演化式計算

期末專題報告(初稿)

題目: 以禁忌搜尋法解決工作匹配與排程問題



開課系所：電通系

指導教授：江傳文

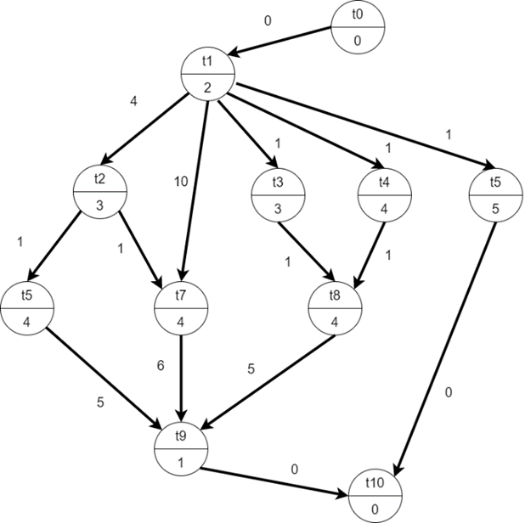
組員: 電通4B 0551076 黃泰源

繳交日期：2020/06/11**摘要**

工作匹配與排程問題是一類組合最佳化問題，此報告介紹使用禁忌搜尋演算法求解此問題，並在一定迭代次數內找到合適的解。

**問題描述**

工作匹配與排程問題定義為: 給予一各工作的執行優先順序(如圖一)並須將各工作匹配到各個處理器上；處理器對於各工作處理時間可能會相同或是有所不同，最後能使執行時間(makespan)越短越好。



圖一、測試案例test\_00

對於工作的執行優先順序可以將其編碼成一有序排程字串，而處理器的匹配可以將其編碼成一無序匹配字串。下表是圖一相對應的字串範例(處理器個數為4):

表一、 排程字串

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 順序 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 工作 | 1 | 3 | 2 | 4 | 7 | 5 | 6 | 8 | 9 | 10 |

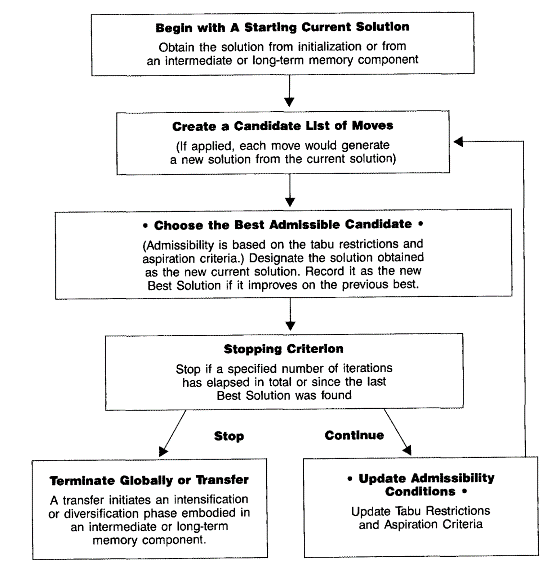
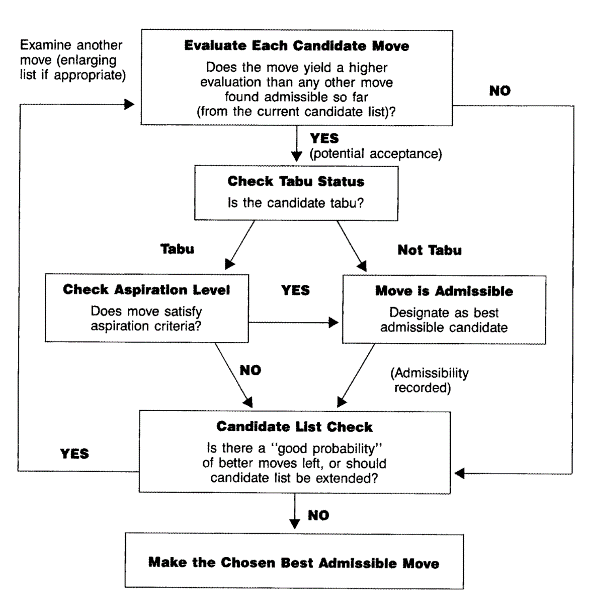
表二、 匹配字串

