

# 動的スケジューリングによるタスク特性とポリシの変動への対応

Dynamic Scheduling Addressing Variation of Task Characteristics and Policy

佐藤 慎也†

Shinya Sato

島川 博光†

Hiromitsu Shimakawa

## 1. はじめに

オペレーティングシステムにおけるタスクのスケジューリングや生産工程のスケジューリングなど、スケジューリング技術は幅広い分野で用いられている。一般的なスケジューリングでは、タスク特性である各タスクの実行時間や優先度は変化しないものとして扱われるが、実行環境の変化などにより変動することがある。これらの変動により、処理前は最適であったスケジュールが処理中に効率が悪いスケジュールになってしまうこともある。

例えば、トンネル内の大気汚染濃度およびトンネル内風速の許容制御基準値の維持を、最小の電力量で実現させることを目的としたトンネル換気制御のリアルタイム・シミュレーションでは、各車の動きから気流を計算するため、渋滞などにより制御タスクの計算時間が変動する。もし、制約時間内に全てのタスクが終了しない場合は、精度を落としてシミュレーションすることもある。この他に、交通渋滞による移動時間の変動も含めた観光スポットの訪問順のスケジューリング[1]や、待ち時間を考慮したテーマパークにおけるアトラクションの巡回順序のスケジューリングも同様の問題として扱われる。

また、最近では携帯可能な通信機器が普及している。スケジューリング結果をのようなモバイル機器で表示することはユーザにとって有益なことである[2][3]。

本稿では、実行時間と優先度の変動が制御の問題より変化に富み、モバイル端末を用いることが必須となるタスクのスケジューリング方法の例として、制御の問題より変化に富む、テーマパークのアトラクションの巡回を対象問題と考え、そのスケジューリングについて考える。

## 2. 実行時間と優先度を意識したスケジューリング

### 2.1 変動に応じた近視眼的スケジューリング

タスクの実行時間が頻繁に変動するスケジューリングでは、いつでも最適なスケジュールを提供するために、その時点での正確な実行時間が取得できなければならない。また、タスクの実行時間の変動に伴いスケジュールも変動するため、スケジューリングのさいの計算コストを抑える必要がある。そのため、状態のオンライン収集と計算コストを抑えたスケジューリングを組み合わせたスケジューリング方式をとる必要がある。

実行時間などの状態をオンラインで収集することで、リアルタイム・データを取得・使用できるため、実行時間の変動に対応することができる。また、Spring のスケジューリング・アルゴリズム[4]のような、近視眼的なスケジューリング方式を採用することで、頻繁にスケジューリングしても計算コストは少なくて済む。

近視眼的なスケジューリングで計算コストを抑えているため、ユーザの気分や予定の変化によってスケジューリング・ポリシを変えて再スケジューリングすることも可能となる。

### 2.2 タスクの実行時間と優先度

優先度とは、その順でタスクをスケジュールするタスク毎の順位である。優先度は扱う問題によって異なる。例えば、到着順 (FCFS : First Come First Service) スケジューリング[5]では、実行可能キューに到着したタスクを順にスケジュールするため、到着順で優先度が決まる。

しかし、実問題では優先度の尺度が変動することがある。本方式で扱うテーマパークにおけるアトラクションのスケジューリング問題では、タスクの優先度をユーザが決めるアトラクションの順位と許容時間で表す。許容時間とは各タスクの実行に許される最大時間であり、この時間を越えるタスクはスケジュールされない。順位と許容時間はユーザの主觀によって決まるため、気分によって変更されることもある。

タスクの実行時間は、アトラクションの待ち時間とアトラクションの体験時間の合計なので、待ち時間によって変動する。このタスクを単に順位の高いものからスケジュールすると、待ち時間の長いタスクのみがスケジュールされることがある。これは、一般的に個人が指定した順位がそのアトラクションの人気の高さに比例するため、人気のある(順位の高い)アトラクションはそれだけ待ち時間が長くなるからである。しかし、テーマパークの訪問者は、アトラクションへの興味と待ち時間の長さを天秤にかけて体験するかを判断する。そのため、許容時間というタスクの実行に許される最大時間を用いる。

