ECE619 Final Presentation

Jiahui Guo

University of Tennessee, Knoxville

May 7, 2013

Problem Outline

- Unit Commitment Model

Minimize

$$\sum_{t=1}^{N_h} \sum_{k=1}^{N_g} [a_k(t) + b_k P_k(t) + c_k P_k^2(t) + Fs_k(1 - Z_k(t-1))] Z_k(t)$$

s.t.

$$\sum_{k=1}^{N_g} P_k(t) = D(t)$$

$$P_{min,k}(t) Z_k(t) \leq P_k(t) \leq P_{max,k}(t) Z_k(t)$$

$$P_k(t) + R_k(t) - \sum_{k=1}^{N_g} P_{max,k} Z_k(t) \leq 0$$

$$Z_k(t) \in \{0, 1\}$$

$$\forall k = 1, 2, ..., N_g$$

$$\forall t = 1, 2, ..., N_h$$

Algorithms

1. Priority List
2. Dynamic Programming
3. Dynamic Programming + Priority List
4. Particle Swarm Optimization

Results and Comparison

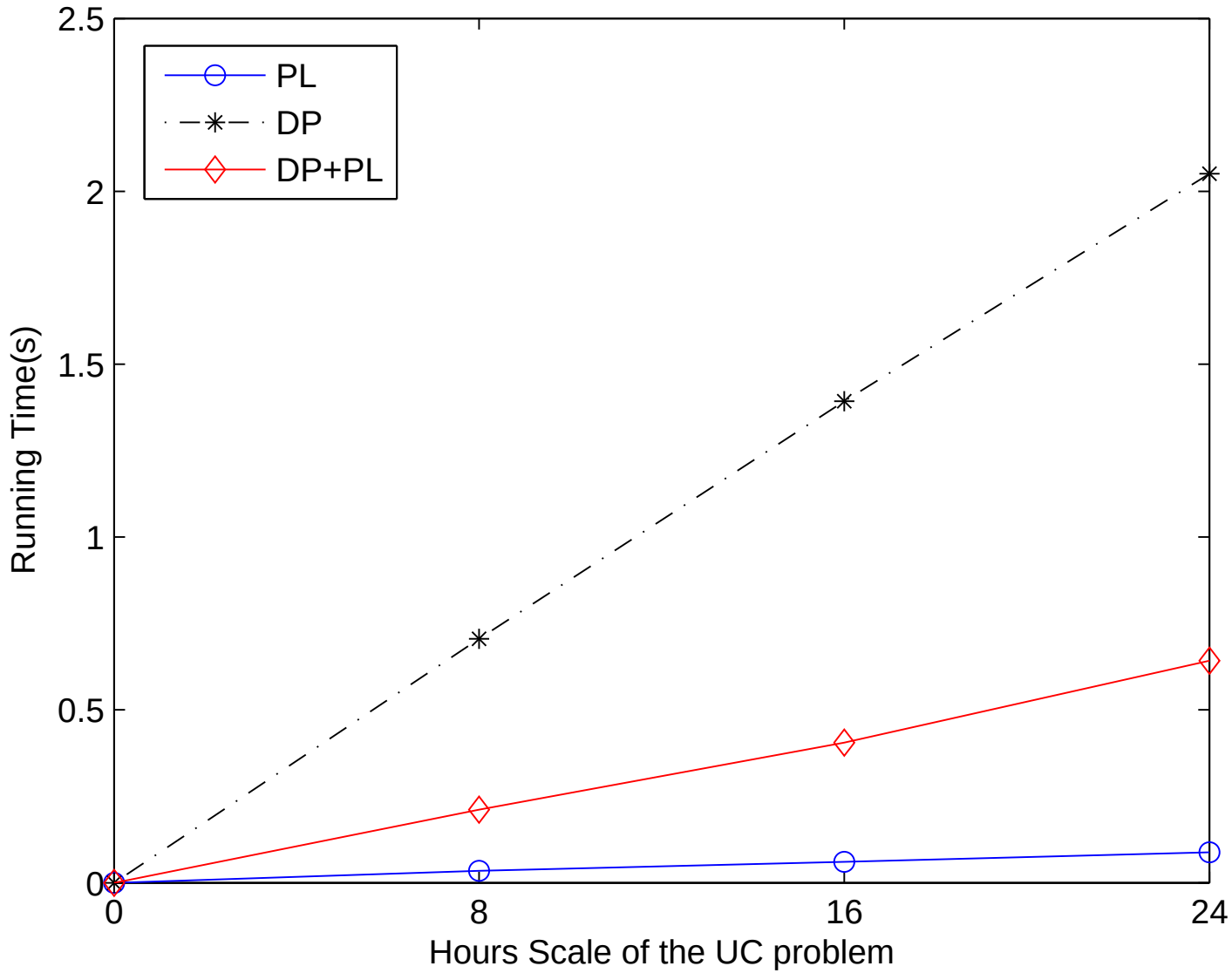
- Test System
  - 4 Generations
  - Scheduling for 8 hours
- Results

Alg.	Cost($)	Time(s)	Comments
PL	73180.39	0.0193	Omit start up cost when scheduling
DP	73180.39	0.6517	Traverse all the possible states
DP+PL	73180.39	0.1860	Traverse the priority list states
PSO	77172.67[1]	343.15	Consider reserve

[1] Best Case in 10 times

Results and Comparison

- Runtime Comparison (4 Generation)



Pros and Cons

- Pros
  - Algorithm
    - DP+PL speed up dynamic program to obtain an optilmal solution
    - PSO is easy to be implemented
  - Programming
    - Well utilizing the matrix manipulation advantage of Matlab
    - Detailed comments in code
    - Proper interface for functions
- Cons
  - Algorithm
    - Optimal solution maybe missed by priority list
    - Detailed UC problem is hard to be modeled by dynamic programming
    - Problem scale is limited for PSO (Much iteration time needed and converge issues)
    - Parameter adjustment in PSO
  - Programming
    - Not all the constraints are considered

Thanks

Any Questions?