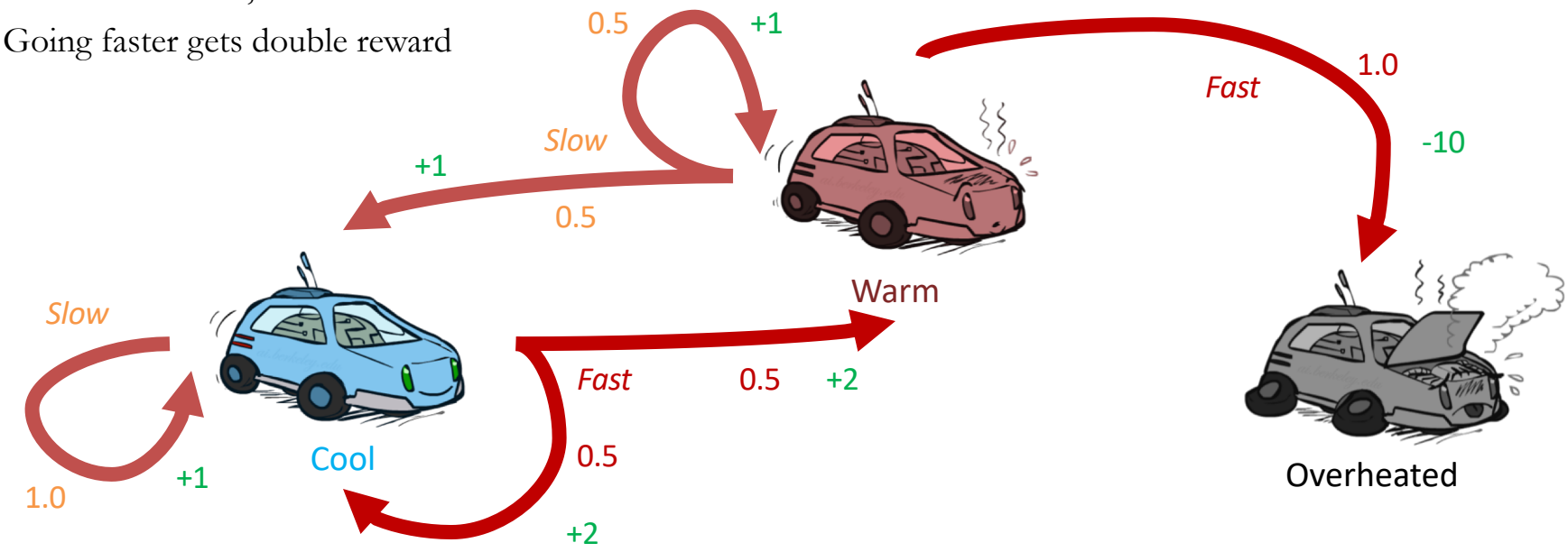
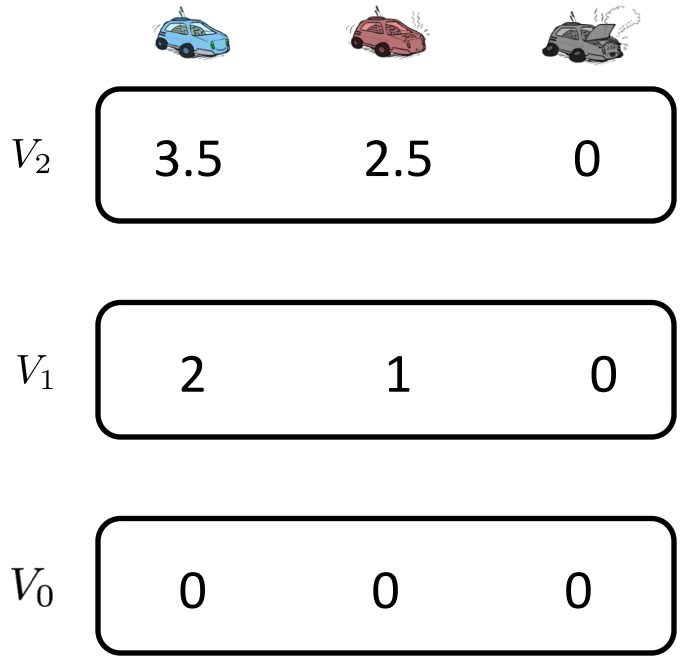
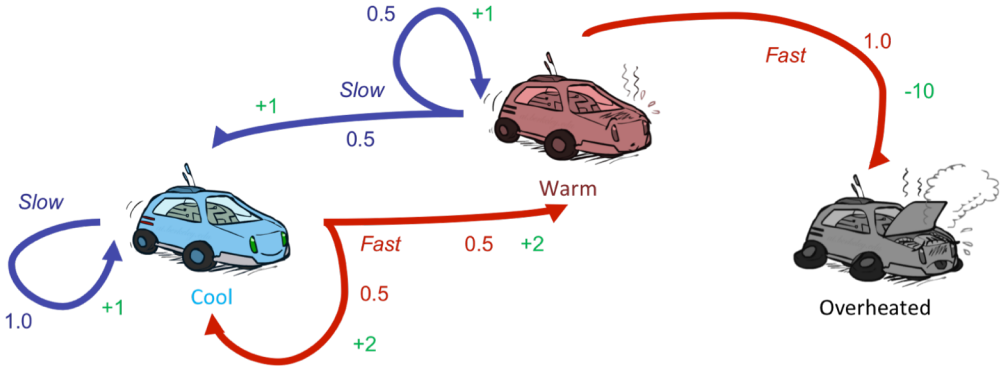
Racing

