Project Prioritization via Optimization

Ali Koç David Morton Elmira Popova

The University of Texas at Austin

Cyberinfrastructure experiences for graduate students, San Diego, 2007

Outline

Me

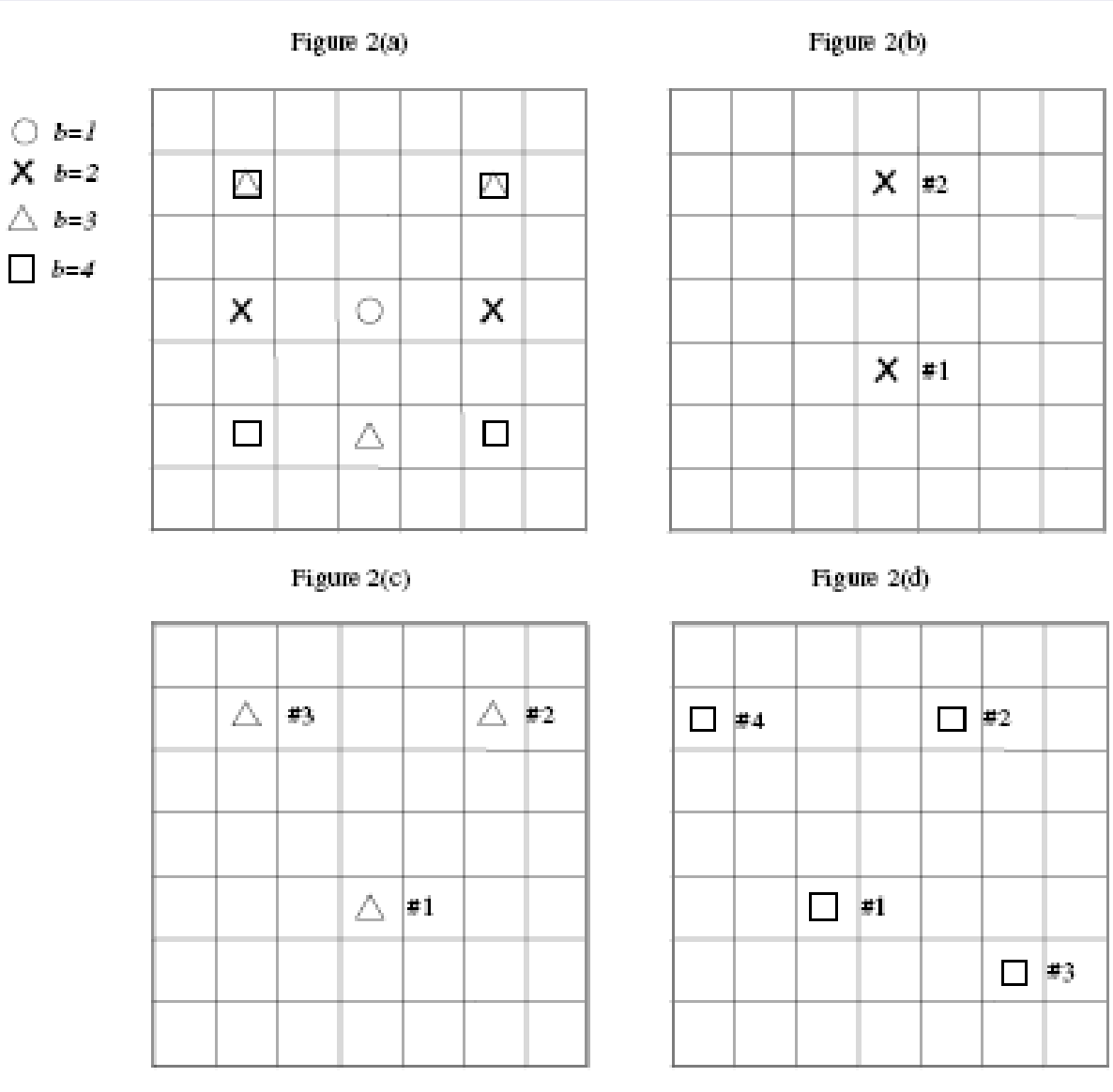
Past

- B.S. 2002. Bilkent University, Turkey.
- M.S. 2005. Bilkent University, Turkey.
- Master's thesis: Assembly/Disassembly Line Balancing using AND/OR Graph.

