

# PENGGUNAAN *ALGORITHMHA GENETIK* UNTUK MENYELESAIKAN MASALAH PENJADUALAN

## Kajian Kes: Jadual Waktu Sekolah

### ABSTRAK

Penjadualan waktu sekolah adalah masalah yang berhubung dengan jadual mingguan bagi kesemua kelas disesebuah sekolah. Masalah ini merangkumi cara bagaimana menentukan satu-satu matapelajaran diajar mengikut period yang akan ditetapkan dengan mengambilkira tiada guru atau kelas bertembung dalam satu-satu masa. Kehendak asas ini biasanya akan melibatkan banyak lagi perkara lain seperti masa pengajaran bagi sesebuah matapelajaran mestilah tersebar dalam masa seminggu.

Tujuan utama kertaskerja ini adalah untuk memahami had-had *keupayaan Algorithma Genetik* dan potensinya dalam menanggapi masalah yang mempunyai kekangan yang tinggi, di mana dalam masalah pengoptimuman, sebarang perubahan kecil kepada penyelesaian yang *feasible* akan menjana kepada satu penyelesaian yang tidak *feasible*. Sebagai ujian, masalah jadual waktu telah dipilih, di mana ianya diketahui adalah *NP-hard*, tetapi ianya amat praktikal kerana telah banyak penyelidikan telah dibuat yang berkaitan dengannya.

## I. Pengenalan

### Masalah Penjadualan

Penjadualan adalah dikenalpasti sebagai masalah yang paling sukar untuk diselesaikan tetapi pada masa yang sama ianya adalah masalah yang paling menarik minat para penyelidik dalam bidang Penyelidikan Operasi dan Kepintaran Buatan. Masalah ini diketahui juga sebagai *NP-complete*, di mana masa yang diperlukan untuk menyelesaikan masalah tersebut adalah dijangkakan meningkat secara *exponen* berbanding dengan saiznya [EK95].

*Algorithma Genetik* (AG) adalah mirip kepada keadaan semulajadi yang terdapat dalam pembiakan haiwan atau tumbuhan. Individu atau kromosom adalah dipilih daripada populasi awal atau dipanggil sebagai *parents* dan akan dibiakkan (mated) melalui proses penyilangan (*crossover*) bagi membentuk individu yang baru (*child*). Individu yang baru lahir ini akan mewarisi ciri-ciri *parents* (*genes*) dan digabungkan bagi membentuk kromosom yang baru. Kromosom tersebut akan berubah jika berlaku mutasi (*mutation*) di mana ciri-ciri yang berlainan dengan *parents* akan terhasil. Tahap kecergasan kromosom akan diuji, jika didapati sesuai dengan persekitaran, maka peluang untuk terus hidup adalah cerah, tetapi jika sebaliknya, ia akan mati dan terkeluar dari generasi yang akan datang [AA92].

1. setiap guru dan kelas mesti dijadualkan mengikut jumlah period yang telah ditetapkan.
2. tidak boleh lebih dari seorang guru dalam sesebuah kelas.
3. tidak boleh ada guru berada di dua buah kelas pada masa yang sama.
4. tidak ada period yang tidak mempunyai guru dalam sesebuah kelas.
5. Subjek yang diajar serentak di dalam kelas yang berlainan atau beberapa kelas digabungkan perlu disetkan mengikut period yang dikehendaki.

Semua maklumat GSK adalah dislotkan secara rawak mengikut kelas kod 1 hinggalah kelas kod terakhir. Populasi awal ini mengandungi matrik  $R$  (iaitu  $m \times n$  matrik dimana  $r_{ij} \in \text{GSK}$ ) di mana setiap baris adalah mewakili guru dan lajur adalah mewakili slot dari 1 hingga 45 [Colorni, 91]. Setiap elemen  $r_{ij}$  matrik  $R$  adalah gene, dan nilai allele adalah bergantung kepada maklumat dalam GSK. Dari populasi awal atau matrik  $R$  inilah akan digunakan oleh AG dalam mencari jadual waktu yang optimal.

Tujuan kriteria-kriteria tersebut ialah bagi menjamin setiap jadual yang dijana adalah bukan sahaja *feasible*, iaitu mirip kepada jadual sebenar. Ini adalah kerana jumlah slot masa yang tetap bagi sesebuah kelas adalah 9 period sehari, dan subjek yang perlu tersebar mengikut period yang telah ditetapkan, akan memudahkan lagi penggunaan Operator Genetik kerana kesemua GSK telah lengkap berada dalam jadual. Penggunaan GSK yang dijana dari maklumat yang diperlukan akan mengurangkan *complexity* tugas penjadualan [Ab,91].

