

Solving Mismatch by Using Stochastic Cellular Automata

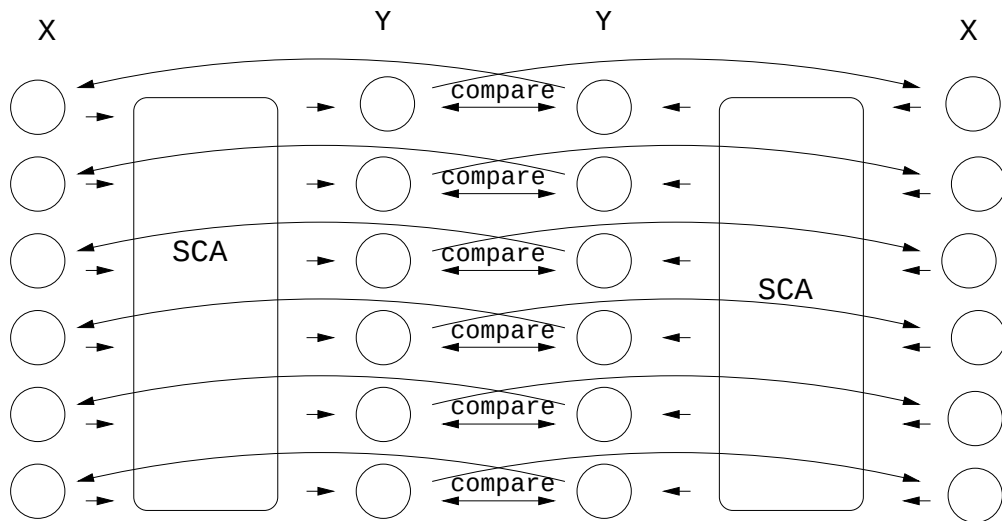
August 28, 2002

1 摘要

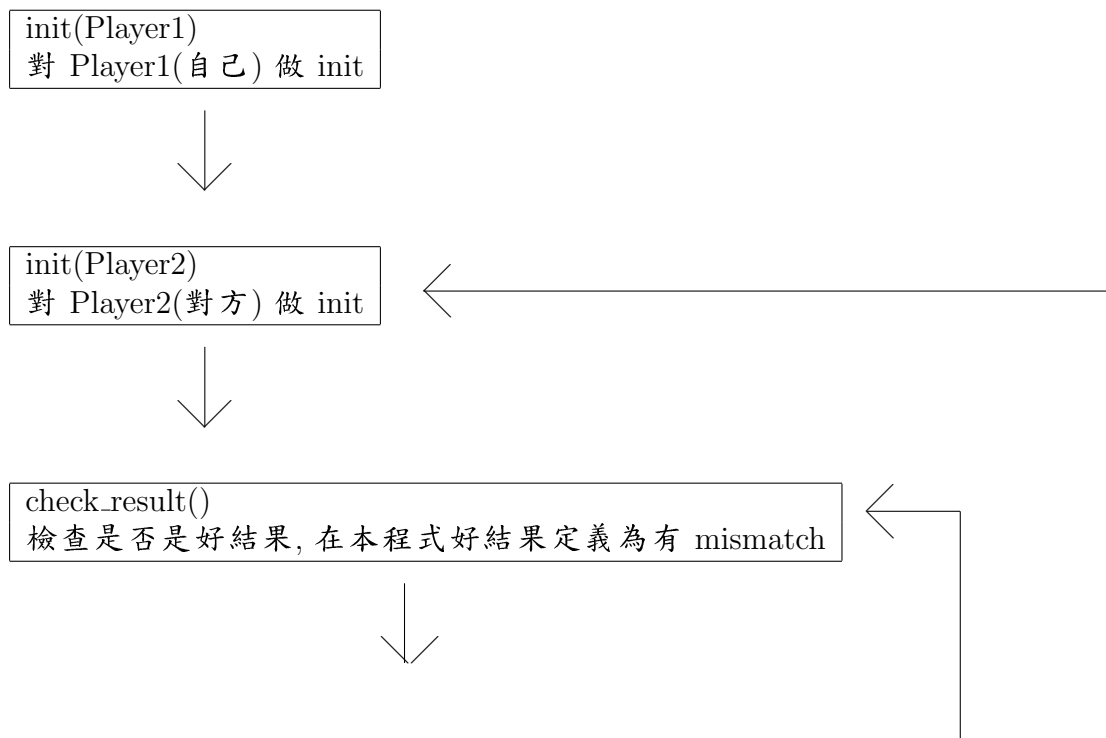
mismatch是由Stuart Kauffman [1] 所提出的. 目的是希望能夠由兩個一樣的程式共同合作, 最後 output 出完全不一樣的結果. 其中這個程式只能根據自己的 output 和對方的 output 來決定下次自己的 output. 在Stuart Kauffman 定的 mismatch problem 中, output 是給定 6 個. 我們使用 stochastic cellular automata(SCA) 來解決這個問題.

