

UCB, par groupes de patients

On administre l'action choisie par UCB à un groupe de patients plutôt qu'à un seul patient.

Cela permet d'accélérer l'implémentation pratique des essais cliniques UCB.

In [1]:

```
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:

```
# fonctions générales
# [ ... ] (cf listings précédents)
```

In [4]:

```
#tirées de l'essai clinique des AVC
actions_q = [0.889, 0.907, 0.896, 0.892, 0.892, 0.893]
```

In [5]:

```
H = 20000
n_patients = 500 #nombre de patients par groupe

actions_Q = [0, 0, 0, 0, 0, 0]
actions_U = [0, 0, 0, 0, 0, 0]
actions_N = [1, 1, 1, 1, 1, 1] #evite la division par 0

actions_compteurs = [[0], [0], [0], [0], [0], [0]]
```

In [6]:

```
t = 1 #evite le log(0)
for groupe in range(H//n_patients):
    #calcul de  $U_t(a)$  pour toutes les actions a
    for a in range(len(actions_q)):
        actions_U[a] = np.sqrt(2 * np.log(t) / actions_N[a])

    #on choisit l'action à administrer au groupe
    a = indice_max(ajouter_listes(actions_Q, actions_U))

    #et on l'administre au groupe entier
    for i in range(n_patients):
        actions_N[a] += 1

    r = generer_reponse(actions_q, a)

    #moyenne incrémentale
    actions_Q[a] = actions_Q[a] + (1/actions_N[a]) * (r - actions_Q[a])

    actions_compteurs[a].append(actions_compteurs[a][-1] + 1)
    for b in range(len(actions_q)):
        if b != a:
            actions_compteurs[b].append(actions_compteurs[b][-1])

    t = t + n_patients
```

In [8]:

```
plt.plot(actions_compteurs[0])
plt.plot(actions_compteurs[1], label='YL')
plt.plot(actions_compteurs[2])
plt.plot(actions_compteurs[3])
plt.plot(actions_compteurs[4])
plt.plot(actions_compteurs[5])

plt.title('n_patients = ' + str(n_patients))
plt.xlabel('temps')
plt.ylabel('actions cumulées')
plt.legend()
plt.savefig('n_patients_' + str(500) + '.png', dpi=300)
```

