

論文紹介

多目的遺伝アルゴリズムによる IT プロジェクトスケジューリング

中市新太

富山県立大学

u020025@st.pu-toyama.ac.jp

October 20, 2023

はじめに

目的条件と制約
条件

条件設計

重複日数特定

全体のアルゴリ
ズム

数値実験

まとめ

はじめに

目的条件と制約
条件

条件設計

重複日数特定

全体のアルゴリ
ズム

数値実験

まとめ

背景

日経コンピュータの調査によると、IT 業界におけるプロジェクトの平均成功率は 31.1 %とされている。IT 人材不足が深刻な状況のため、スケジューリングを行うプロジェクトマネージャの負担は高まっている。

目的

プロジェクトマネジメントにおいてスケジューリング作業の効率向上によるプロジェクトマネージャの負担軽減が目的である。

先行研究

プロジェクトスケジューリングの問題のなかで、優先順位制約やリソース制約など資源の制限を課した問題を「資源制約付きプロジェクトスケジューリング問題」とよぶ。その中でも時間とコストを考えるものをDTCTPと呼ぶ。先行研究では品質、コスト、納期などを目的変数として採用しているが、ソフトウェアにおいては品質を図ることが容易ではない。

IT プロジェクトにおける条件定義

あらかじめ制約条件を入力すると、複数のパレート準最適解をデータとして出力し、ユーザがその中から選択できるものとする。IT プロジェクトにおいて品質は考えないと同時に、考えなければいけないことも増える (1) プロジェクトは知識労働集約型であり、コストのほとんどを人件費が占めるため、適切な要員配置をする必要がある (2) 要員ごとの能力差が大きいため、能力のある要員に複数のタスクを渡す場合がある。 (3) 各シフトの合計勤務人数と各グループからの人数に下限と上限を設定する。 (4) 労働負荷を考慮する。

目的条件

- (1) プロジェクト完了日数
 - (2) 全要因の重複タスクの日数の合計
 - (3) プロジェクトに参加する要員数
- これらの最小化を目的とする.

制約条件

- (1) 各タスクに先行タスクと日数がある
 - (2) 最初以外はすべてのタスクには先行タスクがある
 - (3) 先行タスクは複数設定できる
 - (4) タスクに割り当てる要因はリスト化されている
 - (5) 1 タスクに割り当てる要員数は 1 名である
 - (6) プロジェクトスケジュールは 1 つのタスクから始まり, 一つのタスクで終わる
- これらの最小化を目的とする.

制約条件の設計

はじめに

目的条件と制約
条件

条件設計

重複日数特定

全体のアルゴリ
ズム

数値実験

まとめ

$$(k_1, d_1, A_1, taskname_1) \quad (1)$$

$$(k_n, d_n, A_n, taskname_n) \quad (2)$$

k_i = タスク番号, d_i = タスクの所要日数

$A_n = (a_{i1}, a_{i2} \dots a_{in})$ (先行タスクの数)

$i = 1, 2, \dots n$ (n はすべてのタスク数)

要員リストは以下で表す

$$(e_1, membername_1) \quad (3)$$

$$(e_c, membername_c) \quad (4)$$

e_i = 要員番号, $i = 1, 2, \dots c$ (要員数)

遺伝子

$$(t_1, m_1), \dots, (t_n, m_n) \quad (5)$$

t_i = タスク開始遅延日数 (前タスク終了との)
 m_i = 要員番号 (要員リストのうちのいずれか一つ)
 $i = 1, 2, \dots, n$ (n はすべてのタスク数)
 $j = 1, 2, \dots, c$ (要員数)

はじめに

目的条件と制約
条件

条件設計

重複日数特定

全体のアルゴリ
ズム

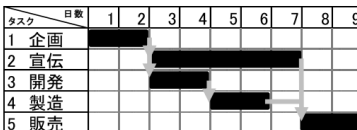
数値実験

まとめ

制約条件

- (1,2,(999999),"企画") (2,5,(2),"宣伝")
(3,5,(2),"開発") (4,2,(3),"製造") (5,2,(2,4),"販売")
- (1,"佐藤")(2,"鈴木")

ガントチャート



凡例

■ . . . タスク

→ . . . 先行制約

図 1: 制約条件を満たすガントチャート

最小化

はじめに

目的条件と制約
条件

条件設計

重複日数特定

全体のアルゴリ
ズム

数値実験

まとめ

$(0,1)(0,1)(0,1)(0,1)(0,1)$

タスク \ 日数	1	2	3	4	5	6	7	8	9
1 企画	佐藤								
2 宣伝			佐藤	佐藤	佐藤				
3 開発			佐藤						
4 製造					佐藤	佐藤			
5 販売								佐藤	

図 2: 納期, 要員数が最小

$(0,1)(0,1)(5,1)(0,1)(0,1)$

タスク \ 日数	1	2	3	4	5	6	7	8	9	10	11	12	13
1 企画	佐藤												
2 宣伝			佐藤	佐藤	佐藤	佐藤	佐藤						
3 開発			佐藤	佐藤	佐藤	佐藤	佐藤	佐藤					
4 製造									佐藤	佐藤			
5 販売												佐藤	