- A robot car wants to travel far, quickly
- Three states: Cool, Warm, Overheated
- Two actions: *Slow*, *Fast*
- Going faster gets double reward



Value iteration

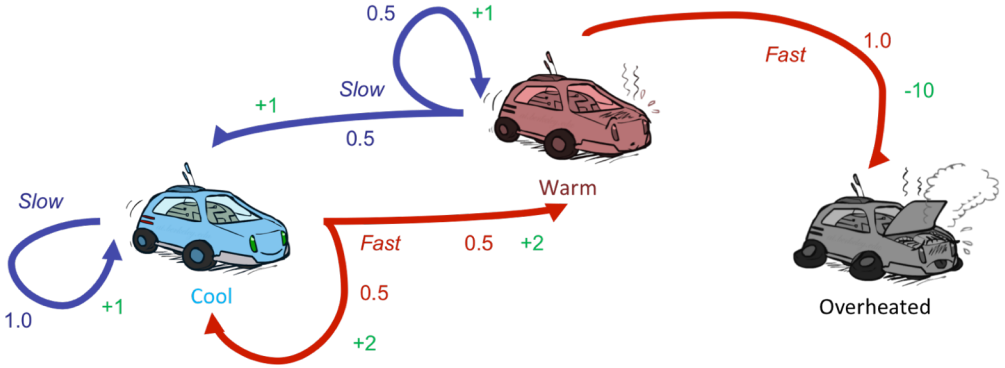




Assume no discount!

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

Value iteration



Assuming a discount factor of 0.5

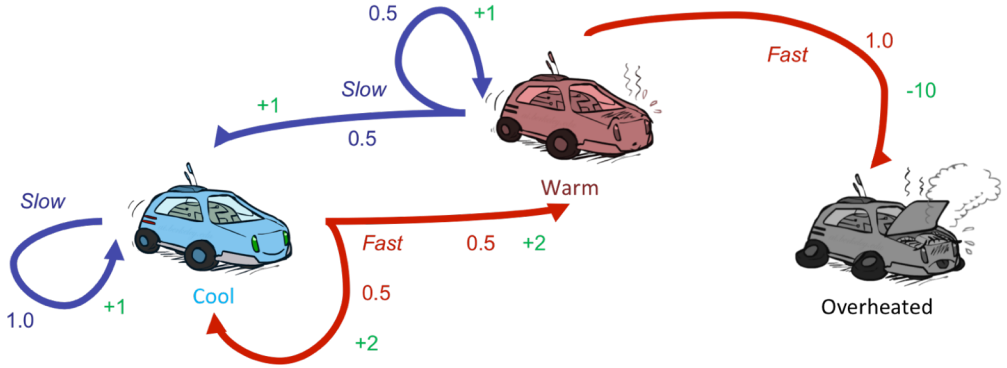
$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

V_2

V_1

V_0 0 0 0

Value iteration



V_2

V_1

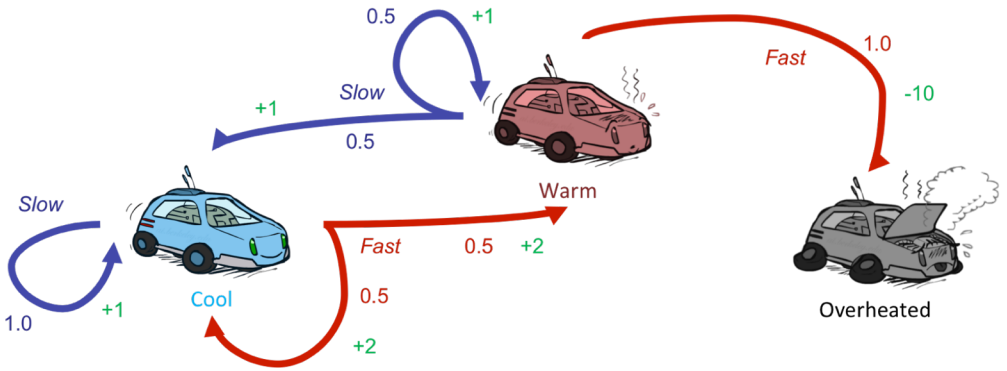
V_0 0 0 0

Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

$V_{k+1}(s) = \max_a \{R(s, a) + \Sigma_{s'}[\gamma * T(s, a, s')V_k(s')]\}$

Value iteration



V_2

V_1 2 1 0

V_0 0 0 0

Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

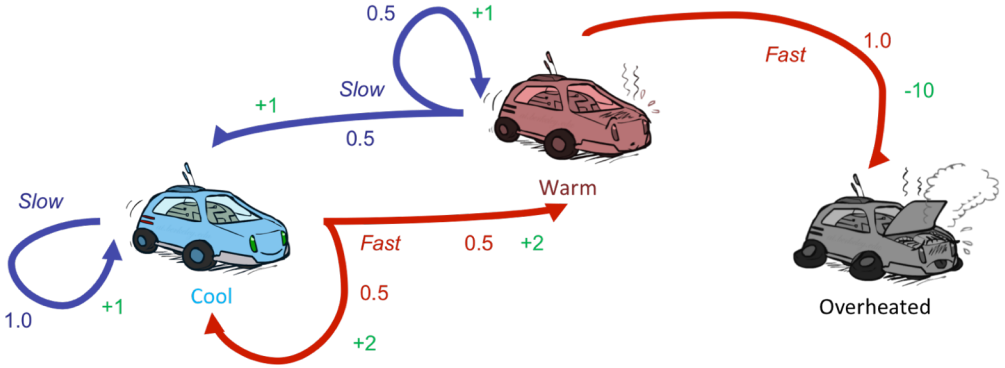
$V_{k+1}(s) = \max_a \{R(s, a) + \Sigma_{s'}[\gamma * T(s, a, s')V_k(s')]\}$

