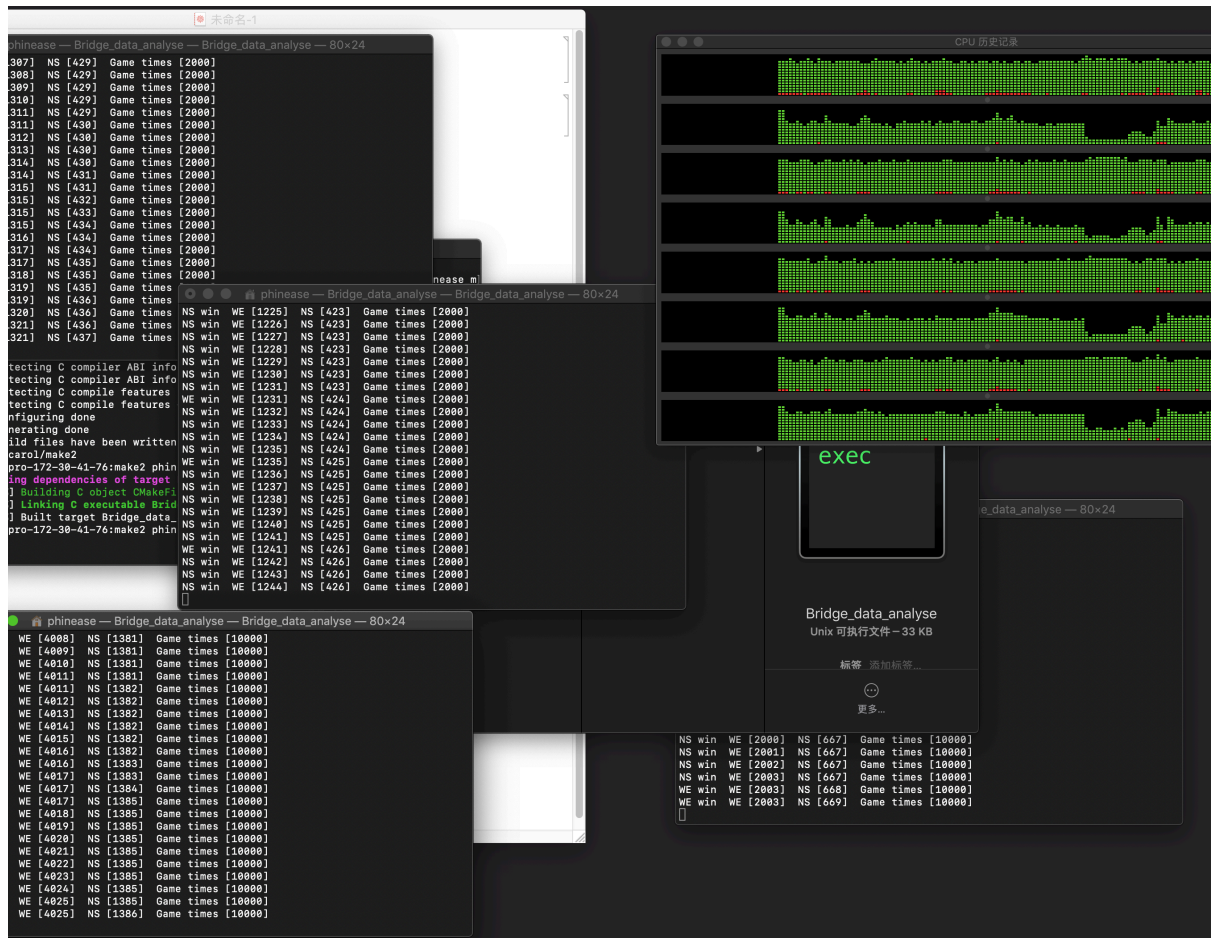


Analyse des données de Bridge



Pour étudier les caractéristiques de Bridge et de son joueur en Monte Carlo, j'ai fait quatre expérimentations pour répondre les questions suivantes:

1. Est-ce que la méthode de Monte Carlo est la solution imbattable contre les joueurs aléatoires? C'est à dire, est-il le meilleur pour gagner ce jeu?
2. Combien de simulation est-il suffit pour trouver une meilleure réponse? Ou peut-être il y a un nombre assez efficace?
3. Combien fois de jeu pour trouver des données stables qui a le sens statistique?
4. Sous le cas de la meilleure solution, qu'est-ce que le taux de réussite pour qu'on a trouvé pour le Monte Carlo contre un joueur aléatoire?
5. Est-ce que le commencement de jeu (le joueur de début) influence les données?
6. Comment la fréquence de simulation influence le Monte Carlo?

1.Est-ce que la méthode de Monte Carlo est la solution imbattable contre les joueurs aléatoires? C'est à dire, est-il le meilleur pour gagner ce jeu?

