COMP767 - Reinforcement Learning

Labeled RTDP: Improving the Convergence of Real-Time Dynamic Programming

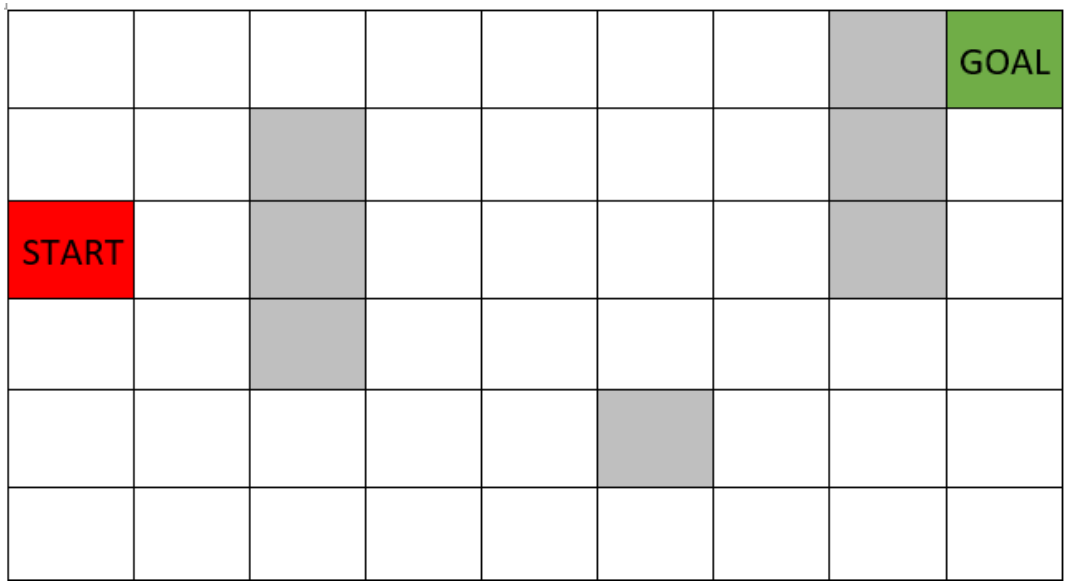
Claudio Sole

École polytechnique de Montréal

March 13, 2017

(Deterministic) Shortest Path Problem on a grid

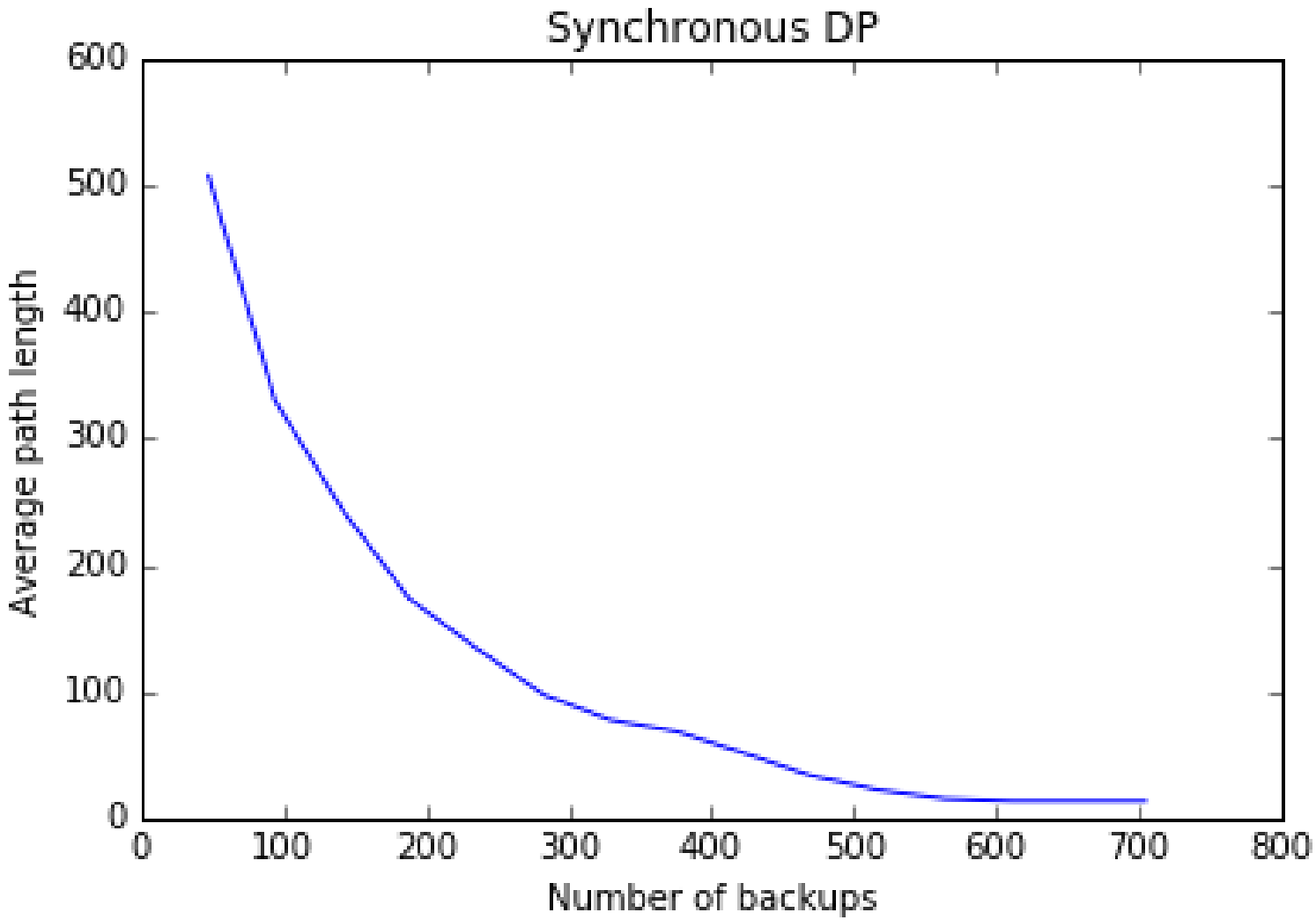
- For each state the possible action are $\{up, down, right, left\}$
- If an action takes the agent out of the grid or against a wall, then the position on the grid remains the same (self-transitions)
- The reward is -1 for all transitions into non-terminal state and 0 for *Goal* state
- $Q(s, a)$ initialized at 0 for every s, a



