Combinatorial Game - Summary-01

I summarized three parts from the paper I read:

1>How to generate the game tree. Apparently, the game tree will be big, if we generate too many trees this means it will waste much storage. So I think we should reuse the game tree cover the old value with the new one. In addition, we can also put some attention on how to store it in the physical memory.

2> Calculate the value after made the decision. It is better to have a table such as *Table1* about the gain for both sides with their step influence. Maybe we can calculate the value using the following formulation.

Table1 Example

|  |  |  |
| --- | --- | --- |
|  | User1 | User2 |
| 三三 | 10 | 15 |
| 三四 | 9 | 20 |

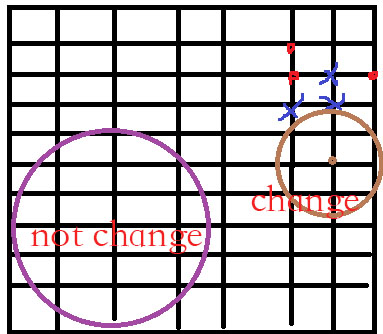
G(i, j) = 10 + .. +; (i, j) is the position.

3> Evaluate the state. The evaluation function should be efficient, it should not spend too much time to calculate the state. Maybe we can calculate the value using the following formulation or another. Obversely, there will be some local values do not change e.g. *Pic1*, so it is a chance to optimize it by not recalculate it.

MaxG1 = max{G(i, j) | 1<=i<=n, 1<=j<=m}

MaxG2 = max{G(i, j) | 1<=i<=n, 1<=j<=m}

EvaluateFunc(state) = MaxG1 – MaxG2



Pic1 Example

Reference :

[1] https://en.wikipedia.org/wiki/Combinatorial\_search

[2] https://en.wikipedia.org/wiki/Game\_tree

[3] 五子棋人机对战系统设计. http://xueshu.baidu.com/s?wd=paperuri%3A%287aaa9d329f022bd0be042f5725b76b63%29&filter=sc\_long\_sign&tn=SE\_xueshusource\_2kduw22v&sc\_vurl=http%3A%2F%2Fwww.doc88.com%2Fp-1012222839583.html&ie=utf-8&sc\_us=15010093921818038321

[4] 一种新的博弈树搜索算法及其应用研究. http://xueshu.baidu.com/s?wd=paperuri%3A%28e6f3dda28e73c44d52ec6d7d5ffd7a86%29&filter=sc\_long\_sign&tn=SE\_xueshusource\_2kduw22v&sc\_vurl=http%3A%2F%2Fwww.doc88.com%2Fp-9713611367961.html&ie=utf-8&sc\_us=5984495121303138993