Condition: 7 scores pour gagner - commence par West - srand(time(NULL))

| 2 jeux par une fois | Premier jeu | Deuxième jeu | En cas de MC échoué tous les deux fois |
|---------------------|-------------|--------------|-----------------------------------------------------------|
| West | Monte Carlo | Aléatoire | Refaire avec même cartes, compte si MC est encore échoué. |
| North | Aléatoire | Monte Carlo | |
| East | Monte Carlo | Aléatoire | |
| South | Aléatoire | Monte Carlo | |

Simulation par tour en 2000 fois - 4000 jeux en total - 5 fois expérimentations — — —> 20000 jeux

Voici les données:

| | Échoué deux fois | Encore échoué |
|------------|--------------------|-------------------|
| Jeu fermé | 239/20000 (1.195%) | 17/20000 (0.085%) |
| Jeu ouvert | 148/20000 (0.74%) | 9/20000 (0.045%) |

Conclusion:

Monte Carlo n'est pas parfaite, il y a encore des fois que le joueur aléatoire gagne. Ça semble logique s'il n'y a pas de bug dans mon logiciel, parce que c'est juste une méthode statistique, sûrement qu'il y a des meilleurs solutions, et ça peut être trouvé au hasard par l'algorithme random. Et comme c'est déjà presque 1% que ça se passe. Alors ce n'est pas très étonnant d'apparaître deux fois consécutives. (Mais $1,195\% \times 1,195\%$ c'est 0,01428%, donc 0,085% est presque 6 fois. Donc je trouve ça un peu confuse.)

2.Combien de simulation est-il suffit pour trouver une meilleure réponse? Ou peut-être il y a un nombre assez efficace?

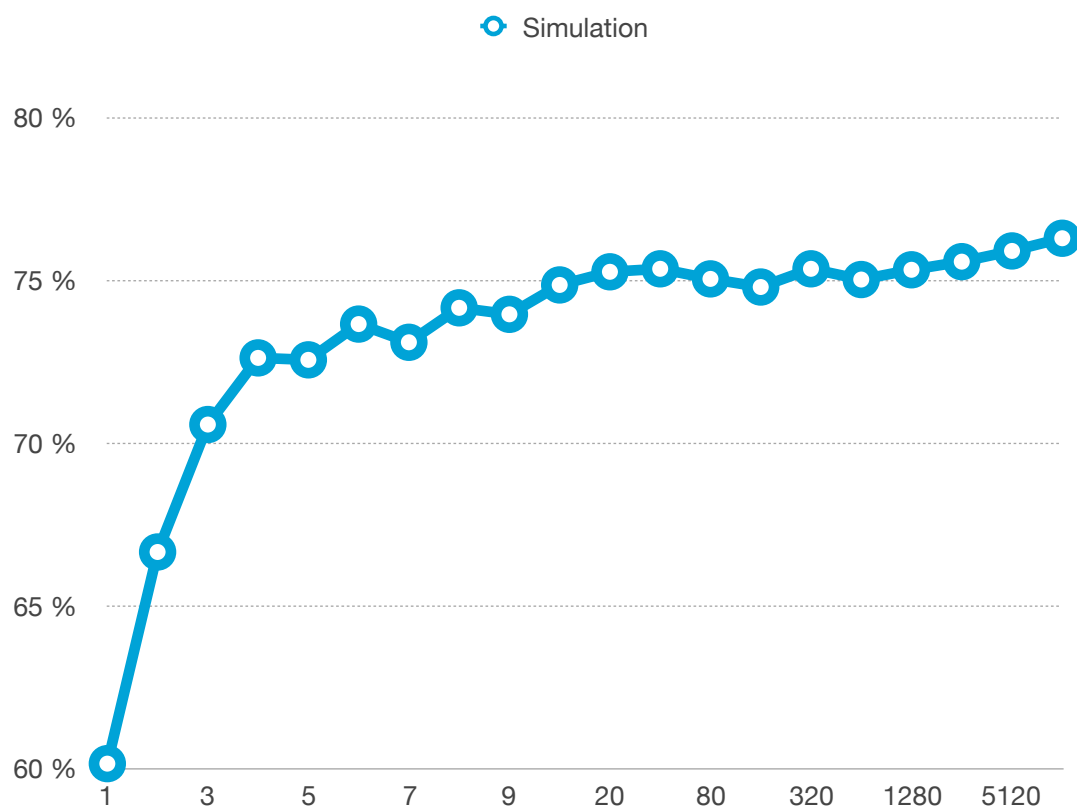
Condition: 7 scores pour gagner - commence par West - srand(time(NULL))
20000 jeux en total

Voici les données:

| Simulation | srand(time) - 20000 |
|------------|---------------------|
| 10 | 74,884 % |
| 20 | 75,288 % |
| 40 | 75,38 % |
| 80 | 75,07 % |
| 160 | 74,82 % |
| 320 | 75,38 % |
| 640 | 75,05 % |
| 1280 | 75,35 % |
| 2560 | 75,6 % |
| 5120 | 75,93 % |
| 10240 | 76,32 % |

(* la cellule orange est la réponse de Question 4)

Je trouve qu'il n'y a pas de grande changement depuis le 10 fois de simulations. Alors peut-être c'est déjà assez de faire moins de 10 fois. Donc j'ai fait des autres expérimentations.