Algorithm 1 Synchronous DP

Initialize V arbitrarily (e.g. $V(s) = 0 \ \forall s \in S$
$\epsilon \leftarrow 10^{-4}$
$\Delta \leftarrow \epsilon + 1$
while $\Delta > \epsilon$ **do**
    $\Delta \leftarrow 0$
    $V_{old} \leftarrow V$
    for $s \in S$ **do**
        $V(s) \leftarrow max_a \sum_{s',r} p(s', r|s, a)\left[r + \gamma V_{old}(s')\right]$
        $\Delta \leftarrow max\{\Delta, |V_{old}(s) - V(s)|\}$

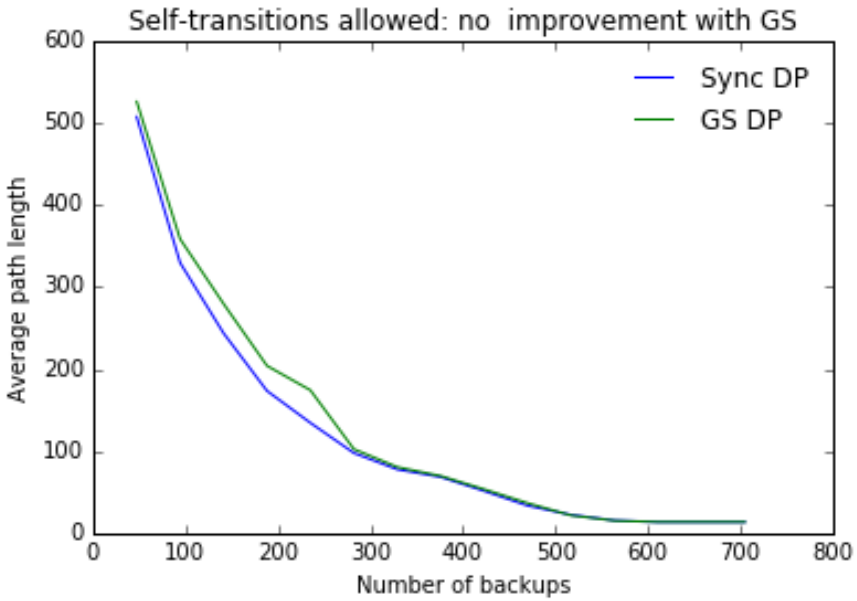
Remark : states ordering has no influence


Synchronous DP
Average path length
Number of backups
0
100
200
300
400
500
600
700
800

