

学士論文

人物センシング情報を考慮した

エレベータかご割り当てに関するシミュレーション解析

Simulation analysis of the elevator allocation considering sensing
information

指導教員 中出 康一 教授 孫 晶 助教

名古屋工業大学

工学部 都市社会工学科

経営システム系プログラム

平成 26 年度入学

(学籍番号) 26117068

(氏名) 舟橋弘喜

(提出日) 平成 30 年 2 月 9 日

目次

第1章 序論

第2章 人物センシング情報を用いたエレベータ群管理

- 2.1 従来研究
- 2.2 人物センシング情報を用いたエレベータ制御モデル
- 2.3 課題

第3章 シミュレーションの概要

- 3.1 シミュレーションソフト(S4 Simulation System)
- 3.2 エレベータモデル
- 3.3 乗客エージェントのモデル
- 3.4 行先と乗車階の設定
- 3.5 イレギュラーな乗客エージェントの発生アルゴリズム
- 3.6 人物センシング情報を用いたかご割当てアルゴリズム
- 3.7 出力結果例

第4章 シミュレーション実験

- 4.1 シミュレーションモデルの設定
- 4.2 想定環境
- 4.3 一般的な時間のシミュレーション結果比較
- 4.4 時間帯別(朝, 夕)によるシミュレーション結果比較

第5章 結論

参考文献

第1章

序論

近年、都市化の発展に伴い、居住人口密度が増加し、高層建築物が年々増えてきている。これらの高層建築物の増加は世界的に見てもいえることであり、それによるエレベータにおける建物内の縦方向の移動はより重要視されている。大規模な高層建築物においては、複数台のエレベータのかご(以下、単に“かご”と呼ぶ)が設置されており、2~6台程度が一つの群として運用されている。エレベータ群管理システムは、複数のかごを群として扱い、効率よく運行スケジューリングすることにより、エレベータ群の輸送の能力を最大化し、乗り場で待つ乗客の平均待ち時間や、行先階までのサービス完了時間を最小化することを目的としている。これらの評価指標の改善を目的として、匹田ら[1]では人工知能技術などの導入により複数台のエレベータの配車を行うエレベータ群管理システムの効率化が図られてきた。また、近年、RFID・バイオメトリクスなどによる認証技術の進歩やセキュリティ面の社会の関心が高まり、監視カメラ・入退管理・セキュリティゲート等の人物センシング情報を利用したシステムが広く普及している。岩田ら[2]では、ビル内に設置されたセキュリティ機器による人物センシング情報を用いて、事前に個々の利用者に割当かごを報知するエレベータ群管理システムを提案することで、運行効率を向上させることができた。

エレベータ群管理の研究論文の本質的な目的は、乗客の平均待ち時間やサービス完了時間の最小化や運転効率の向上にある。しかし、ほとんどの研究では、総走行距離や電力コストなどの評価指標を改善する目的で、結果運転効率を上げることに繋がるもの(小堀ら[3])や新たなエレベータ群管理システムを提案することで、改善するものがほとんどであり、ビル内の入退管理・監視システム・セキュリティゲート等のセキュリティ面を考慮した研究はあまりない。

[2]で人物センシング情報を用いたエレベータ群管理を提案することで、セキュリティ面での考慮がなされ、かつ運転効率の向上が図れたことから、本論文ではセンシング情報を用いたエレベータ群管理について考えていく。しかし、[2]では乗客が割当てられたかごに必ず乗車し、事前に登録された行先階に向かうことが前提となっており、日常的な場面を考慮するならば、乗客が必ずしも割当られたかごに乗車しない場合や事前に登録された行先階に向かわない場合等のイレギュラーなケースを考える必要がある。そこで、シミュレーション上で上記のようなイレギュラーなケースをある一定の確率で発生させ、サービス完了時間や平均待ち時間の比較を行う。また、乗客の発生間隔や行先階の分布を変化させながら、人物センシング情報を用いたかご割当に関して、どれほどの影響があるのか調べる。

本論文の構成は以下のとおりである。2章では人物センシング情報を用いたエレベータ群管理について説明する。3章では、本研究で扱うシミュレーション上でのエレベータや乗客のモデルやアルゴリズムについて説明する。4章では、各パラメータについて設定していきながらシミュレーションを行い、それぞれについてサービス完了時間・平均待ち時間の結果を示す。5章では結論を示す。