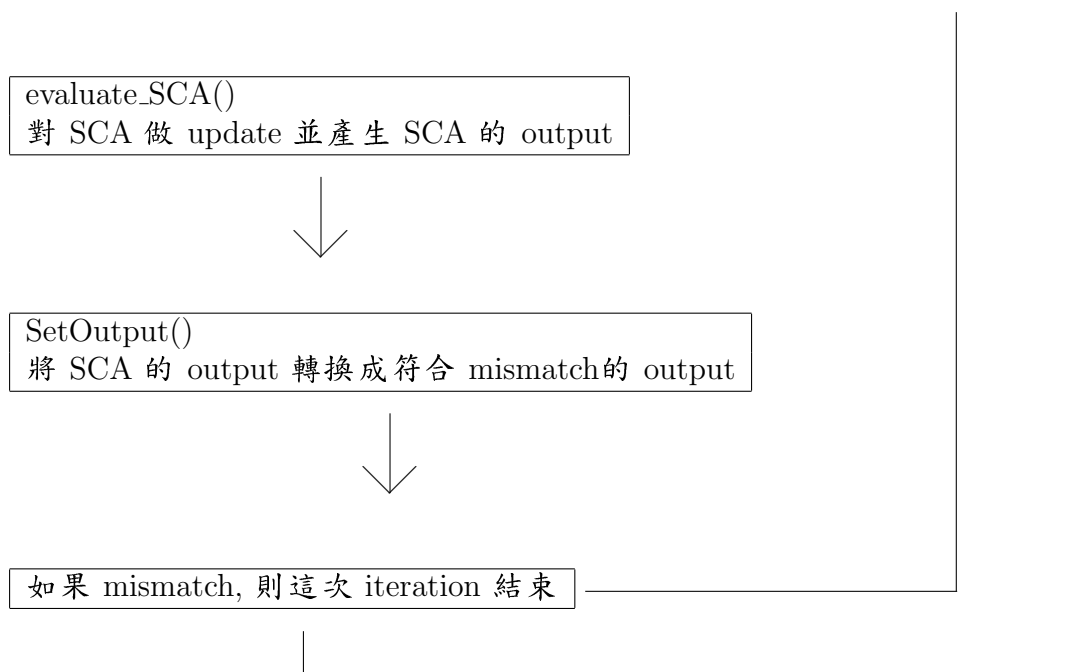
2 大略解說

Mismatch.cpp 是整個執行 mismatch problem 的環境. MismatchPlayer.cc 是 mismatch problem 的那個一樣的程式. MismatchCA.cc 是我們使用的 SCA 程式. 我們利用 SCA 來解決 mismatch problem. 其中 SCA 的 input 是對方的 output, SCA 的 output 是自己的 output. SCA 好的結果是指完全 mismatch, 不好的結果是指沒有完全 mismatch 到. 目的是希望當遇到不同的對方時 (對方的 SCA 重新做 initialize), 能夠在一定的 iteration 內完成. 一開始先對自己的部份做 initialize 的動作. 在每次遇到不同對方的時候, 先對對方做 initialize 的動作, 然後互相比較其 output, 然後再用對方的 output 來 update 自己的 SCA 並產生自己的 output, 同樣的對方也是用我們自己的 output 來 update 對方的 SCA 並產生對方的 output, 最後如果 mismatch 了, 就是與對方完成 mismatch problem, 就換另外一個對方重新來 (不改變我們的 SCA, 改變對方的 SCA), 直到完成我們預期要跟多少個對方完成 mismatch problem 為止. 示意圖如下:



3 流程圖





4 程式/公式對照表

公式	程式
外顯的 output SCA	output in class Player brain

參考文獻

- [1] Stuart A. Kauffman. *The Origins of Order*, pages 221–226. Oxford University Press, 1993.