Algorithm 2 Gauss-Seidel DP

Initialize V arbitrarily (e.g. $V(s) = 0 \ \forall s \in S$
$\epsilon \leftarrow 10^{-4}$
$\Delta \leftarrow \epsilon + 1$
while $\Delta > \epsilon$ **do**
    $\Delta \leftarrow 0$
    ~~$V_{old} \leftarrow V$~~
    for $s \in S$ **do**
        $v \leftarrow V(s)$
        $V(s) \leftarrow max_a \sum_{s',r} p(s', r|s, a)\left[r + \gamma V_{\cancel{old}}(s')\right]$
        $\Delta \leftarrow max\{\Delta, |v - V(s)|\}$

Remark: states ordering influences convergence rate



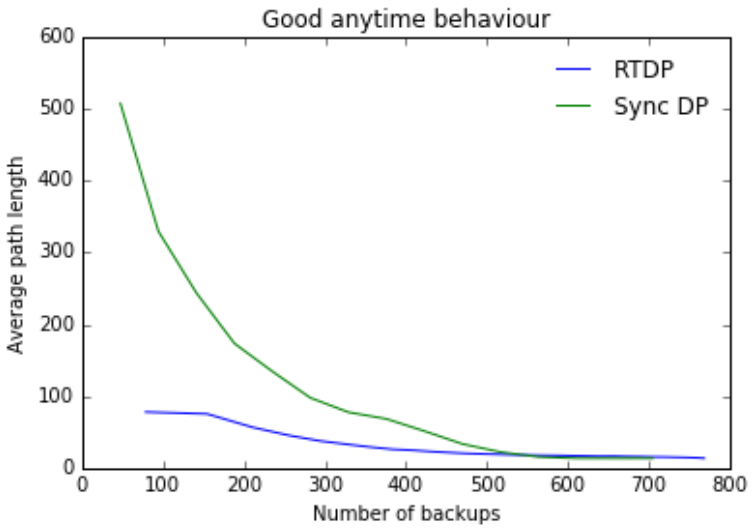
self-transitions	Yes		No	
	Sync	GS	Sync	GS
# sweeps	15	15	15	11
# backups	705	705	705	517

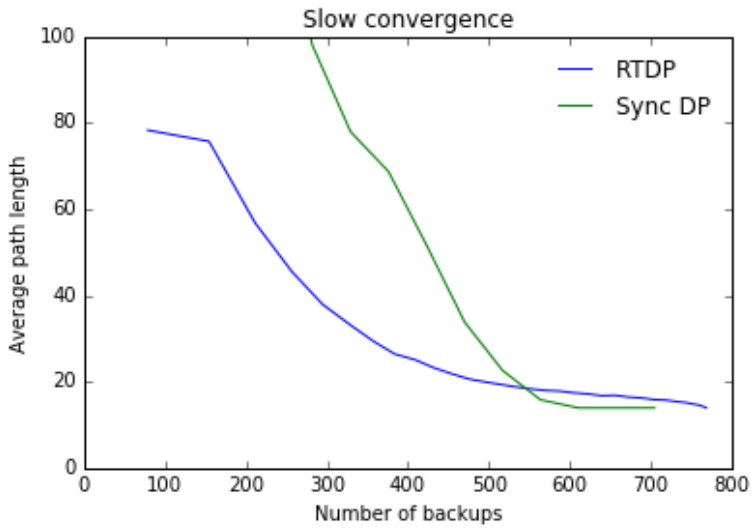
Remark: for the Gauss-Seidel implementation a random ordering for state backups has been chosen.

Algorithm 3 TRIAL-BASED RTDP

function RTDP( ):
    while not converged **do**
        RTDP_TRIAL(*start*)

function RTDP_TRIAL(s):
    while s is not *goal* **do**
        $greedy\_a \leftarrow argmax_a \sum_{s',r} p(s', r|s, a)\left[r + \gamma V(s')\right]$
        $V(s) \leftarrow \sum_{s',r} p(s', r|s, greedy\_a)\left[r + \gamma V(s')\right]$
        $s \leftarrow s'$ with probability $P(s'|s, greedy\_a)$


Good anytime behaviour
RTDP
Sync DP
Average path length
Number of backups
0
100
200
300
400
500
600
700
800


Slow convergence
RTDP
Sync DP
Average path length
Number of backups
0
20
40
60
80
100
0
100
200
300
400
500
600
700
800

Definitions:

- **Greedy graph(s):** graph made up of the states reachable from s following the greedy policy
- **Greedy envelope(s):** set of states in the greedy graph of s
- **Residual(s)** $= \left|V(s) - Q(s, greedy\_a)\right|$