### 2.3 スケジューリング方法

本方式では、優先度順にタスクをスケジューリングする方法をとる。優先度はユーザが決める順位と許容時間で用いる。全てのタスクを順位の高い順に並べ替え、順に現在の実行時間が許容時間内であるかを判定する。

また、本方式ではタスクを全体  $n$  個中の近視眼的に  $k$  個分をスケジュールする ( $k \leq n$ )。タスクの許容時間を  $A$ 、現在の実行時間を  $E$  として、以下にスケジューリング手順を示す。

- ・優先順位の高いタスクを順に並べる。これを  $I$  とする。
- ・ $I$  の先頭よりひとつずつ要素  $e_i$  を取り出す。
- ・ $e_i$  ( $i=0, \dots, n-1$ ) に対して  $A(e_i) \geq E(e_i)$  ならばスケジュールする。
- ・スケジュールされたタスクが  $k$  を超えた場合、処理は終了となる。

このような流れでスケジューリングは行われる。また、ユーザが入力した許容時間や順位、スケジューリング・ポリシの変更があった場合、変更後の状態を用いて再スケジューリングを行う。

† 立命館大学大学院 理工学研究科

## 2.4 スケジューリング・ポリシの変更

常にユーザの希望に合ったスケジュールを提供するためには、スケジューリング・ポリシを変更することが必要な場合もある。

本方式では、ユーザの入力によりスケジューリング・ポリシを変更する2種類の方法を用いる。

### 2.4.1 実行するタスク数を増やす場合

ユーザが提示されたスケジュールに対して、待ち時間が長すぎて限られたアトラクションしか体験できないという不満を持つことが考えられる。本方式では、順位と時間を考慮したスケジューリング方法を採用しているため、順位を意識しすぎると時間当たりのタスク数が少なくなることがある。一般的に、順位が上位のタスクは人気のあるタスクであるため、上位のタスクが多くスケジュールされたときにこのような問題は起こる。

ユーザが現在のスケジュールよりタスク数を増やしたいと考えた場合、処理時間の短いタスクがスケジュールされるように、実行時間が長いタスクの許容時間を短くし、実行時間が短いタスクの許容時間を長くする実行時間の逆数を重視した許容時間の設定方法を取る。

以下に、優先順位が1から15の場合のポリシの変更例を示す。優先順位リストを、優先順位を元に5分割して1から3を $g_1$ 、4から6を $g_2$ 、7から9を $g_3$ 、10から12を $g_4$ 、13から15を $g_5$ とする。この分割数は対象とする問題によって異なると考える。なお、 $\alpha$ および $\beta$ は定数であり、 $\alpha > \beta > 0$ である。

スケジュールされるタスク数を増やすようにポリシを変更させるとときには要素 $e_i$ について、

$$\begin{aligned} A(e_i) &= A(e_i) - \alpha & (e_i \in g_1) \\ A(e_i) &= A(e_i) - \beta & (e_i \in g_2) \\ A(e_i) &= A(e_i) + \beta & (e_i \in g_4) \\ A(e_i) &= A(e_i) + \alpha & (e_i \in g_5) \end{aligned}$$

これにより、一般的に実行時間が長い順位が上位であるタスク群 $g_1$ および $g_2$ に含まれるタスクは、許容時間が減らされたためスケジューリングされにくくなる。一方、順位が下位のタスクは実行時間が短いため、許容時間を増やしスケジュールされやすくなる。このような操作により、時間当たりのタスク数を増やす。

### 2.4.2 優先度をより重視する場合

ユーザが決めた許容時間が短かった場合、順位の低いタスクが多くスケジュールされることがある。ところが、ユーザは待ち時間が長くても順位の高いアトラクションを体験したいと考えることもある。

一般に、順位の高いタスクほど実行にかかる時間は長いため、順位の高いタスクは許容時間が短ければスケジュールされない可能性が高くなる。全体のタスク数は減らしても、順位が上位のタスクを多くスケジュールしたいといった、より強く優先度を意識する場合、許容時間を延ばすことで解決する。

