# CS 747

# Assignment 2 - Report

*Goutham Ramakrishnan, 140020039*

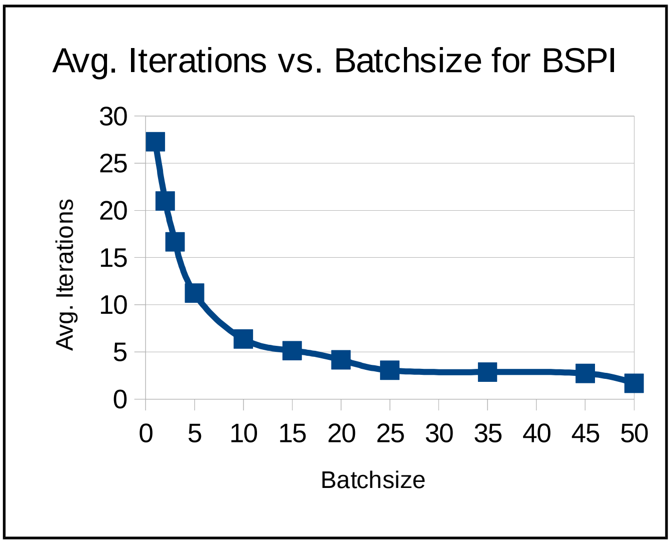
## Process used to obtain MDP instances:

- 100 instances of 50-state, 2-action MDPs were generated for the simulation.
- The reward and transition functions of the MDPs were generated randomly.
- The randomseed was set deterministically for each instance.
- The reward values were sampled from a uniform (-1,1) distribution.
- The transition values were sampled from a uniform (0,1) distribution and then normalized to make them valid probability distributions.
- Gamma was sampled randomly from a uniform (0.9,0.99) distribution.
- Due to the random generation of MDP instances, we can be reasonably sure of obtaining a varied sample space in which to test the algorithms, in order to obtain a reliable average.

## Summary of simulation results:

Simulation of the the three policy iteration algorithms was done on 100 different 50-state 2-action MDPs. The average number of iterations for convergence for each algorithm has been summarized in the below table. The raw data values can be viewed in the attached spreadsheet.
Note: The number of iterations excludes the iteration in which the algorithm discovers that the current policy is optimal. Therefore it represents the number of times the policy had to be modified in order to attain the optimal policy.

| 50-state, 2-action MDP | Batchsize | Average Number of Iterations Till Convergence |
|---|---|---|
| Howard's PI | - | 1.68 |
| Mansour and Singh's PI | - | 6.86 |
| Batch PI | 1 | 27.28 |
| | 2 | 21 |
| | 3 | 16.68 |
| | 5 | 11.22 |
| | 10 | 6.36 |
| | 15 | 5.14 |
| | 20 | 4.16 |
| | 25 | 3.05 |
| | 35 | 2.87 |
| | 45 | 2.72 |



## Observations and Interpretation of Results:

- Howard’s PI outperformed Randomized PI and Batch-Switching PI.
- Remarkably, HPI almost always converged within one or two iterations. This is remarkable as theoretically switching all improvable states may not necessarily be optimal. The performance of HPI is sure to be worse on carefully constructed 2-action MDPs or on MDPs with more actions.
- As expected the average number of iterations to convergence reduces with increase in batchsize for BSPI. With very small batchsizes, the algorithm is rather inefficient and slow. As the batchsize increases, the performance drastically improves initially and later saturates.
- In the graph plotted above, the data point corresponding to batchsize=50 is the value obtained for Howard’s PI, which is nothing but BSPI with batchsize=S.
- From the graph, we can observe that the data point for Howard’s PI(batchsize=50) fits well into the curve.
- Even though Randomized PI and BSPI have better theoretical bounds than HPI, in practice HPI seems to work much better. The graph seems to suggest that the theoretical bound for HPI is the theoretical bound for BSPI for batchsize=S.

## Attached:

Shell script to run program: planner.sh
Primary python script: mdp.py
Spreadsheet with raw simulation data
Python script used to generate MDPs and run simulation: mdp_analyze.py

## References:

- → Python Documentation
- → Pulp Documentation
- → CS-747 Lecture slides on Policy Iteration
- → StackOverFlow
- → https://www.cse.iitb.ac.in/~shivaram/papers/kmg_ijcai_2016.pdf