Idea

We keep track of the states over which the value function has converged and avoid visiting those states again

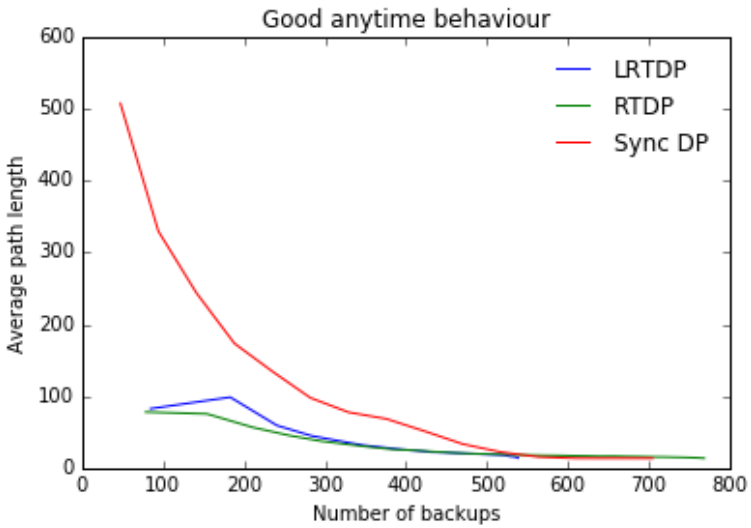
Algorithm 4 CheckSolved(s)

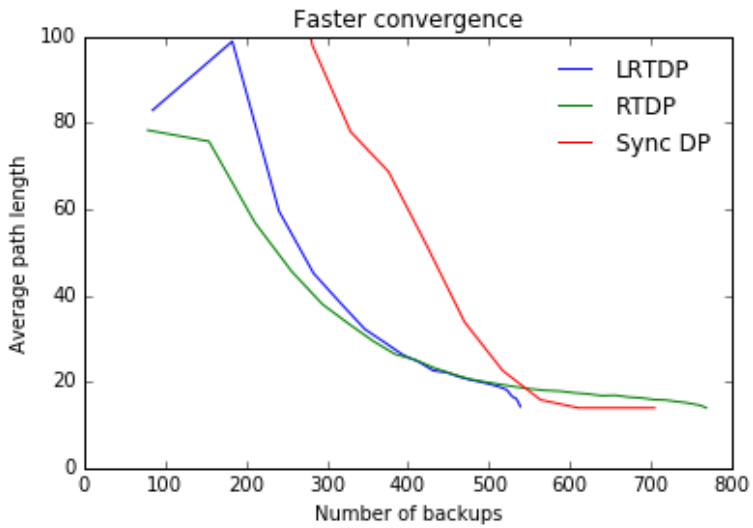
1: **if** $\nexists s' \in \mathit{greedyEnvelope}(s) : \mathit{Residual}(s') > \epsilon$ **then**
2: mark as SOLVED s
3: mark as SOLVED each states in the greedy envelope of s
4: return TRUE
5: **else**
6: backup states in the greedy envelope of s
7: return FALSE

Algorithm 5 TRIAL-BASED LRTDP

function LRTDP( ):
    while *start* not SOLVED **do**
        LRTDP_TRIAL(*start*)

function LRTDP_TRIAL(s):
    visited = empty Stack
    while *s* is not *SOLVED* **do**
        visited.push(s)
        if *s* is *goal* **then**
            break
        $greedy\_a \leftarrow argmax_a \sum_{s',r} p(s', r|s, a)\left[r + \gamma V(s')\right]$
        $V(s) \leftarrow \sum_{s',r} p(s', r|s, greedy\_a)\left[r + \gamma V(s')\right]$
        $s \leftarrow s'$ with probability $P(s'|s, greedy\_a)$
    while *visited* $\neq$ EMPTY **do**
        $s =$ *visited*.pop()
        if not checkSolved(s) **then**
            break


Good anytime behaviour
LRTDP
RTDP
Sync DP
Average path length
Number of backups
0
100
200
300
400
500
600
700
800


Faster convergence
LRTDP
RTDP
Sync DP
Average path length
Number of backups
0
20
40
60
80
100
0
100
200
300
400
500
600
700
800


Barto, Andrew G. and Bradtke, Steven J. and Singh, Satinder P.
Learning to act using real-time dynamic programming
Artificial Intelligence
1995


Bonet, Blai and Geffner
Labeled RTDP : Improving the Convergence of Real-Time Dynamic Programming
Proceedings of Thirteenth International Conference on Automated Planning and Scheduling
2003