以下に、2.4.1と同様に優先順位が1から15の場合のポリシの変更例を示す。優先順位リストを、優先順位を元に3分割して1から5を $h_1$ 、6から10を $h_2$ 、11から15を $h_3$ とする。なお、 $\gamma$ 、 $\delta$ 、 $\epsilon$ は定数であり、 $\gamma > \delta > \epsilon > 0$ である。

要素 $e_i$ について、以下のように許容時間を変更する。

$$A(e_i) = A(e_i) + \gamma \quad (e_i \in h_1)$$

$$A(e_i) = A(e_i) + \delta \quad (e_i \in h_2)$$

$$A(e_i) = A(e_i) + \epsilon \quad (e_i \in h_3)$$

これにより、許容時間が増えよりスケジュールされやすくなる。よって、より強く優先度を意識したスケジュールが提供できるようになる。

## 3. 実装

実際にスケジュールの生成や提示を行うために、テーマパークにおけるアトラクションのスケジューリング用のプロトタイプシステムを実装した。

### 3.1 利用形態

ユーザは、市販されている携帯可能なPDA、パソコン・コンピュータを用いる。タスクの順位や許容時間などのパラメータは、訪問前に決定できるため、自宅のパソコン・コンピュータを用いて事前に入力できる。テーマパーク内では歩きながら操作ができる小型の通信機器が必要となるため、ネットワーク接続可能なPDAを利用する。

### 3.2 データ構造

本手法では、ユーザの情報、ユーザ毎のアトラクション情報およびアトラクションの属性を用いてスケジューリングする(図1)。これらは、データベースに格納され、スケジューリング時に取得される。また、これらのデータは変更があるたびに更新する。

ユーザの情報はIDやパスワードなどの登録内容や、デッドラインまでの時間を計算するための入園時間と退園時間を持つ。ユーザ毎のアトラクション情報は、アトラクションの順位と許容時間を持つ。アトラクションの属性は、各アトラクションの現在の待ち時間を持つ。