V1(cool) = max{
slow: 1 + 1.0 * 0.5 * 0 = 1
fast: 2 + 0.5 * 0.5 * 0 + 0.5 * 0.5 * 0 = 2
} = 2

V1(warm) = max{
slow: 1 + 0.5 * 0.5 * 0 + 0.5 * 0.5 * 0 = 1
fast: -10 + 1.0 * 0.5 * 0 = -10
} = 1

V1(overheated) = V0(overheated) = 0

Value iteration



Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

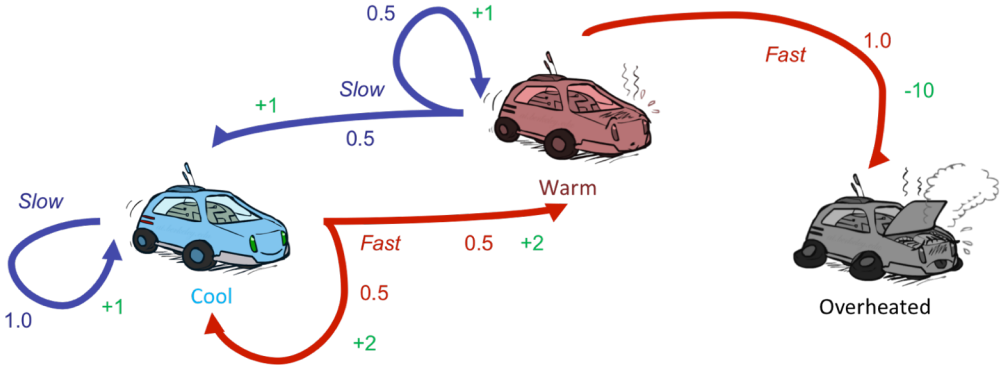
$V_{k+1}(s) = \max_a \{R(s, a) + \Sigma_{s'}[\gamma * T(s, a, s') V_k(s')]\}$

V_2

V_1 2 1 0

V_0 0 0 0

Value iteration



V_2	2.75	1.75	0
V_1	2	1	0
V_0	0	0	0

Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s')\right]$$

$V_{k+1}(s) = \max_a \{R(s, a) + \Sigma_{s'}[\gamma * T(s, a, s')V_k(s')]\}$

V2(cool) = max{
slow: 1 + 1.0 * 0.5 * 2 = 2
fast: 2 + 0.5 * 0.5 * 2 + 0.5 * 0.5 * 1 = 2.75
} = 2.75

V2(warm) = max{
slow: 1 + 0.5 * 0.5 * 2 + 0.5 * 0.5 * 1 = 1.75
fast: -10 + 1.0 * 0.5 * 0 = -10
} = 1.75

V2(overheated) = V1(overheated) = 0

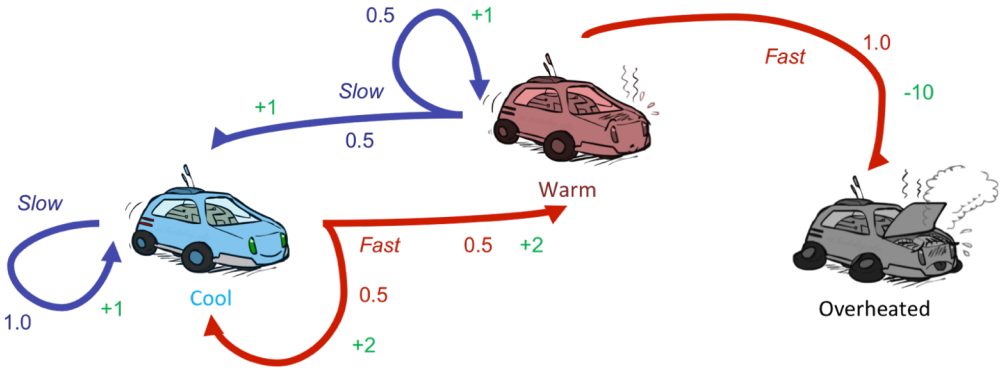
Policy evaluation

π | Fast | Fast | -

V_2

V_1

V_0 | 0 | 0 | 0



Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

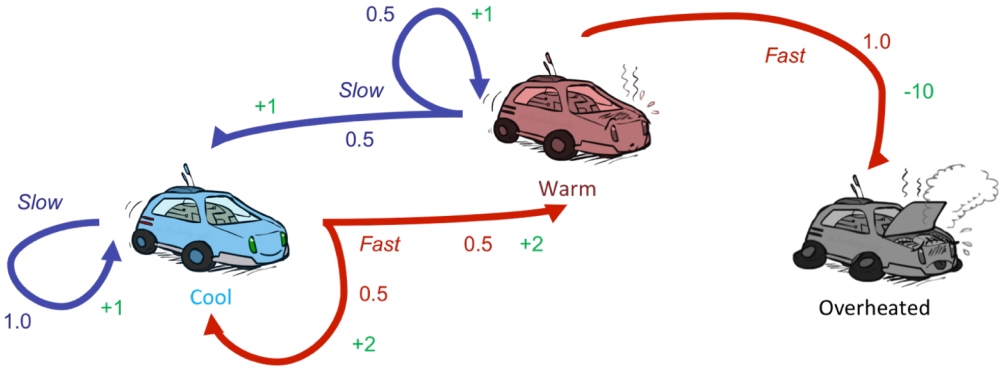
Policy evaluation

π | Fast | Fast | -

V_2

V_1

V_0 | 0 | 0 | 0



Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

$V_{k+1}(s) = R(s, \boldsymbol{\pi}(s)) + \boldsymbol{\Sigma}_{s'} [\gamma * T(s, \boldsymbol{\pi}(s), s') V_k(s')]$