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 工作 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 處理器 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 2 | 1 | 1 |

禁忌搜尋演算法會針對這兩個字串進行擾動以求得最佳解，擾動方式會於後面章節詳細介紹。

**禁忌搜尋演算法**

禁忌搜尋演算法希望藉由tabu list的記憶機制來避免反覆搜索區域最佳解，並希望能提升搜索空間的多樣性。禁忌搜尋演算法的基本流程如下圖:



圖二、Tabu Search流程圖 圖三、尋找允許的解流程

**實作細節說明**

**擾動方式**

針對問題的編碼方式擾動方式一共有兩種:

* 排程字串擾動方式為task i往前插入task j，如果task j的優先權比task i大的話則不執行此擾動，並從解空間找下一個解。
* 匹配字串擾動方式為隨機抽取處理器但不跟現在匹配的處理器一樣。

**解空間與鄰居的選擇**

因為每個工作可以決定是否要往前插入或替換處理器，所以解空間為兩倍工作的數量。鄰居的選擇則是隨機抽取並放入鄰居池，如果在評價所有鄰居之後發現沒有合適的解(admissible solution)則從解空間再隨機取出新的鄰居直到解空間沒有解。

**Tabu List設計**

為了避免task j被task i超前後又重新超前回去:

* 排程字串的tabu list紀錄被插入的task j，task j不可再往前插入n個回合。

為了希望task i能夠使用其他處理器:

* 匹配字串的tabu list則是task i不可以在使用被替換掉的處理器持續n個回合。

以上兩個tabu list是分開紀錄的，更新則是取決於當下合適的解(admissible solution)所使用的擾動來更新該擾動所屬的tabu list。

**Aspiration Critetria**

如果有一解是禁止的解，但其品質比現行最佳還要好，可以使tabu list失效，並成為合適的解(admissible solution)並取代最佳解。

**Tabu list的長度與鄰居池的大小**

tabu list的長度是以工作數量乘以一介於0到1的浮點數;各tabu list長度由各浮點數變數控制，而鄰居池大小是以解空間大小乘以一介於0到1的浮點數。這些浮點數都是由使用者決定大小。

**實驗結果**

下表為測試四個範例各經五次試驗所得之實驗結果:

表三、實驗結果(實驗所使用電腦的cpu: intel i5-6300HQ 3.2GHz ram: 8GB)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| filename | Best | Worst | Avg. | sd | Avg.time(ms) |
| n4\_00.dag | 440 | 510 | 472 | 22.2711 | 8.8 |
| n4\_02.dag | 425.9 | 515 | 472.32 | 35.1636 | 8.8 |
| n4\_04.dag | 445.7 | 470.2 | 457.38 | 11.0485 | 9 |
| n4\_06.dag | 436.6 | 467.4 | 451.68 | 12.9858 | 8.8 |

表四、演算法相關參數設定

|  |  |  |  |
| --- | --- | --- | --- |
| 迭代數 | 插入工作tabu list 控制變數 | 處理器tabu list  控制變數 | 鄰居池的長度  控制變數 |
| 200 | 0.7 | 0.5 | 0.3 |

上述實驗結果從標準差欄位反映了每次試驗變化都很大，且最差的解跟最好的解有一段差距。

**結果與討論**

由上述實驗可以針對演算法在解的品質穩定度上面進行改進，並嘗試提高最差的解的品質。改善著手的部分可以藉由觀察控制變數對解品質的影響，以及鄰居池的選擇方式是否要改進，tabu list的規則改變或更新方式的改變。利用剩下的兩個禮拜對演算法進行調整。