## II. Analisa Masalah dan Perlaksanaan

Keberkesanan penggunaan AG adalah berdasarkan kepada teknik pengiraan kos objektif atau kos fungsi bagi matrik  $R$  yang ditentukan dengan tepat bagi menilai satu-satu jadualwaktu. Fungsi ini adalah menjadi kayu pengukur sejauh mana kualiti satu-satu jadualwaktu dihasilkan. Pengiraan fungsi kos adalah berdasarkan beberapa jenis kekangan :

A. **Kekangan Utama** : mengira jumlah *infeasibilities* ( $A$ ) dalam matrik  $R$  seperti:

- 1 . tidak boleh lebih dari seorang guru dalam sesebuah kelas.
2. tidak boleh ada seorang guru berada di dua buah kelas pada masa yang sama.
3. tidak ada period yang tidak mempunyai guru dalam sesebuah kelas.

### B. Kekangan Rendah

1. tidak lebih dari 6 period pengajaran sehari bagi setiap guru (pemberat: B1).
2. tidak boleh guru yang sama pada period terakhir setiap hari (pemberat: B2).
3. subjek tersebar secara uniform dalam masa seminggu (pemberat: B3).
4. subjek yang mempunyai period dua masa (pemberat: B4).
5. tidak kurang dari 2 period pengajaran sehari bagi setiap guru (pemberat: B5).
6. masa rehat bagi selepas setiap pengajaran (pemberat: B6).

## **JadualWaktu**

Jadualwaktu adalah masalah penjadualan yang sangat istimewa kerana sifatnya yang mempunyai kekangan yang tinggi. Teknik carian yang menggunakan kaedah heuristik (kaedah cuba jaya) telah banyak digunakan dengan menghasilkan kejayaan yang baik tetapi ianya tidak menjamin penyelesaian yang optimal [TB95].

Sebagai perbandingan, pendekatan AG dapat menyelesaikan seberapa banyak kekangan yang hampir membawa kepada penyelesaian yang optimal. Ini dapat dilakukan kerana AG akan membuat carian berdasarkan ruang yang terdapat dalam jadualwaktu yang *feasible*.

## **Masalah JadualWaktu Sekolah**

Kajian kes ini dibuat di Sekolah Menengah Jitra, Kedah yang mempunyai sesi pagi dan petang. Sebagai kajian awal dalam penyelidikan ini, kami hanya menumpukan kepada penjanaan jadualwaktu bagi tingkatan 3 hingga 5 yang dianggap paling kritikal oleh guru yang bertanggungjawab dalam masalah ini.

Sebanyak 28 buah kelas dan seramai 60 orang guru mengajar dalam pelbagai subjek dalam tempoh 45 period seminggu. Hasil perbincangan yang diadakan dengan guru terbabit, adalah lebih baik jika maklumat awal bagi penjanaan ini diambil dari maklumat yang sedia ada. Maklumat Guru-Subjek-Kelas (GSK) adalah data manual yang di buat berdasarkan kepada pengalaman, kekananan dan mengambilkira semua aspek keperluan sekolah seperti nisbah bilangan murid dan guru serta lain-lain faktor.

Maklumat GSK ini mempunyai kod yang unik bagi setiap GSK yang dikenalpasti. Contohnya, kod GSK 1 mempunyai maklumat kod guru 18 (Ali), mengajar dikelas kod 3 (3Agama3) dan mengajar subjek kod 2 (Matematik). Maklumat GSK ini mestilah di buat berdasarkan jumlah masa yang diambil oleh setiap guru mengajar subjek tersebut. Contohnya, bagi subjek Matematik di Tingkatan Tiga, ianya mempunyai 5 period seminggu dengan susunan 2-2-1. Oleh itu GSK dipecahkan kepada dua kod iaitu kod GSK 2, untuk subjek matematik dua period dan kod GSK 3, untuk subjek matematik satu period. Tujuannya ialah untuk mengelakkan kesukaran menjana jadual yang membabitkan jumlah masa yang tidak sama di mana subjek yang mempunyai dua period akan diberi keutamaan yang tinggi. Selain dari itu maklumat kelas gabung dan dijalankan secara serentak serta kelas makmal juga perlu dimasukkan dalam GSK.