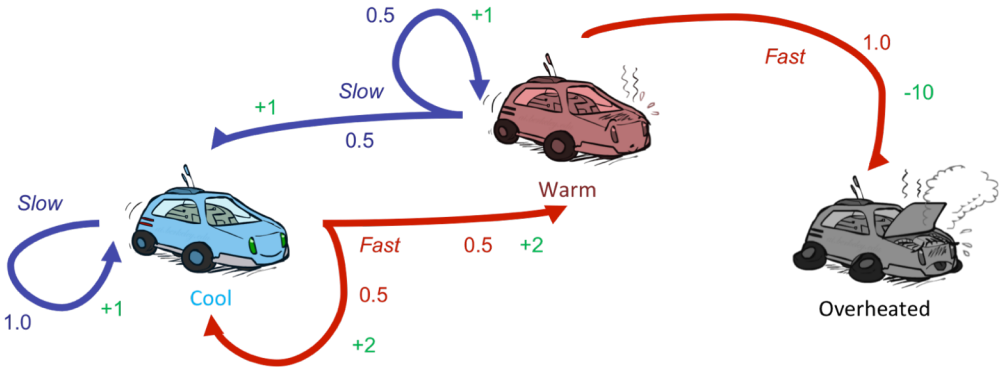
Policy evaluation

π | Fast | Fast | -

V_2 | | |

V_1 | 2 | -10 | 0

V_0 | 0 | 0 | 0



Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

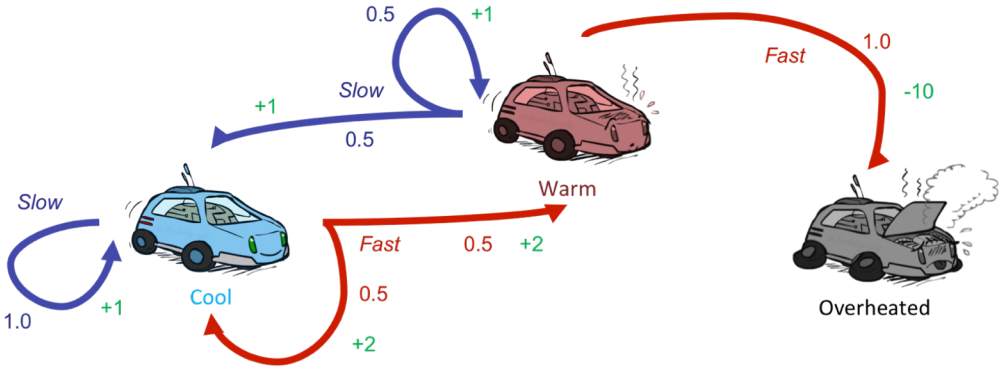
$V_{k+1}(s) = R(s, \pi(s)) + \Sigma_{s'} [\gamma * T(s, \pi(s), s')V_k(s')]$

V1(cool) = fast: 2 + 0.5 * 0.5 * 0 + 0.5 * 0.5 * 0 = 2
V1(warm) = fast: -10 + 1.0 * 0.5 * 0 = -10
V1(overheated) = V0(overheated) = 0

Policy evaluation

π | Fast | Fast | -

V_2			
V_1	2	-10	0
V_0	0	0	0



Assuming a discount factor of 0.5

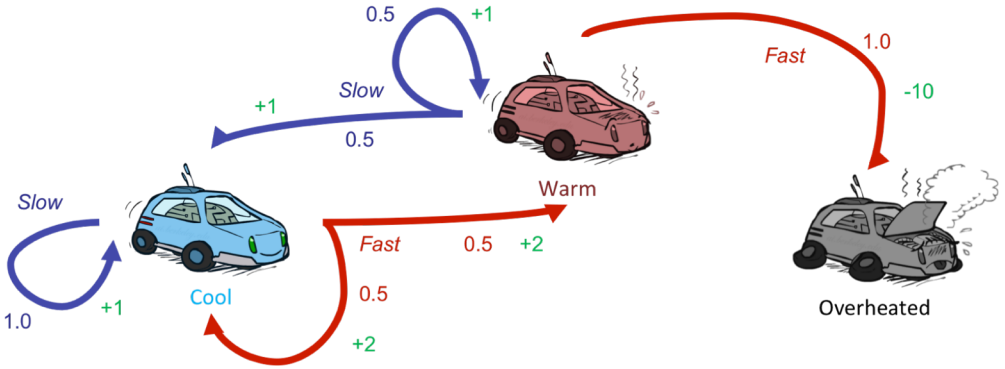
$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

$V_{k+1}(s) = R(s, \boldsymbol{\pi}(s)) + \boldsymbol{\Sigma}_{s'} [\gamma * T(s, \boldsymbol{\pi}(s), s') V_k(s')]$

Policy evaluation

π	Fast	Fast	-

V_2	0	-10	0
V_1	2	-10	0
V_0	0	0	0



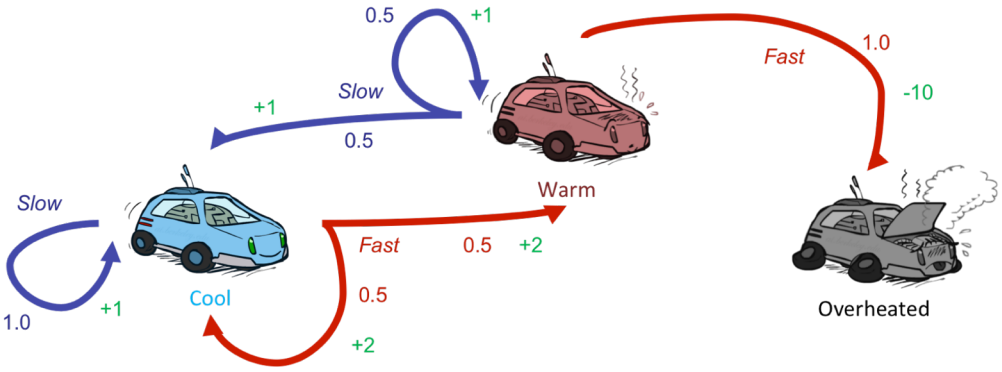
Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