Now

- Started PhD in 2005 in the ORIE department of University of Texas at Austin.
- Working with Dr. David Morton and Dr. Elmira Popova on "Prioritization via Stochastic Optimization".
- Took a two-day intensive course entitled *Introduction to Parallel Computing* offered by TACC.

Facility Location Prioritization



Project Prioritization

Data from current STPNOC Practice

c_{it}	2006	2007	2008	2009	2010+	NPV
1	\$6.740	\$6.134	\$10.442			\$60.589
2	\$0.425					\$0.667
3	\$0.030	\$0.030	\$0.688			\$0.122
4			\$0.122	\$0.103	\$0.013	\$0.824
⋮		⋮				⋮

Project Prioritization

Solution for various budget levels

Budget Levels (M)	1	2	3	4	5	6	7	8	9	NPV (M)
11	0	1	0	1	1	1	0	0	0	$ -23.58 M
12	0	1	1	1	1	1	0	0	0	$ -23.46 M
13	0	0	1	1	1	1	1	0	0	$ -18.95 M
14	0	1	1	1	1	1	1	0	0	$ -18.29 M
15	0	0	1	1	1	0	1	1	0	$ -17.27 M
16	0	0	1	1	1	1	0	0	1	$ -1.67 M
⋮										⋮

As the budget levels vary, some of the projects are alternatively in and out of the portfolio

Forming a robust portfolio

Heuristically

Solve the problem for the lowest budget level. Then, solve it for the next highest budget level keeping all the previously selected projects in the portfolio. Go until the highest budget level

Optimally

Determine the probability of each budget occurring. Put the projects into an optimal priority list such that no matter what the budget level is, a lower ranked project can not be selected unless all of the higher ranked ones are selected

