Line 28

The results consist of many solutions. All solutions are optimal. It up to users that they will pick which solution to use. Furthermore, this proposed method consumes the cycle of process less than the linear programming solver.

Equation 1 บอกว่า DC + IC คือ Contract price ด้วย

Equation 4 บอกว่า Late penalty fee คำย่อคือ LPF ด้วย

Equation 7 9 10 11 เปลี่ยน I, t เป็นตัวห้อย subscript ด้วย

Line 306 แก้ chromosome เป็น solution แก้ chromosome ที่อื่นด้วย ให้เป็น solution ทั้งหมดทุกที่

คำอธิบาย figure 6 เพิ่มตัวห้อย i ที่ตัวแปรด้วย

เปลี่ยน shift day เป็น shifting day ทุกที่นะครับ

Ai is the activity i in the construction project.

Si is the number of shifting day from original plan {0,1, 2,...TF}

Xi is the binary decision variable to select the predecessor activities of activity i {0,1}

3.2 Genetic Operation

Line 332 แก้ rank-based mechanism เป็น uniform random mechanism

แก้ uniform crossover routine เป็น one-point crossover พร้อมแก้คำอธิบาย

Line 334 แก้ sapping the positions เป็น uniform randomization around the old value

แก้ swaps เป็น mutation

3.3.1 Pseudo code ไฟล์แนบ pseudo code.txt

3.3.2 Create population

In each solution, we randomize the value into the shifting time set (Set 1) and predecessor option set (Set 2). Set 1 was uniformly randomized from 0 to sigma parameter. And set2 was uniformly randomized from 0 to 1. the sigma parameter can be adjusted by the experiment. In this problem, this paper set the sigma parameter is 10.

3.3.3 Parent selection

This paper applies the uniform random technique to select 2 parents. This technique makes all parents have the same selection probability. After we selected both parents. These parents will be breeds by one-point crossover.

3.3.4 Crossover

Line 351 แก้ binary string เป็น solution value

3.3.5 N-point Mutation

This paper proposed the n-point mutation technique to reduce the complexity of computation. This technique follows the steps. First, we selected n decision variables in the set by uniform randomization. After that, we mutate all decision variables by uniform randomization around the old value. The uniform randomization has 2 bounds. the lower bound is old value minus sigma and the upper bound is old value plus sigma. If the mutated value has been out of possible value. Then this value was uniform randomize in possible value bound. In this problem, this paper set the possible value is between 0 to TF. if the shifting time is out of bound. Then we uniform randomize from 0 to sigma instead.

3.3.6 Fitness calculation

This problem has a challenge for the PDM network. Because the algorithm must create the new PDM Network every time from mutated decision variables sets before fitness scores calculation. It takes too much complexity of computation. And this problem has 3 objective functions to minimize. Hence, this paper applied the NSGA-II algorithm to solve the problem. Because this algorithm can solve the multi-objective problem using the fitness scores. The fitness scores can find non-dominated solutions. It was called the Pareto front.

3.3.7 Pareto front selection

The Pareto front is the solution which won at least 1 objective against with all solution. But some algorithm's generations have a few or too many the number of Pareto front. So, we need to keep the population size inbound. We defined the minimum and maximum bound of population size. If the number of Pareto front in population less than the minimum bound. Then we add some solutions which are not Pareto front until reaching the lower bound. Otherwise, If the number of Pareto front in population more than the maximum bound. Pareto front selection is based on the tournament of crowding distances, we use the crowding distance technique to reduce the number of Pareto front until reaching the upper bound.

4.2 Result comparison (แนบ paper เก่า)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Same Mx^2 score at | # Pareto front | Runtime |
| LP Solver | 10,000 generation | 1 solution | 8 hours |
| NSGA-II | 282 generation | 70 solutions | 4 days |

In the actual construction time objective (Ta) equals 720 days, this proposed model wins our previous paper[ref] at 282nd generation. In previous our paper, we applied the linear programming solver to find an optimal point. It takes 10,000 generations.