AN IMPROVED LAGRANGIAN RELAXATION METHOD FOR MAXIMISING THE NET PRESENT VALUE OF LARGE RESOURCE-CONSTRAINED PROJECTS *

HANYU GU†, MARK G. WALLACE‡, AND PETER J. STUCKEY§

Key words. Resource constrained project scheduling, Lagrangian relaxation, maximal flow, min-cut clustering

AMS subject classifications. 90C06, 90C35, 90C90

EXTENDED ABSTRACT. Resource-Constrained Project Scheduling Problem (RCPSP) has been extensively studied in the past few decades due to its wide application in diverse industries and the computational challenges it poses as an NP-hard problem. Despite of the rapid theoretical and technical advances in this field [8], heuristics are still the only viable approach for large scale industrial applications. In this paper we aim to provide both tight upper bound and lower bound for the problem of maximising the Net Present Value (NPV) of large projects with resource constraints by using Lagrangian relaxation. The relaxed version of the RCPSP can be represented as a network, and solved as a maximal flow problem. A standard representation has a node for each task at each time-point when it could start, and an edge between each pair of tasks that has precedence relationship. For a problem with thousands of tasks and thousands of possible start times per task, the resulting network has millions of nodes. For a problem instance we have solved which has 1400 activities and a project deadline of 4000, the network has about 5 million nodes and it takes on average 4 minutes to solve the maximal flow problem. For some larger cases we could not even set up the network model on a desktop computer with 16GB memory. To overcome this issue we relax some precedence constraints so that activities can form clusters that are independent from each other. Our goal here is to relax as fewer as possible the precedence constraints but still obtain activity clusters small enough to be solved efficiently. Another problem is that the Lagrangian dual problem converges more slowly due to the high dimensions. We observed that the Lagrangian multipliers related to the precedence constraints converges even more slowly. This makes the Lagrangian relaxation based list scheduling perform poorly. Some preliminary results are reported on the stope scheduling problems ranging from 1400 to 11000 activities.

Main results. We use the Lagrangian Relaxation (LR) method to calculate the upper bound as in [7]. Let T be the deadline of the project, R_k be the capacity of resource $k \in R$, r_{jk} be the resource requirement of activity $j \in J$ on resource k, p_j be the processing time of activity j, w_{jt} be the net present value of activity j when starting at time t, and precedence relation $(i, j) \in L$ if activity j cannot start before activity i completes. The time-indexed formulation for the RCPSP problem is as

*NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

†National ICT Australia, Victoria Research Laboratory, The University of Melbourne, Victoria 3010, Australia (Hanyu.Gu@nicta.com.au).

‡Monash University, Victoria 3100, Australia(mark.wallace@monash.edu).

§National ICT Australia, Department of Computing and Information Systems, The University of Melbourne, Victoria 3010, Australia (pstuckey@unimelb.edu.au).

follows:

$$\text{maximise} \quad npv(x) = \sum_j \sum_t w_{jt} x_{jt} \tag{0.1}$$

$$\text{subject to} \quad \sum_t x_{jt} = 1 \qquad j \in J \tag{0.2}$$

$$\sum_{s=t}^{T} x_{is} + \sum_{s=0}^{t+p_j-1} x_{js} \leq 1 \qquad \forall (i,j) \in L, t = 0, \cdots, T \tag{0.3}$$

$$\sum_j r_{jk} \Big(\sum_{s=t-p_j+1}^{t} x_{js} \Big) \leq R_k \quad k \in R, t = 0, \cdots, T \tag{0.4}$$

$$\text{all variables binary} \tag{0.5}$$

The Lagrangian Relaxation Problem (LRP) obtained by dualizing the resource constraints (0.4) provides a valid upper bound for the RCPSP problem. This upper bound is further optimised by solving the related Lagrangian dual problem.

The LRP for RCPSP can be transformed into a maximal flow problem [7] . The network flow model has $O(|J|T)$ nodes and $O((|J| + |L|)T)$ edges, a state of the art max-flow solver [1] can solve it in $O(|J||L|T^2 \log(T))$. We use the push-relabel implementation in c++ BOOST BGL [9]. In our case the number of nodes V may be in the millions, or for larger instances, hundreds of millions. To overcome this problem we can relax some precedence constraints so that activities can form clusters that are independent from each other. Consequently we can solve a sequence of maximal flow problems more efficiently with respect to both CPU time and memory for large RCPSP problems. However, many more Lagrangian multipliers have to be introduced for (0.3) which will make the Lagrangian dual problem even harder. We could also use a weaker formulation of (0.3)

$$\sum_t t(x_{jt} - x_{it}) \geq p_i, \quad \forall (i,j) \in L \tag{0.6}$$

but the achieved upper bound would be weaker.

