

Exploiting Cyberinfrastructure to Solve Real-Time Integer Programs

Mustafa R. Kilinc
Advisor: Jeff T. Linderoth

April 23, 2007

Mixed Integer Programming

- A mixed integer program can be stated as

$$\max\{c^T x + d^T y \mid Ax + Gy = b, x \in \mathbb{Z}_+^n, y \in \mathbb{R}_+^p\}$$

- MIP is \mathcal{NP} -Hard, it is unlikely that a good algorithm exists.
- There are powerful and reliable commercial software systems, CPLEX and XPRESS-MP.
- Significant computing power is needed to solve MIP
- Solving a MIP in real-time may well be impossible
- Many MIPs arising in real-time is consecutive instances, differ only slightly.

Parametric Mixed Integer Programming

- The framework we conceptualize in real-time is a parameterized family of MIP

$$\max\{c(\omega)^T x + d(\omega)^T y \mid A(\omega)x + G(\omega)y = b(\omega), x \in \mathbb{Z}_+^n, y \in \mathbb{R}_+^p\}$$

- ω might be a random vector defined on a probability space $(\Omega, \mathcal{F}, \mathcal{P})$ or an element of some uncertainty set \mathcal{U}
- When we observe the event $\hat{\omega}$, in a fixed amount of time, find a solution for

$$\max\{c(\hat{\omega})^T x + d(\hat{\omega})^T y \mid A(\hat{\omega})x + G(\hat{\omega})y = b(\hat{\omega}), x \in \mathbb{Z}_+^n, y \in \mathbb{R}_+^p\}$$

- We claim an effective real-time MIP technology
 - Proactive techniques, anticipate and prepare for the event $\hat{\omega}$.
 - Reactive techniques, employ “real-time” resources once $\hat{\omega}$ is observed.

Proactive Resources

- During proactive(pre-compute) phase, solve problem for different realizations, gather data such as feasible or nearly feasible solutions, partial solutions, dual information, bounds.
- Parallelizing the solution of many simultaneous MIP instances.
- We need massive computational power offered over a long period of time
- A computational grid is the most appropriate resource.
- Used only Condor for solving simultaneous MIP instances.

Reactive/Urgent computing Resources

- On demand resources for the execution of real-time requests.
- Implementation of high-performance MIP heuristics.
 - Neighborhood search of previously stored feasible solutions
 - Parallelized strong-branching used for reducing infeasibility
- High performance resources are needed upon urgent computer requests
- MPI to parallelize heuristics

I plan to learn from this workshop

- How to use grid resources to generate many solutions over a long time span
- How to use high performance computing resources to offer good solutions quickly to a given instance.
- Scheduling mechanisms that could accomodate urgent computing requests
- Parallelize primal heuristics via MPI