第2章

人物センシング情報を用いたエレベータ群管理

2.1 従来研究

エレベータ群管理は不確実な乗客発生情報に基づいた複雑な最適化問題であるため、人工知能技術などを用いて「知識」を導入し、より少ない演算量でより良い制御を行うアプローチが試みられている。

エレベータ群管理のAI技術応用の最初の研究として匹田ら[4]、Amanoら[5]のファジィルールを適用した手法があげられる。この研究では、専門家の記述したファジィルールを用いて、エレベータ群管理の性能向上を図っている。また、海外を中心にエレベータホールで行先階を登録して個々の利用者に割り当てかごを報知する方式(DCS: Destination Control System)が普及しつつある。このDCS方式では、先に登録された行先階情報を利用して停止階床数を削減するエレベータ群管理方式が採用されている。出勤時運転において停止する階床数の上限値を適切に選択することにより、運転効率の向上を図った方式に関する研究(岩田ら[6])があげられる。最近、画像処理・RFIDなどの人物センシング情報を用いたエレベータシステム及びアルゴリズムの研究が行われている。RFIDのID情報とステレオカメラから得られる乗客の動線情報を用いたエレベータシステムに関する研究(Kim & Moon[7])や、RFIDによる自動呼び登録に対応したエレベータ群管理に関する研究(Strong & Bauer[8])がある。また、[2]のような複数台から割当かごを決定する手法を用いて、事前に個々の利用者に割当かごを報知するエレベータシステムを提案している研究もある。

2.2 人物センシング情報を用いたエレベータ制御モデル

従来は乗客がエレベータホールのボタンを操作して初めてエレベータシステム側は乗客発生を検出していたため、運転制御に必要な将来までの乗客発生状況を利用することができなかった。一方、人物センシング情報を用いた場合、エレベータホール到着前に個々の乗客の発生を検出することが可能となる。また、エレベータホール到着時刻・行先階を検出し、検出時に各乗客に割当かごの番号を報知する。エレベータは割当済の乗客がエレベータに到着する時刻まで戸開して、乗り遅れをなくす。実用的なエレベータの割当かごの報知システムとして、図1に操作フローを示し、以下の①～④に代表的な操作シーケンスを記す。

- ① 各乗客がエレベータホールから一定距離離れた場所に設置されたセキュリティゲー

トで ID 認証する。ID カード認証時に乗客 ID と事前登録された情報に基づいて行先階を自動登録する。

- ② エレベータ群管理システムは乗車階および行先階に基づいて割当かごを決定し、セキュリティゲート前方の表示器に割当かごの番号(例えば「A」)と表示する。
- ③ 乗客はエレベータホールに移動し、表示された割当号機番号の前で待機する。
- ④ 割当かごが到着すると、乗客は乗車して登録された行先階まで移動する。