$V_{k+1}(s) = R(s, \boldsymbol{\pi}(s)) + \boldsymbol{\Sigma}_{s'} [\gamma * T(s, \boldsymbol{\pi}(s), s') V_k(s')]$

V2(cool) = fast: 2 + 0.5 * 0.5 * 2 + 0.5 * 0.5 * -10 = 0
V2(warm) = fast: -10 + 1.0 * 0.5 * 0 = -10
V2(overheated) = V1(overheated) = 0

Policy iteration



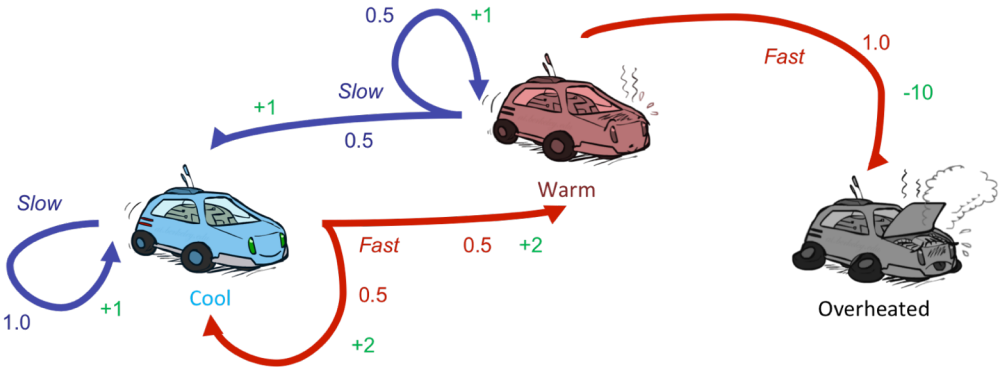
Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

$V_{k+1}(s) = R(s, \boldsymbol{\pi}(s)) + \boldsymbol{\Sigma}_{s'} [\gamma * T(s, \boldsymbol{\pi}(s), s')V_k(s')]$

$\boldsymbol{\pi}_0$	Fast	Fast	-
$\boldsymbol{\pi}_1$			
$\boldsymbol{\pi}_2$			
$\boldsymbol{\pi}_3$			

Policy iteration



π_0 Fast Fast -

π_1

π_2

π_3

Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

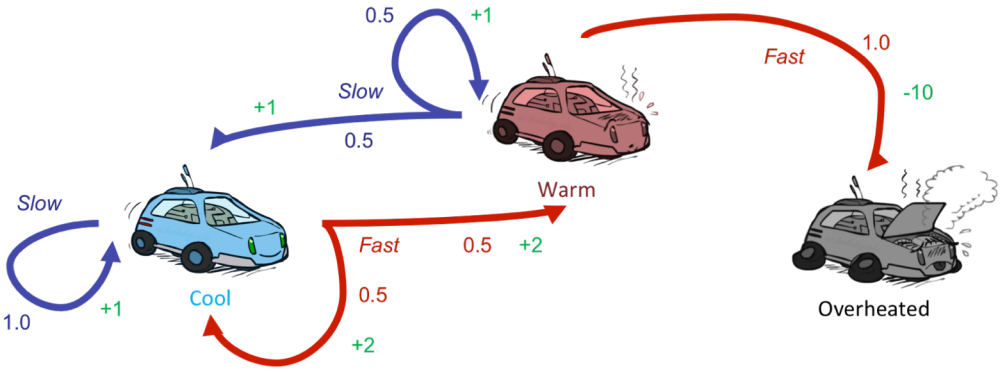
$V_{k+1}(s) = R(s, \pi(s)) + \Sigma_{s'} [\gamma * T(s, \pi(s), s')V_k(s')]$

Policy evaluation via linear programming:

V(cool) = 2 + 0.5 * 0.5 * V(cool) + 0.5 * 0.5 * V(warm)
V(warm) = -10 + 1.0 * 0.5 * V(overheat)
V(overheat) = 0

V(cool) = -2/3
V(warm) = -10
V(overheat) = 0

Policy iteration



Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

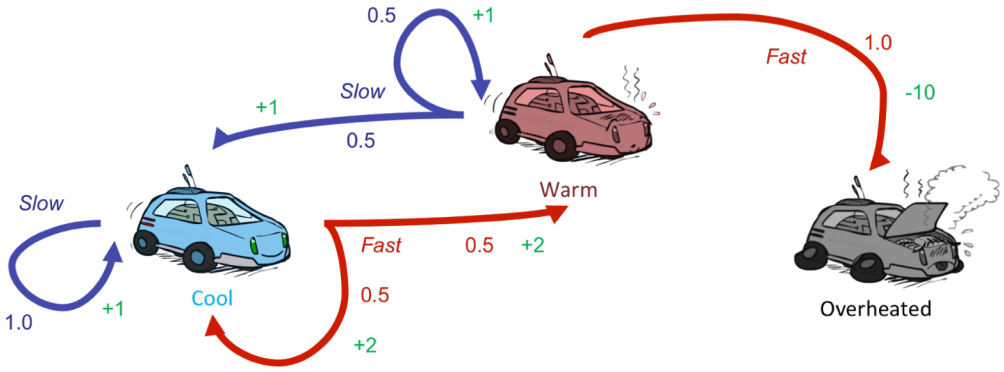
$V_{k+1}(s) = R(s, \boldsymbol{\pi}(s)) + \boldsymbol{\Sigma}_{s'} [\gamma * T(s, \boldsymbol{\pi}(s), s')V_k(s')]$