Comparing the Heuristic and Optimal Priority Lists

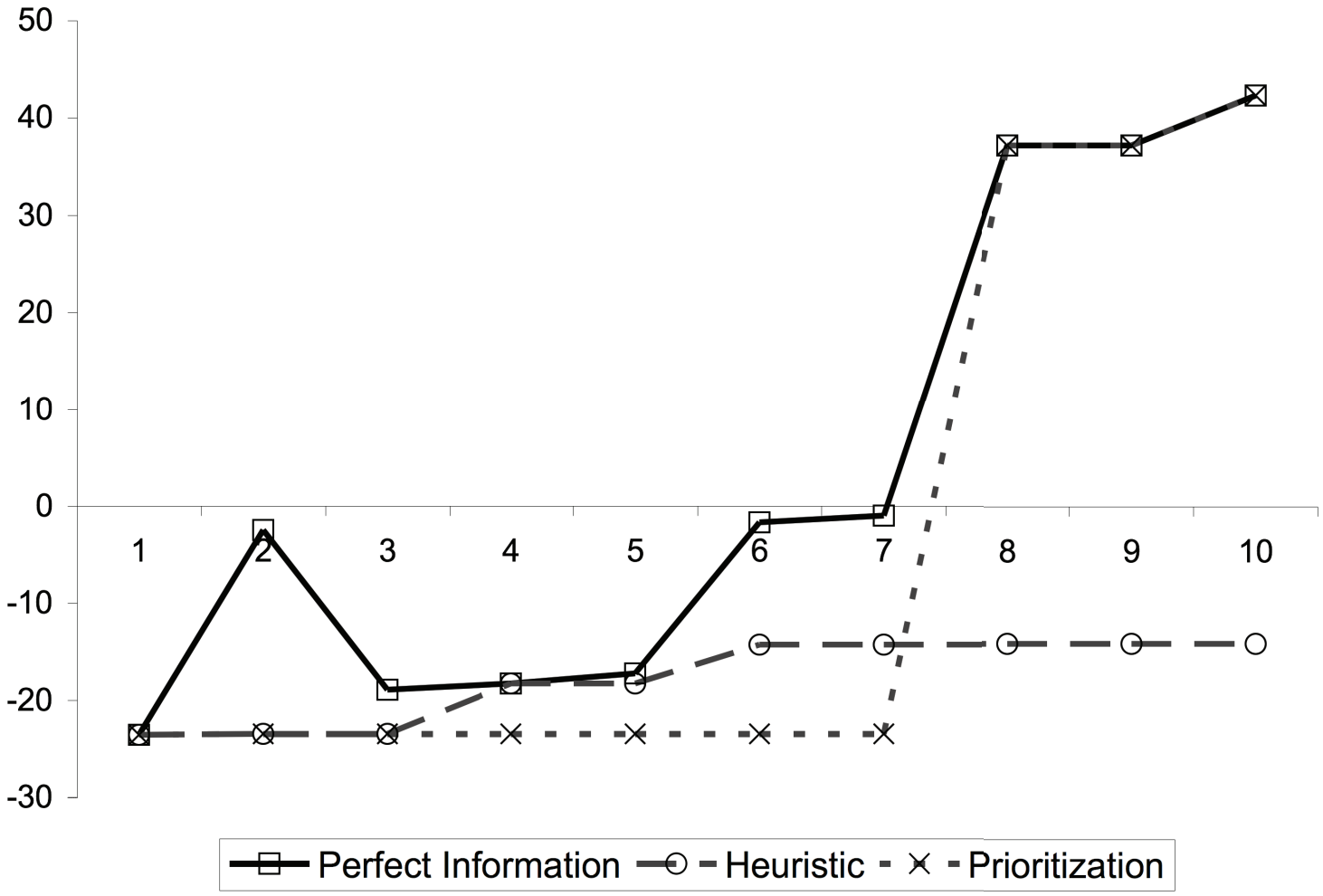
Solutions

Heuristic		Optimal	
Priority	Project	Priority	Project
1-4	{2,4,5,6}	1-4	{2,4,5,6}
5	3	5	3
6	7	6	1
7	8	7	7
8-9	{1,9}	8-9	{8,9}

Expected NPVs

Perfect Information	Optimal	Heuristic
$11.90 M	$ 2.60 M	$ -15.42 M

Comparing the Heuristic and Optimal Priority Lists



How about the problem difficulty
Deterministic Problem

$$\max_{x} \quad \sum_{i \in I} a_i x_i \tag{1a}$$

$$\text{s.t.} \quad \sum_{i \in I} c_{it} x_i \leq b_t, \qquad t \in T \tag{1b}$$

$$x_i \in \{0, 1\}, \qquad i \in I. \tag{1c}$$

How about the problem difficulty

Stochastic Problem

$$\max_{x,y,z} \quad \sum_{\omega\in\Omega} q^{\omega} \sum_{i\in I} a_i x_i^{\omega} \tag{2a}$$

$$\text{s.t.} \quad \sum_{i\in I} c_{it} x_i^{\omega} \le k^{\omega} b_t, \qquad t \in T, \ \ \omega \in \Omega \tag{2b}$$

$$\sum_{i\in I} z_{ip} = 1, \qquad p \in P \tag{2c}$$

$$\sum_{p\in P} z_{ip} \le 1, \qquad i \in I \tag{2d}$$

$$|P| y_{ii'} \ge \sum_{p\in P} (|P| - p)(z_{ip} - z_{i'p}), \quad i \ne i', i, i' \in I \tag{2e}$$

$$y_{ii'} + y_{i'i} = 1, \qquad i < i', \ \ i, i' \in I \tag{2f}$$

$$x_i^{\omega} \ge x_{i'}^{\omega} + y_{ii'} - 1, \qquad \omega \in \Omega, i \ne i', i, i' \in I \tag{2g}$$

$$x_i^{\omega},\ y_{ii'},\ z_{ip} \in \{0,1\}, \quad i \ne i', i, i' \in I, \omega \in \Omega, p \in P \tag{2h}$$

Decomposition Approaches

Sequential

- Dual decomposition: Uses lagrangian duality and branch-and-price, the latter being better usually
- Primal decomposition: Uses cutting planes

Parallel

- Primal decomposition: Does not seem parallelizable since the sequential part of the algorithm increases dramatically during the course of the algorithm
- Dual decomposition: Can be parallelized since the sequential part of the algorithm increases slightly.

Parallel Decomposition Approaches
State of the art

- Some work on parallel lagrangian duality
- One implementation of parallel branch-and-price (Lulli and Sen 2004)
- Grid computing implementations (Linderoth and Wright 2003)

A parallel branch-and-price implementation that utilizes the theoretical and practical aspects of both branch-and-price methodology and parallel computation (on clusters) will be a good research practice

My expectations

The Algorithm

Consists of an unknown number of loops, each of which has high parallel-to-sequential code ratio, and in each iteration of the loop that ratio is almost constant

Goals

- Implement parallel branch-and-price on SMPs and distributed memory processors (using OpenMP and MPI), and compare the two
- Achieve scalability and high speed up
- Test various dual decompositions for branch-and-price
- Test various architectures of distributed memory processors