Implementing Value Iteration

1. Make it work
2. Make it right
3. Make it fast

Implementing Value Iteration

1. Make it work
2. Make it right
3. Make it fast

Problem 4

Implementing Value Iteration

1. Make it work
2. Make it right
3. Make it fast

Problem 4

Problem 5

Implementing Value Iteration

1. Make it work
2. Make it right
3. Make it fast

Problem 4

Problem 5

First step for making it fast (in any language, not just julia):

Implementing Value Iteration

1. Make it work
2. Make it right
3. Make it fast

Problem 4

Problem 5

First step for making it fast (in any language, not just julia):

Find out what is slow (by profiling)!

Bellman Operator

$$U' = B[U]$$

$$B[U](s) = \text{max}_a \underbrace{\left(R(s, a) + \gamma \sum_{s'} T(s' \mid s, a) U(s') \right)}_{Q(s,a)}$$

Bellman Operator

$$U' = B[U]$$

$$B[U](s) = \max_a \underbrace{\left(R(s,a) + \gamma \sum_{s'} T(s' \mid s,a) U(s')\right)}_{Q(s,a)}$$

i = index of s; j = index of s'

Naive implementation:

$$U'[i] = \max_a \left(R[a][i] + \gamma \sum_j T[a][i,j] U[j]\right)$$

Bellman Operator

$$U' = B[U]$$

$$B[U](s) = \max_a \underbrace{\left(R(s, a) + \gamma \sum_{s'} T(s' \mid s, a) U(s')\right)}_{Q(s,a)}$$

i = index of s; j = index of s'

Naive implementation:

$$U'[i] = \max_a \left(R[a][i] + \gamma \sum_j T[a][i, j] U[j]\right)$$

$$y = Mx$$

$$y[i] = \sum_j M[i, j] x[i]$$