Policy improvement via policy extraction

V(cool) = -2/3
V(warm) = -10
V(overheat) = 0

$\boldsymbol{\pi}_0$	Fast	Fast	-
$\boldsymbol{\pi}_1$			
$\boldsymbol{\pi}_2$			
$\boldsymbol{\pi}_3$			

Policy iteration



π_0	Fast	Fast	-
π_1	Slow	Slow	-
π_2			
π_3			

Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s')\right]$$

$V_{k+1}(s) = R(s, \pi(s)) + \Sigma_{s'} [\gamma * T(s, \pi(s), s')V_k(s')]$

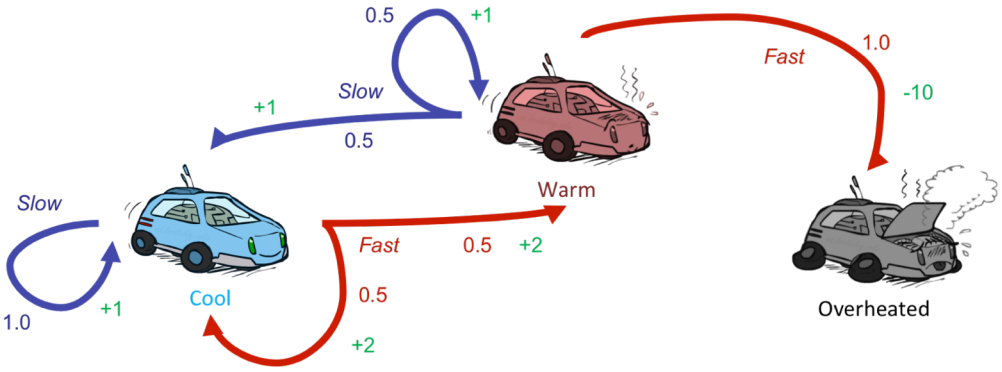
Policy improvement via policy extraction:

V(cool) = -2/3
V(warm) = -10
V(overheat) = 0

π(cool) = argmax{
slow: 1 + 1.0 * 0.5 * -2/3 = 2/3
fast: 2 + 0.5 * 0.5 * -2/3 + 0.5 * 0.5 * -10 = -2/3
} = slow

π(warm) = argmax{
slow: 1 + 0.5 * 0.5 * -2/3 + 0.5 * 0.5 * -10 = -5/3
fast: -10 + 1.0 * 0.5 * 0 = -10
} = slow

Policy iteration



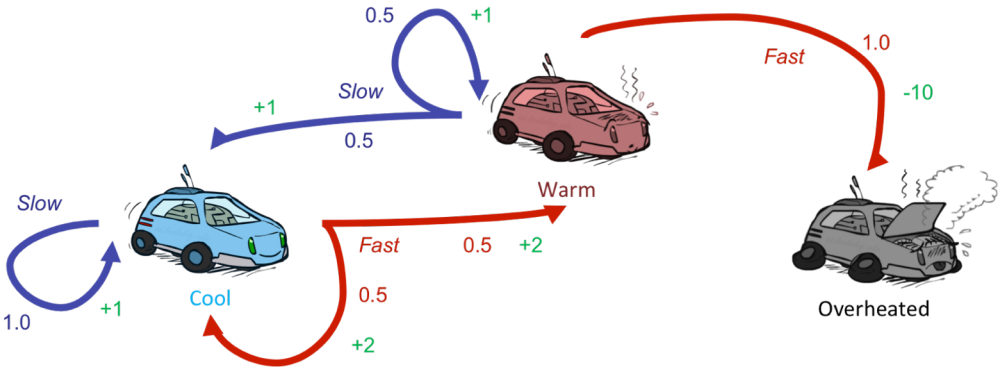
Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

$V_{k+1}(s) = R(s, \boldsymbol{\pi}(s)) + \boldsymbol{\Sigma}_{s'} [\gamma * T(s, \boldsymbol{\pi}(s), s')V_k(s')]$

$\boldsymbol{\pi}_0$	Fast	Fast	-
$\boldsymbol{\pi}_1$	Slow	Slow	-
$\boldsymbol{\pi}_2$			
$\boldsymbol{\pi}_3$			

Policy iteration



π_0 Fast Fast -

π_1 Slow Slow -

π_2

π_3

Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s')\right]$$

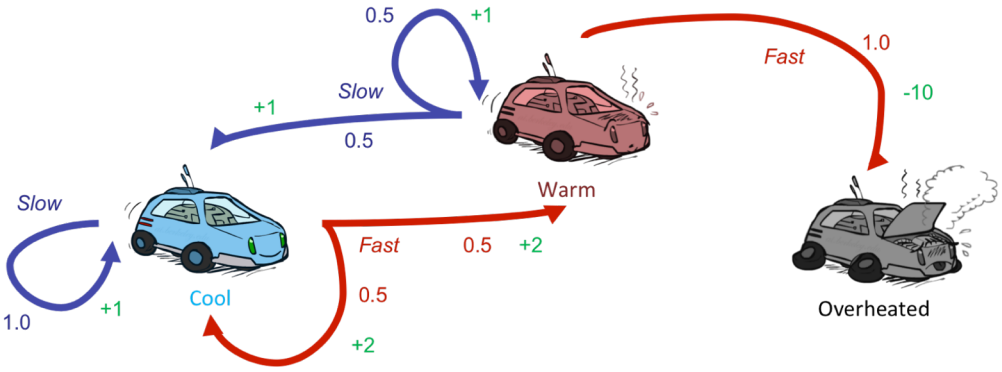
$V_{k+1}(s) = R(s, \pi(s)) + \Sigma_{s'} [\gamma * T(s, \pi(s), s')V_k(s')]$