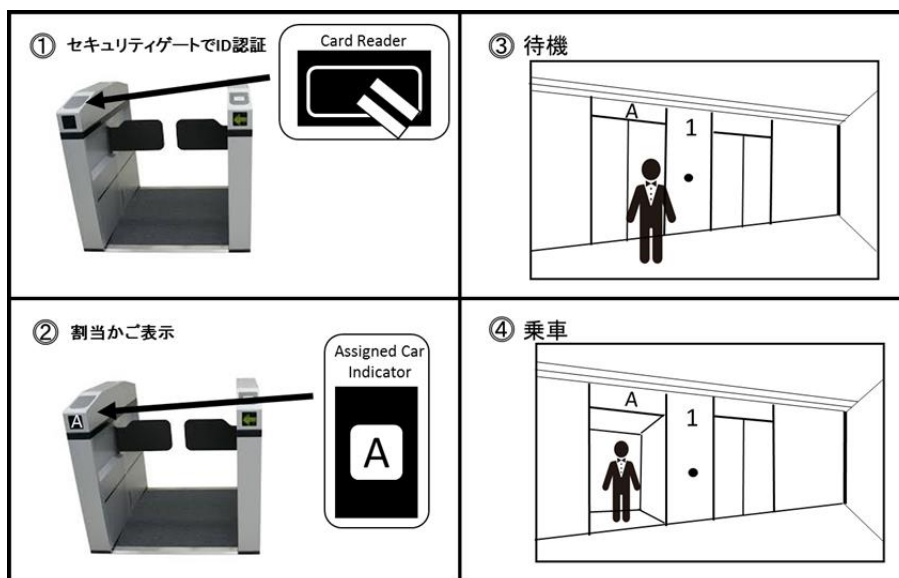


図 1

2.3 課題

[2]で述べているように、一般にエレベータ運転効率はロビー階を出発して上方階にサービス後再びロビー階に戻ってくるまでの周回時間が短く、出発時の乗車人数が多いほど単位時間当たりの輸送人数は大きくなる。一方、周回時間が長く、乗車人数も少ない場合、単位時間当たりの輸送人数は小さくなり、出発時の積み残しが発生し、乗客の待ち時間は悪化する。

人物センシング情報を用いたエレベータシステムでは、乗客をエレベータホール到着前に検知可能なことから、乗客発生間隔が疎の場合、[7]・[8]が示すように事前にエレベータを配車することで、乗客の待ち時間は短縮することができる。しかし、乗客発生間隔が密の場合、単に事前に配車するだけでは運転効率は向上しない。この場合における課題を図 2(a), (b)に図示して説明する。

図 2(a), (b)のように、センシング位置からエレベータホールまで多くの乗客が移動中であり、混雑時には乗客の積み残しが発生しないように単位時間当たりの輸送人数を大きくすることが重要である。

図2(a)は、検出した乗客に対して最初の乗客と同じかごを割り当てることにより、歩行移動区間を移動中の全乗客を同一のかごAに割り当てるケースである。このケースでは、1周回あたりの輸送人数は上がるが戸開して待機する時間が増えるため、周回時間が長くなり単位時間当たりの輸送人数が小さくなる。一方、図2(b)では、歩行移動区間を移動中の乗客を複数のかごに分散して割り当てるケースである。このケースでは、乗客を複数のかごに分散して割り当てることで、各階で戸開して待機する時間が小さくなる。これにより1周回時間は短縮されるが乗車人数が少なくなる為、単位時間当たりの輸送人数は少ない。その他にもエレベータが乗客より先に乗車階に到着するケースも考えられる。その場合かごが待機する形となり、混雑時には無駄な待機時間となってしまう、運転効率の悪化につながる。

よって、人物センシング情報を用いたエレベータ群管理では、周回時間の短縮および乗車人数の確保、かご内の待機時間の削減等、個々の乗客をセンシングした時点の情報に基づく割当かご決定が解決すべき課題となる。

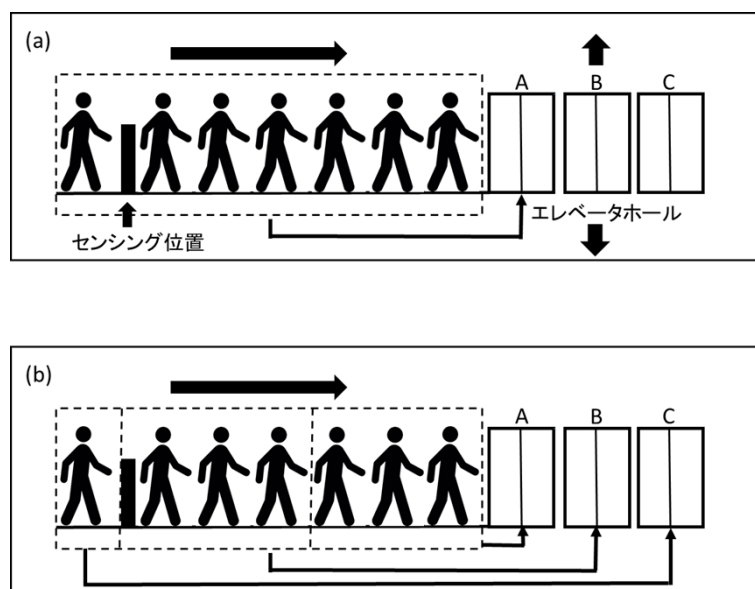


図 2

第3章

シミュレーションの概要

3.1 シミュレーションソフト(S4 Simulation System)