## **Populasi Awal**

Penjanaan populasi awal adalah secara rawak, yang dibuat oleh [CDM91] adalah untuk mendapatkan sebuah jadual waktu yang *feasible* apabila memenuhi beberapa kriteria berikut:

### C. Kekangan Kehendak Guru

1. guru yang tidak boleh mengajar di slot yang awal (pemberat: D1).
2. guru yang tidak boleh mengajar di slot yang terakhir (pemberat: D2)

Fungsi objektif bagi jadual waktu matrik R adalah :

$$z(\mathbf{R}) = \mathbf{P} \cdot \mathbf{A} + \mathbf{Q1} \cdot \mathbf{B} + \mathbf{Q2} \cdot \mathbf{D}$$

Parameter bagi objektif fungsi ialah :

P= 1000	B1=2	B5=1
Q1=100	B2=1	B6=5
Q2= 300	B3=1	D1=0.5
	B4=6	D2=2

### 111. Operator Genetik

#### 1. Penyilangan (Crossover)

Setelah populasi awal dijana, fungsi kos atau *fitness function* (*f.f*) akan dikira mengikut persekitaran atau kekangan yang telah dikenalpasti. Bergantung kepada jumlah generasi yang akan dijana, setiap populasi awal akan menjadi populasi lama apabila sudah terbentuk populasi baru melalui fungsi operator genetik [CDM90].

Algorithma:

```
pilih secara rawak dua individu (parents) dari populasi awal
for setiap individu dan dengan kebarangkalian  $P_c$  do
  begin
    kira l.f.f (local fitness function) setiap baris (guru) dalam kedua-dua
    individu.
    jana dua anak (child) berdasarkan dua individu :
      (anak yang pertama diambil dari individu yang mempunyai f.f
      yang terendah (parent yang terbaik) dan pilih baris yang
      mempunyai l.f.f yang terendah. Baris yang seterusnya diambil
      dari individu yang lemah, iaitu f.f yang tinggi).

      (anak yang kedua diambil dari baris yang tidak digunakan dari
      kedua-dua individu).
  end
```

Proses penyilangan ini adalah berbeza jika dibandingkan dengan teknik tradisional dalam AG yang dipopularkan oleh Goldberg [G089], di mana beliau menggunakan satu atau dua titik penyilangan bagi kedua-dua individu. Dalam algorithma ini, kami sependapat dengan [CDM90] bahawa, dengan memilih baris atau guru yang mempunyai l.f.f terendah, terbukti dapat menghasilkan anak yang lebih baik dan yang lebih penting tidak menyebabkan kerosakan yang teruk (tidak mewarisi ciri-ciri *parents*) kepada anak yang dihasilkan. Teknik ini menghasilkan anak yang masih mewarisi sedikit sebanyak sifat-sifat yang dimiliki oleh *parents*.

## 2. *Mutasi*

Operator ini berfungsi dengan setiap baris (guru) dalam individu mempunyai kebarangkalian untuk berlaku proses mutasi. Operator ini akan mengambil satu gene (GSK) secara rawak dan swap ( dengan gene yang berdekatan dalam baris atau kod guru yang sama). Dengan kata lain, dua GSK yang berdekatan akan bertukar tempat dalam jadual guru yang sama. Dalam ujian yang dilakukan, kami hanya menumpukan pemilihan gene yang mempunyai dua period dan memilih subjek yang tidak mempunyai kelas gabung atau serentak. Ini bagi mengelakkan lebih banyak kerosakkan pada anak yang dijana dan memberi kesan kepada fungsi kos.

## 3. *Filtering (Genetic Repair)*

Selepas operator genetik berfungsi, hasil ujian mendapati jadualwaktu yang terhasil adalah tidak *feasible* seperti mana yang dikehendaki mengikut kriteria dalam populasi awal. Oleh itu satu fungsi yang dikenali sebagai *filtering* digunakan untuk menjana semula GSK yang hilang atau ujud lebih dari satu GSK dalam setiap baris individu baru. Ini boleh dilakukan dengan melihat pada lajur matrik, di mana setiap kelas mesti hanya ujud sekali dalam satu-satu lajur.

Jika ujud lebih dari satu kelas, contohnya lajur 28 atau slot 28, terdapat kelas kod 3 pada baris guru kod 17 tetapi slot 28 juga didapati mempunyai kelas kod 3 pada baris guru kod 65. Ini menunjukkan 2 GSK atau ujud 2 guru dalam satu kelas pada masa yang sama. Jika lajur tidak diwakili oleh salah satu kod kelas, ini menunjukkan terdapat GSK yang hilang dari jadual. Oleh itu tujuan utama *filtering* ialah untuk menghasilkan jadualwaktu yang *feasible* dengan menjanakan semula GSK yang hilang atau ujud lebih dari satu pada satu-satu lajur matrik R.