Policy evaluation via linear programming:

V(cool) = 1 + 1.0 * 0.5 * V(cool)
V(warm) = 1 + 0.5 * 0.5 * V(cool) + 0.5 * 0.5 * V(warm)
V(overheat) = 0

V(cool) = 2
V(warm) = 2
V(overheat) = 0

Policy iteration



Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

$V_{k+1}(s) = R(s, \boldsymbol{\pi}(s)) + \boldsymbol{\Sigma}_{s'} [\gamma * T(s, \boldsymbol{\pi}(s), s')V_k(s')]$

Policy improvement via policy extraction:

V(cool) = 2
V(warm) = 2
V(overheat) = 0

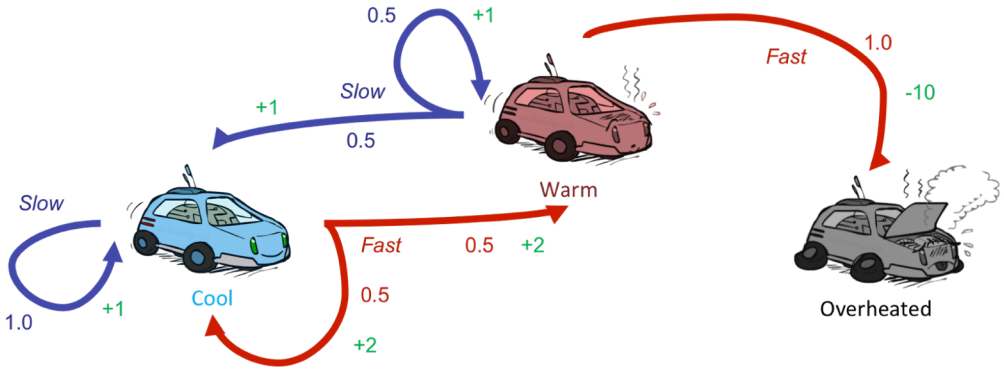
$\boldsymbol{\pi}_0$ Fast Fast -

$\boldsymbol{\pi}_1$ Slow Slow -

$\boldsymbol{\pi}_2$

$\boldsymbol{\pi}_3$

Policy iteration



π_0	Fast	Fast	-
π_1	Slow	Slow	-
π_2	Fast	Slow	-
π_3			

Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s')\right]$$

$V_{k+1}(s) = R(s, \pi(s)) + \Sigma_{s'} [\gamma * T(s, \pi(s), s')V_k(s')]$

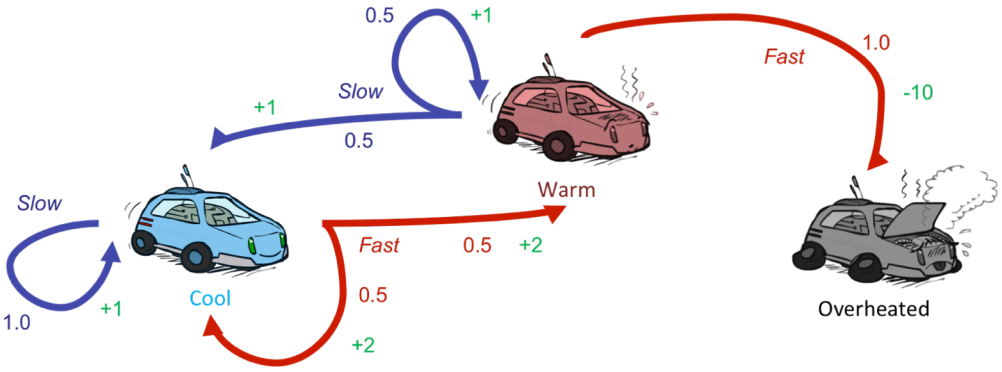
Policy improvement via policy extraction:

V(cool) = 2
V(warm) = 2
V(overheat) = 0

π(cool) = argmax{
slow: 1 + 1.0 * 0.5 * 2 = 2
fast: 2 + 0.5 * 0.5 * 2 + 0.5 * 0.5 * 2 = 3
} = fast

π(warm) = argmax{
slow: 1 + 0.5 * 0.5 * 2 + 0.5 * 0.5 * 2 = 2
fast: -10 + 1.0 * 0.5 * 0 = -10
} = slow

Policy iteration



Assuming a discount factor of 0.5

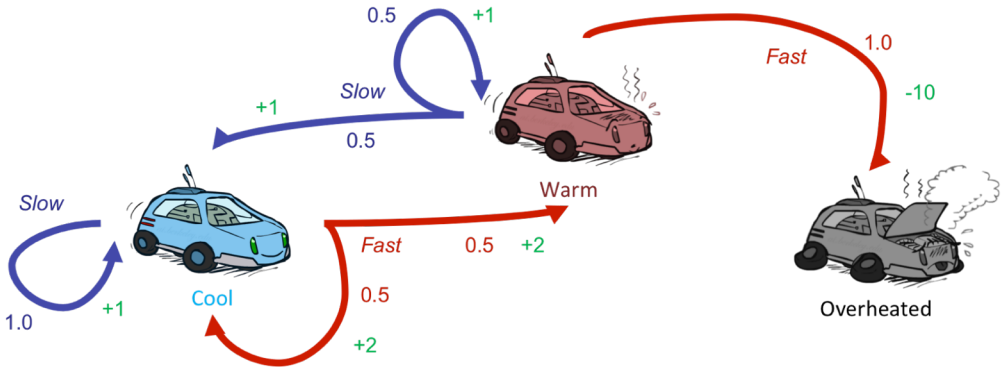
$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

