```
discount=0.99999, discountB=0.99):
```

```
structure = np.array([
['E',  'E',  'E',  'E'],
['E',  'E',  'E',  'T'],
['E',  'E',  'E',  'E'],
['T',  'E',  'T',  'E'],
['E',  'E',  'E',  'E']
])

# Labels of the states
label = np.array([
[(),      (),    ('c',),()],
[(),      (),    ('a',),('b',)],
[(),      (),    ('c',),()],
[('b',),  (),    ('a',),()],
[(),      ('c',), (),   ('c',)]
], dtype=np.object)
```

Q

```
Q = csrl.q_learning(T=100,K=100000)
```

```
[[1 3 0 2]
 [1 3 2 5]
 [1 3 0 2]
 [5 0 6 0]
 [3 0 0 0]]
```

```
[[1.   1.   0.   1.  ]
 [1.   1.   0.81 1.  ]
 [1.   1.   0.   1.  ]
 [1.   1.   1.   1.  ]
 [1.   0.   0.8  0.  ]]
```

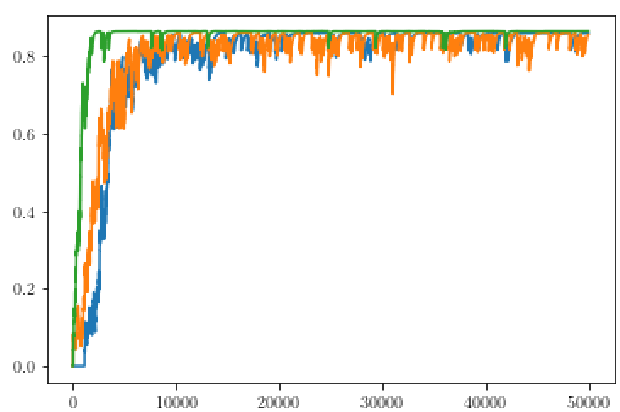
PG

```
T = 100
K = 50000
```

```
PG_state = np.prod(gamma_hist[0:t3:1])*(G_t_hist[t3] - V[state]) * Grad_Pi
PG[state][0:len(PG_state):1] += PG_state.flatten()
Grad_V[state] = G_t_hist[t3] - V[state]

theta = theta + 0.1*PG
V     = V     + alpha * Grad_V
```

```
[[1 3 0 1]
 [1 3 2 5]
 [1 3 0 0]
 [5 2 6 3]
 [0 0 0 0]]
[[0.85 0.85 0.   0.77]
 [0.86 0.85 0.66 0.86]
 [0.86 0.85 0.   0.72]
 [0.86 0.79 0.86 0.78]
 [0.77 0.   0.68 0.  ]]
```




```
T = 100
K = 50000
```

```
PG_state = np.prod(gamma_hist[0:t3:1])*(G_t_hist[t3]) * Grad_Pi
PG[state][0:len(PG_state):1] += PG_state.flatten()
Grad_V[state] = G_t_hist[t3] - V[state]

theta = theta + 0.2*PG
V     = V     + alpha * Grad_V
```

```
[[1 1 0 1]
 [2 2 2 5]
 [1 3 0 0]
 [1 2 6 3]
 [3 0 0 0]]
[[0.71 0.63 0.   0.77]
 [0.7  0.7  0.7  0.86]
 [0.81 0.79 0.   0.76]
 [0.81 0.78 0.86 0.77]
 [0.78 0.   0.66 0.  ]]
```