## IV. Keputusan dan Kesimpulan

Perisian ini ditulis dengan menggunakan bahasa pengaturcaraan C dan menggunakan komputer peribadi 486 (IBM-PC compatible). Ini adalah bertujuan supaya perisian ini dapat digunakan dikebanyakan sekolah yang mempunyai komputer jenis tersebut. Masalah yang timbul ialah dari segi masa yang diambil untuk menghasilkan jadual yang benar-benar hampir kepada optimal akan memakan masa yang lama.

Ujian awal di buat berdasarkan sampel data yang kecil membabitkan 22 orang guru dan 8 buah kelas dengan memasukkan kesemua kekangan yang diperlukan. Ujian telah dilakukan berdasarkan kepada kriteria-kriteria berikut:

1. Probabiliti pnyilangan dan mutasi yang berlainan.
2. Kos bagi *infeasibiliti* disetkan tinggi (1000) dan paling rendah (3). Ini adalah untuk menguji sejauhmana carian tempatan (local search) memberi kesan dalam menghasilkan jadual yang *feasible* dalam keadaan kos *infeasibiliti* yang tinggi dan rendah.

Keputusan hasil dari ujian ini amat memberangsangkan, tetapi masih lagi memerlukan beberapa pengubahsuaian terutama dalam fungsi *genetic repair*. Ini adalah untuk mencari kaedah *filtering* yang paling meminimalkan atau tanpa memburukkan lagi kos kekangan rendah. Untuk ujian seterusnya, data penuh sekolah akan di uji.

## Rujukan

### I. Journals

- [AA92] D. Abramson and J. Abela. (1992). "A Parallel Genetic Algorithm for Solving the School Timetabling Problem", *15 Australian Computer Science Conference*, Hobart.
- [Ab91] D. Abramson. (1991). "Constructing School Timetables using Simulated Annealing: Sequential and Parallel Algorithms", *Management Science*, Vol. 37, No. 1, p 98-113.
- [CC95] F.Carmusciano and D. De Luca Cardillo. (1995). "A Simulated Annealing with Tabu List Algorithm for the School Timetable Problem", *ICPTAT '95*, p 231-243.
- [CDM90] A. Coloni, M. Dorigo and V. Maniezzo. (1990). "Genetic Algorithms: A New Approach to the Timetable Problem", *Lecture Notes in Computer Science - NATO ASI Series*, Vol. F 82, Combinatorial Optimization,(Ed. M. Akgul and others), Springer-Verlag, p 235-239.
- [CDM91] A. Coloni, M. Dorigo and V. Maniezzo. (1991). "Genetic Algorithms and HighlyConstrained Problems: The Timetable Case", *Proceeding of the First International Workshop on Parallel Problem Solving from Nature, Dortmund, Germany, Lecture Notes in Computer Science 496, Springer-Verlag*, p 55-59.
- [CDM96] A. Coloni, M. Dorigo and V. Maniezzo.(1996). "Metaheuristics for High-School Timetabling", submitted to *Computational Optimization and Application Journal*.
- [CK95] T.B. Cooper and J.H. Kingston. (1995). "A Program for Constructing High School Timetables", *ICPTAT '95*, p 132-143.
- [EK95] W. Erben and J. Kepler. (1995). " A Genetic Algorithm solving a Weekly Course-Timetabling Problem". *ICPTAT '95*, p 21-32.
- [LPP95] H. Luchian, B. Paechter and M. Petriuc. (1995). "Fine-Tuning a Genetic Algorithm for the General Timetable Problem", *ICPTAT '95*, p 435-442.
- [TB95] Ahmad Tajudin Khader and J.T. Buchanan. (1995). "School Timetabling: A Knowledge-based Approach", *ICPTAT '95*, p 99-111.
- [We85a] D. de Werra. (1985). "An Introduction to Timetabling", *European Journal of Operational Research* 19, p 151-162.

### II. Buku

- [Go89] D.E. Goldberg. (1989). "Genetic Algorithms in Search, Optimization & Machine Learning", *Addison-Wesley*.
- [Ho75] J.H. Holland. (1975). "Adaptation in natural and artificial systems", *The University of Michigan Press*, Ann Brbor, Michigan.