$V_{k+1}(s) = R(s, \boldsymbol{\pi}(s)) + \boldsymbol{\Sigma}_{s'} [\gamma * T(s, \boldsymbol{\pi}(s), s')V_k(s')]$

Policy evaluation via linear programming:

$\boldsymbol{\pi}_0$	Fast	Fast	-
$\boldsymbol{\pi}_1$	Slow	Slow	-
$\boldsymbol{\pi}_2$	Fast	Slow	-
$\boldsymbol{\pi}_3$			

Policy iteration



π_0 Fast Fast -

π_1 Slow Slow -

π_2 Fast Slow -

π_3

Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

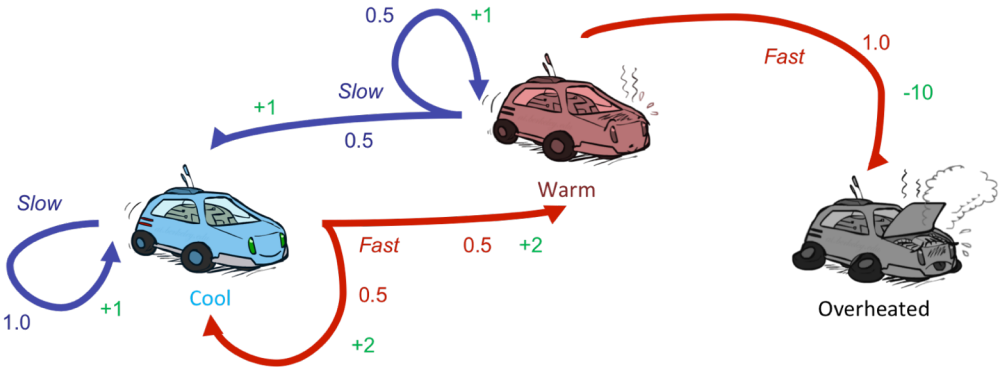
$V_{k+1}(s) = R(s, \pi(s)) + \Sigma_{s'} [\gamma * T(s, \pi(s), s')V_k(s')]$

Policy evaluation via linear programming:

V(cool) = 2 + 0.5 * 0.5 * V(cool) + 0.5 * 0.5 * V(warm)
V(warm) = 1 + 0.5 * 0.5 * V(cool) + 0.5 * 0.5 * V(warm)
V(overheated) = 0

V(cool) = 3.5
V(warm) = 2.5
V(overheat) = 0

Policy iteration



π_0 Fast Fast -

π_1 Slow Slow -

π_2 Fast Slow -

π_3

Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

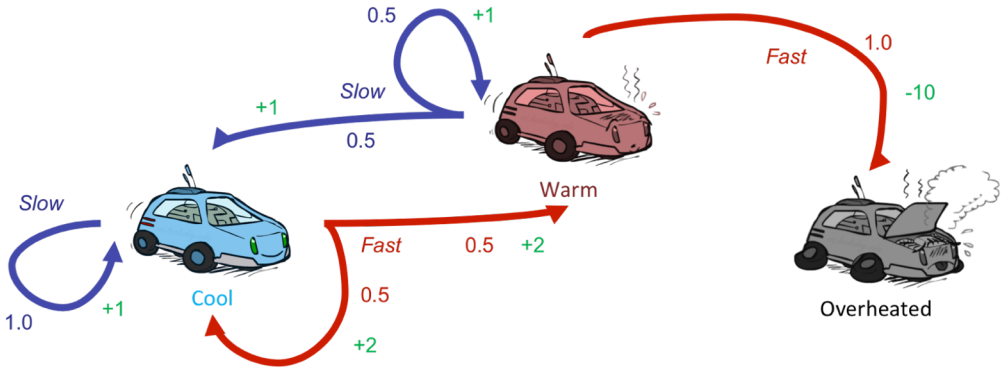
$V_{k+1}(s) = R(s, \pi(s)) + \Sigma_{s'} [\gamma * T(s, \pi(s), s')V_k(s')]$

Policy evaluation via linear programming:

V(cool) = 3.5
V(warm) = 2.5
V(overheat) = 0

Ira A. Fulton Schools of Engineering
ARIZONA STATE UNIVERSITY

Policy iteration



π_0	Fast	Fast	-
π_1	Slow	Slow	-
π_2	**Fast**	**Slow**	-
π_3	**Fast**	**Slow**	-

Assuming a discount factor of 0.5

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s')\right]$$

$V_{k+1}(s) = R(s, \pi(s)) + \Sigma_{s'} [\gamma * T(s, \pi(s), s')V_k(s')]$

Policy improvement via policy extraction:

V(cool) = 3.5
V(warm) = 2.5
V(overheat) = 0

π(cool) = argmax{
slow: 1 + 1.0 * 0.5 * 3.5 = 2.75
fast: 2 + 0.5 * 0.5 * 3.5 + 0.5 * 0.5 * 2.5 = 3.5
} = fast

π(warm) = argmax{
slow: 1 + 0.5 * 0.5 * 3.5 + 0.5 * 0.5 * 2.5 =2.5
fast: -10 + 1.0 * 0.5 * 0 = -10
} = slow