図 3: 要員数, 重複日数が最小

アルゴリズム

c : 要員数, n : 総タスク数, $lastDay$: 納期,

t_k : k 番目タスク開始遅延日数

m_k : k 番目タスク要員番号,

$taskStart_k$: タスク開始日

$taskEnd_k$: タスク終了日, ovd : 重複日数

1. $ovd = 0$
2. For $i = 0$ To c
3. $E[0..lastDay] = 0$
4. For $j = 0$ To n
5. If $m_j == i$ Then
6. For $l = taskStart_j + t_j$ To $taskEnd_j + 1$
7. $E[l] = E[l] + 1$
8. For $j = 0$ To $lastDay$
9. If $E[j] > i$ Then
10. $ovd = ovd + E[j] - 1$
11. Return ovd ;

重複日数特定の流れ

遺伝子が $(0,1),(0,1),(2,1),(0,2),(0,1)$ のとき

トポロジカルソートを用いて各タスクの開始終了日を求める。

それにより、全体日数が 10 日であることがわかる

トポロジカルソート：すべての要素について依存関係が一方方向に流れるように並べる

要員が 2 人であることを特定し、重複日数を数値化することができる

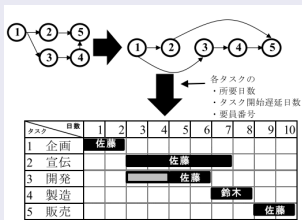


図 4: 流れ 1

タスク \ 日数	1	2	3	4	5	6	7	8	9	10
1 企画	佐藤									
2 宣伝				佐藤						
3 開発				佐藤						
4 製造							鈴木			
5 販売									佐藤	

佐藤	1	1	1	1	2	2	1	0	1	1
鈴木	0	0	0	0	0	0	1	1	0	0

(佐藤の 5 日目の重複タスク数-1)

+ (佐藤の 6 日目の重複タスク数-1) = 2 日

図 5: 流れ 2

はじめに

目的条件と制約
条件

条件設計

重複日数特定

全体のアルゴリ
ズム

数値実験

まとめ

アルゴリズム

- 多目的遺伝的アルゴリズムである
NSGA-II を用いる

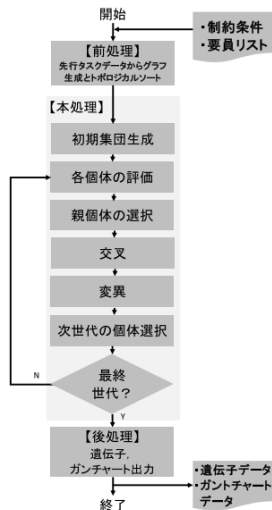


図 6: ソフトウェア全体の流れ

初期集団に厳密解を加える

12/14

3つの目的関数のうち2つを最小化したうえで、最後の1つをできるだけ最小化する厳密解を初期集団に加える

例の制約条件を使って説明する

納期

重複日数=0, 要員数=1 で最小化すると

$(0,1)(0,1)(5,1)(0,1)(0,1)$

納期は 13 日となる.

重複日数

納期=9, 要員数=1 で最小化すると

$(0,1)(0,1)(0,1)(0,1)(0,1)$

重複日数は 4 日となる

はじめに

目的条件と制約
条件

条件設計

重複日数特定

全体のアルゴリ
ズム

数値実験

まとめ

要員数

重複日数=0, 納期=9(上で求めた) とすると
 $(0, m_1)(0, m_2)(0, m_3)(0, m_4)(0, m_5)$
 重複がないように要員リスト順に入れていくと,
 $(0, 1)(0, 1)(0, 2)(0, 2)(0, 1)$
 要員数は 2 となる

$(0, 1)(0, 1)(0, 2)(0, 2)(0, 1)$

タスク \ 日数	1	2	3	4	5	6	7	8	9
1 企画	佐藤								
2 宣伝			佐藤	佐藤	佐藤	佐藤	佐藤		
3 開発			佐藤						
4 製造					佐藤				
5 販売								佐藤	

図 7: 納期, 重複日数が最小

数値実験

100 タスクの制約条件と 50 人の要員リストで数値実験を行う.

遺伝的アルゴリズムの世代数は 300

個体数は 200, 交叉率は 0.8(80 %), 突然変異率は 0.05(5 %)

Simulated Binary Crossover(交叉) と Polynomaial Mutation(突然変異)
の μ は 30.0 とした

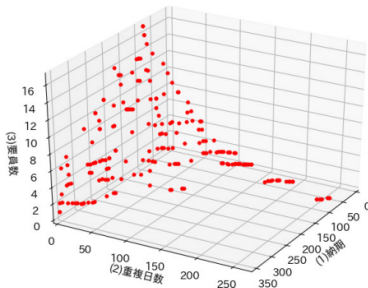


図 8: 数値実験

はじめに

目的条件と制約
条件

条件設計

重複日数特定

全体のアルゴリ
ズム

数値実験

まとめ

まとめ

自動スケジューリングソフトウェアを開発した。
厳密解を初期集団に含めることとしたところ、多様性を持った解集合を確保できた。
962 から 1000 タスクの数値実験を一時間以内で計算できた。
実務家へのインタビューの結果、活用可能であると評価をもらった

今後の展開

ほかのプロジェクトデータを用いた追加実験
タスクと要員のマッチング機能、タスク属性の付与
ガントチャートの特性を示す指標開発
計算時間短縮