12], autoclaves [4], and tanks [13]. In this paper, we tackle the problem of minimizing the maximum lateness, $L_{\max}$, in a single machine parallel batching problem where each job has an individual due date and size.

We use the following notation: a set $\mathcal{J}$ of n jobs is to be assigned to a set of n batches $\mathcal{B} = \{B_1, \ldots, B_n\}$. Batches can hold multiple jobs or remain empty. Each job j has a processing time, p_j, a size, s_j, and a due date, d_j. Jobs can be assigned to arbitrary batches, as long as the sum of the sizes of the jobs in a batch does not exceed the machine capacity, b.

The single machine processes one batch at a time. Each batch B_k has a *batch start date* S_k, a *batch processing time*, defined as the longest processing time of all jobs assigned to the batch, $P_k = \max_{j \in \mathcal{B}_k}(p_j)$, and a *batch completion date*, which must fall before the start time of the next batch, $C_k = S_k + P_k \leq S_{k+1}$.

The lateness of a job j, L_j, is the completion time of its batch C_k less its due date d_j. The objective function is to minimize the maximum lateness over all jobs, $L_{\max} = \max_{j \in \mathcal{J}}(L_j)$. Since we are interested in the maximum lateness, only the earliest-due job in each batch matters and we define it as the *batch due date* $D_k = \min_{j \in \mathcal{B}_k}(d_j)$.

The problem can be summarized as $1|$*p-batch*$; b < n;$ *non-identical*$|L_{\max}$ [4,11], where *p-batch*$; b < n$ represents the resource's parallel-batch nature and its finite capacity. A version with identical job sizes was shown to be strongly NP-hard by Brucker et al. [14]; this problem, therefore, is no less difficult.