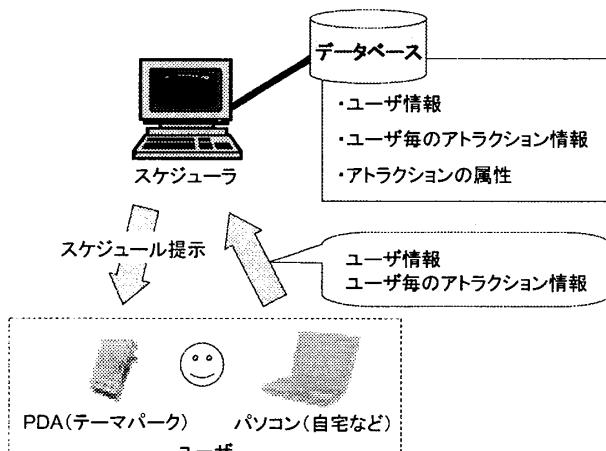


図1：スケジューリングに必要な情報

### 3.3 システム動作

実装したグラフィカル・ユーザ・インターフェース(GUI)を用いてスケジューラの動作を、事前入力部とテーマパーク内の行動中のスケジュール提示部に分けて説明する。

#### 3.3.1 事前入力部

ユーザはシステムにログイン後、アトラクション情報を入力する。まず、エリアを選択する。図2において線で区切られた7つのエリアのうち、1つを選択する。

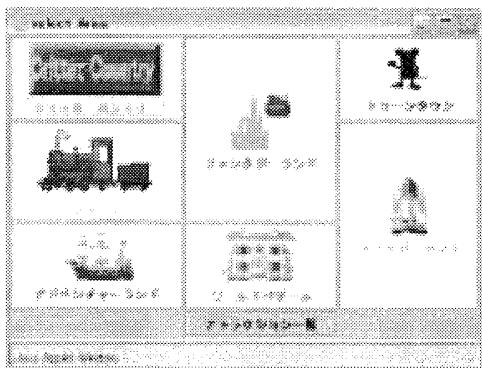


図2：エリア選択画面

選択したエリア内のアトラクションのリストが表示されるのでこの中から体験したいものを選び、それぞれの順位と許容時間を決定する。

### 3.3.2 スケジュールの提示部

図3では、近視眼的に3個分のアトラクションをスケジュールした場合を示す。実行順序は簡単なガント・チャートで示される。

提示されたスケジュールを変更したい場合、スケジューリング・ポリシの変更を選択するか、入力したアトラクションの許容時間または順位を変更し再スケジューリングを行う。

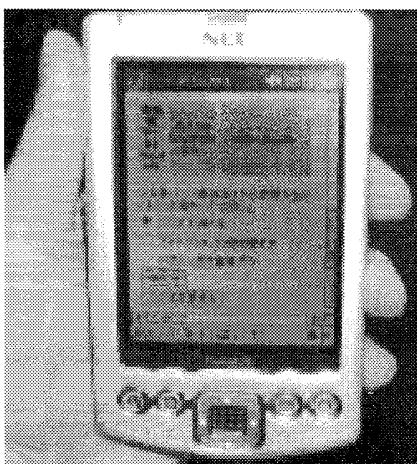


図3：PDAによるスケジュールの表示

## 4. 評価

実装したスケジューラと、既存のスケジューリング・システムの比較を行う。比較対象は、市内の観光案内を行う

Lancaster ガイド・プロジェクト（以後、Lancaster）[1]と、テーマパークでのスケジューリングを行う web 上のシステム[6]とする。評価の一覧を表1に示す。

本方式は他のシステムに比べて、目的地の順位付け、スケジューリング・ポリシの変更、許容時間の変動に対応しているため、よりユーザの好みや気分を反映したスケジュールを提案できる。また、Lancaster で問題とされていた、入力の煩わしさは、事前入力とグラフィカルで退屈しない入力画面の提供により解決できると考える。

## 5. 終わりに

本稿では、タスクの実行時間、優先度が変動する場合の動的スケジューリング方式を、テーマパークにおけるアトラクションの巡回順序のスケジューリングの例で示した。タスク数を増やす、優先度をより重視するといったスケジューリング・ポリシの変更によりユーザの気分を反映したスケジュールを提供可能とした。

今後は、異なる問題分野のスケジューリングにも応用することを検討する。また、テーマパークの訪問者であるユーザの意向だけではなく、テーマパークの主催者側の意向も含めたスケジューリングを行うことを検討する。

## 参考文献

- [1] Davies, N., Cheverst, K., Mitchell, K. and Efrat, A., "Using and Determining Location in a Context-Sensitive Tour Guide", IEEE Computer, Vol.34, No.8, pp.35-41 (2001).
- [2] Miguel A. Muñoz, Marcela Rodríguez, Jesus Favela, Ana I. Martínez-García, and Victor M. González, "Context-Aware Mobile Communication in Hospitals", IEEE Computer, Vol.36, No.9, pp.38-46(2003).
- [3] Yasuto Nakanishi, Kazunari Takahashi, Takayuki Tsuji, and Katsuya Hakozaiki, "iCAMS: A Mobile Communication Tool Using Location and Schedule Information", IEEE pervasive computing, Vol.3, No.1, pp.82-88(2004).
- [4] Krithi Ramamritham, John A. Stankovic, and Perng-Fei Shiah, "Efficient Scheduling Algorithms for Real-Time Multiprocessor Systems", IEEE Transactions on Parallel and Distributed Systems, Vol.1, No.2, pp.184-194(1990).
- [5] 大久保英嗣, "オペレーティングシステムの基礎", サイエンス社, (1997).
- [6] WEB コンシェルジェ/USJ,  
<https://member.usj.co.jp/svw/uwc?method=displayParkActionPlanTop>

表1：システム評価

評価内容	本方式	Lancaster	web
事前に必要事項の入力が可能	○	×	○
ユーザの好みを反映	○	○	○
目的地毎に行きたさの順位付けが可能	○	×	×
移動距離を考慮	×	○	×
スケジューリング・ポリシの変更が可能	○	×	×
目的地での所要時間の変動に対応	○	○	×