| Simulation | srand(time) - 10000 |
|------------|---------------------|
| 1 | 60.16 % |

| Simulation | srand(time) - 10000 |
|------------|---------------------|
| 2 | 66,67 % |
| 3 | 70,59 % |
| 4 | 72,64 % |
| 5 | 72,58 % |
| 6 | 73,68 % |
| 7 | 73,12 % |
| 8 | 74,18 % |
| 9 | 73,98 % |

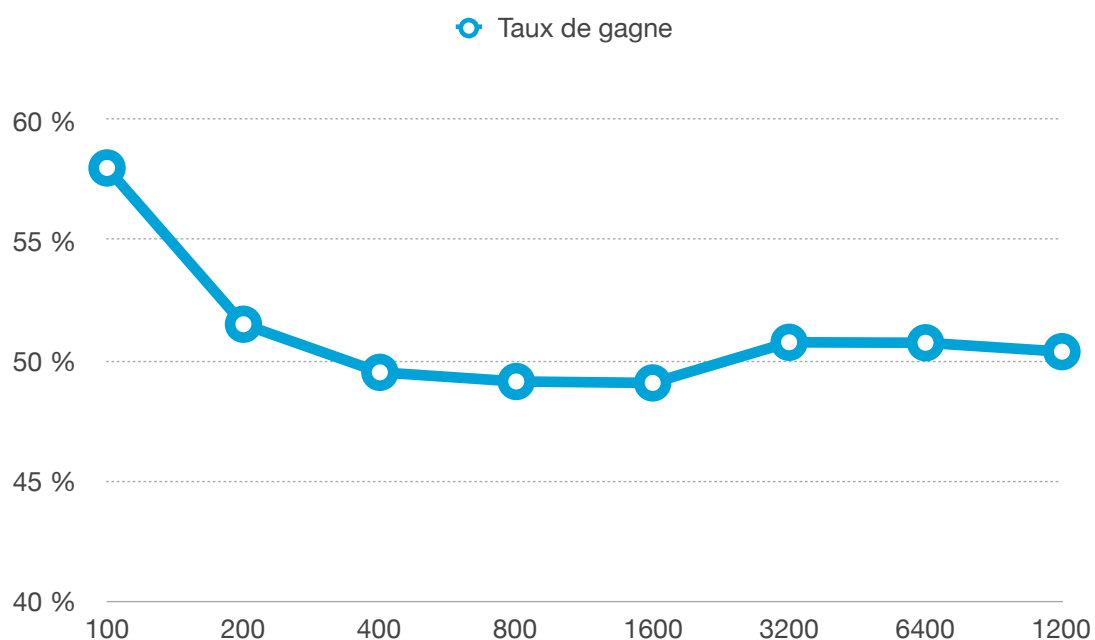
Conclusion:

Plus de 6 fois de simulation est déjà arrivée dans un point assez stable. Mais généralement c'est toujours meilleur si on augmente la simulation. Cela dépend de la ressource de temps et de la puissance de calcul.

3. Combien de jeu pour trouver une limite stable de donnée en moyenne?

Condition: 7 scores pour gagner - commence par West - srand(1)

Voici les données:



| Jeux | Joueur aléatoire * 4 |
|-------|------------------------------|
| 100 | 53 % < 58 % < 63 % |
| 200 | 47.96 % < 51.5 % < 55.03 % |
| 400 | 47 % < 49.5 % < 52 % |
| 800 | 47.352 % < 49.12 % < 50.88 % |
| 1600 | 47.81 % < 49.06 % < 50.31 % |
| 3200 | 49.86 % < 50.75 % < 51.63 % |
| 6400 | 50.1 % < 50.73 % < 51.35 % |
| 10000 | 49.86 % < 50.36 % < 50.86 % |

Conclusion:

Pour une précision de 10^{-1} , c'est 10000 fois de jeux pour y arriver. Mais je pense que 800 de jeux est déjà assez pour la plupart de cas.

5. Est-ce que le commencement de jeu (le joueur de début) influence les données?

Condition: 7 scores pour gagner - commence par West - srand(1)

10000 jeux

| Taux de gagne | West - East |
|------------------------|-------------|
| Commence par West | 50.36 % |
| Commence aléatoirement | 49,36 % |

Conclusion:

Il y a très peu d'influence.

6. Comment la fréquence de simulation influence le Monte Carlo?

Condition: 7 scores pour gagner - commence par West - srand(1)

10000 jeux - 4 joueurs Monte Carlo - WE 2 fois de simulations que NS

| Simulation (WE/NS) | WE | NS |
|--------------------|---------|---------|
| 200/100 | 51,35 % | 48,65 % |
| 100/50 | 51,33 % | 48,67 % |
| 50/25 | 51,53 % | 48,47 % |
| 24/12 | 51,56 % | 48,44 % |
| 12/6 | 52,99 % | 47,01 % |

| Simulation (WE/NS) | WE | NS |
|--------------------|---------|---------|
| 6/3 | 54,17 % | 45,83 % |
| 2/1 | 56,43 % | 43,57 % |
| 10/1 | 60,71 % | 39,29 % |

Conclusion:

Il n'y a que de grande influence si le nombre est inférieur de 5. Si non, même si on a une différence de 10000, ça ne change pas beaucoup.