Our goal here is to relax as fewer as possible the precedence constraints but still obtain activity clusters small enough to solve efficiently as a maximal flow problem. This can be formulated as the Min-Cut Clustering problem (MCC) as in [3]

$$\text{minimise} \quad \sum_{g=1}^{U} \sum_{e \in L} z_{eg} \tag{0.7}$$

$$\text{subject to} \quad \sum\nolimits_{g=1}^{U} x_{ig} = 1 \qquad i \in J \tag{0.8}$$

$$x_{ig} - x_{jg} \leq z_{eg} \qquad \forall e = (i,j) \in L, g = 1, \cdots, U \tag{0.9}$$

$$l \leq \sum\nolimits_{i \in J} x_{ig} \leq u \qquad g = 1, \cdots, U \tag{0.10}$$

$$\text{all variables binary} \tag{0.11}$$

where U is the upper bound of the number of clusters, x_{ig} is 1 if activity i is included in the cluster g, and otherwise 0.

MCC is also NP-hard, and only small problems can be solved to optimality. For our purpose we can use heuristics to generate good partitions. Our experimentation with METIS [4] shows that the project with 11,000 activities can be partitioned into 100 balanced parts by relaxing just 384 precedence constraints.

The normal Subgradient Algorithm (SA) may converge very slowly on large problems due to the zig-zag phenomenon and small steps. The relax-and-cut idea in [6] identifies the set of active constraints to be dualized at each iteration, which leads to a problem of lower dimension. The simplified bundle method in [10] tried to overcome the zig-zag problem by using $\epsilon-$subgradient as in the bundle algorithm. We found that for large RCPSP the relax-and-cut method improves more rapidly in the first few iterations but may slow down thereafter. The reason may be that the active set of constraints changes too frequently in consecutive iterations. We tried different fuzzy membership functions in the simplified bundle method. It can converge faster than the other two methods for some test cases.

The Lagrangian relaxation DLRP produces upper bounds for the original NPV problem. But in practice we are interested in finding feasible solutions of high value. We can use the Lagrangian relaxation solution to create a heuristic which created strong solutions. Combining Lagrangian relaxation with list scheduling has been previously successfully applied to different variants of RCPSP [7] [5] problems.

The basic idea is motivated by the intuition that violation of relaxed constraints tend to be reduced in the course of the subgradient optimization. In [7] the activities are sorted in the increasing order of the so called $\alpha-$point. A parallel list scheduling scheme [2] is then employed to produce feasible solutions. For RCPSPDC left and right shifting techniques are used to further improve the solution quality [5] of the parallel list scheduling using just the start time in the Lagrangian relaxation solution.

We use the parallel list scheduling scheme to generate feasible solution at each iteration of the subgradient algorithm. However the quality of our heuristic deteriorates dramatically after relaxing precedence constraints. In the end we found that the culprit is the pre-mature convergence of the precedence multipliers. Our improvements on the Lagrangian relaxation method produced very competitive results on large underground mining scheduling problems in our preliminary test results.

REFERENCES

[1] B.V. Cherkassky and A.V. Goldberg, *On implementing push-relabel method for the maximum flow problem*, IPCO '95, pp. 157–171.

[2] R. L. Graham, *Bounds for certain multiprocessing anomalies*, Bell System Tech. J., (1966), pp. 1563–1581.

[3] E.L. Johnson, A. Mehrotra, and G.L. Nemhauser, *Min-cut clustering*, Mathematical Programming, 62 (1993), pp. 133–151.

[4] George Karypis, *METIS, A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices, Version 5.0*, 2011.

[5] A. Kimms, *Maximizing the net present value of a project under resource constraints using a lagrangian relaxation based heuristic with tight upper bounds*, Annals of Operations Research, 102 (2001), pp. 221–236.

[6] Abilio Lucena, *Non delayed relax-and-cut algorithms*, Annals OR, 140 (2005), pp. 375–410.

[7] Rolf H. Möhring, Andreas S. Schulz, Frederik Stork, and Marc Uetz, *Solving project scheduling problems by minimum cut computations*, Management Science, 49 (2003), pp. 330–350.

[8] Andreas Schutt, Thibaut Feydy, Peter J. Stuckey, and Mark G. Wallace, *Explaining the cumulative propagator*, Constraints, 16 (2011), pp. 250–282.

[9] Jeremy G. Siek, Lie-Quan Lee, and Andrew Lumsdaine, *The Boost Graph Library: User Guide and Reference Manual*, Addison-Wesley, 2001.

[10] X. Zhao and P. B. Luh, *New bundle methods for solving lagrangian relaxation dual problems*, J. Optim. Theory Appl., 113 (2002), pp. 373–397.