本研究では、人物センシング情報を用いて、乗客が発生してからの、かごの割当、エレベータホールまでの移動、乗車から降車までのエレベータの制御を行う。計算は、S4 Simulation System (株式会社 NTT データ数理システム) を用いて離散時間でシミュレーションする。S4 Simulation System は、一定のルールに従って行動するエージェントの振る舞いをシミュレーションし、エージェント同士の相互作用から現れる複雑な社会現象を分析・予測する。S4 Simulation System を使用して、エージェントシミュレーションをモデル化する際、(本ソフト独自の)psim 言語ライブラリを用いた Python 言語によるプログラミングを利用する。以下、実際にシミュレーションで用いる変数名・配列名を用いてモデルの概要を説明する。

3.2 エレベータモデル

エレベータは、個々が検出された時点の情報に基づいて乗車階へと移動し、乗客を乗せる。基本的な動きとして、例えばエレベータが上向きの場合、乗車階よりも上方を行先階とする乗客しか乗せず、エレベータが下向きの場合、その逆である。乗車の際に、乗車時刻・エージェント ID・エレベータ ID・ホール待ち時間・行先階・乗車階のデータを乗客に与える。そのデータを基にエレベータは行先階へと移動し、乗客を降車させる。エレベータが乗客待ちのある階に移動する際、乗客待ちのある階のリスト(`e.waitingPassenger`)にある階から順に移動する。乗客待ち階に移動後、乗客を乗せ、各階の乗客待ちのリスト(`e.waitingAgentAry[self.floorNum]`)から待っていた乗客を削除する。その際、`e.waitingPassenger` も削除するが、そのかごに割当られている乗客が他にも存在する場合、削除しない。

3.3 乗客エージェントのモデル

人物センシング情報を利用したエレベータ群管理シミュレーション環境の主要機能の詳細を以下の(a)~(e)に示す([2]を参考)。

- (a) 個々の乗客エージェントはエレベータホールから $x[m]$ 離れた位置に発生する。

- (b) 乗客エージェントがエレベータホールから $x[m]$ 離れた位置において、自動的に乗場行先階呼びを登録し、乗客エージェントは割当かご番号をエレベータシステムから受け取る。
- (c) 乗客エージェントは、エレベータホールまで速度 $v[\%]$ で移動する。
- (d) エレベータホール到着後、乗客エージェントに割り当てられたエレベータが乗車方向で到着すると、戸開完了後にかごに順次乗車する(確率 $\text{prob } a$ で発生したイレギュラーな乗客は、乗車後、行先階を変更)。このとき、エレベータは割り当てられた全ての乗客エージェントが乗車するまで戸開待機する。
- (e) 乗客エージェントは、乗車したかごが行先階に到着すると戸開完了後に降車する。

3.4 行先と乗客到着の設定

・行先階の設定

乗車階毎の行先階の設定は図 3 の重み行列(ws)の値を変更して行う。図 3 は行が乗車階、列が行先階となっており、1階で発生した乗客は2~16階に上がる行先階分布は1と、一定になっている。2階で発生した乗客の場合、1階に下がる重みが3、3~16階に上がる重みを1とし、乗客が乗車階より下がる移動に関しては上がる移動よりも重みの値を大きくすることで、ロビー階(1F)へ下がる移動を多くする。

行先の設定

```
ws = [[ 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1], # 1F のユーザ
      [ 3, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1], # 2F のユーザ
      [ 3, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1], # 3F のユーザ
      [ 3, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1], # 4F のユーザ
      [ 3, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1], # 5F のユーザ
      [ 3, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1], # 6F のユーザ
      [ 3, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1], # 7F のユーザ
      [ 3, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1], # 8F のユーザ
      [ 3, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1], # 9F のユーザ
      [ 3, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1], # 10F のユーザ
      [ 3, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1], # 11F のユーザ
      [ 3, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1], # 12F のユーザ
      [ 3, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1], # 13F のユーザ
      [ 3, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1], # 14F のユーザ
      [ 3, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1], # 15F のユーザ
      [ 3, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0] # 16F のユーザ
    ]
```

図 3

・到着間隔の設定

乗車階毎の乗客の到着(発生)間隔の設定は図 4 の `arriveMean` のリストの値を変

更して行う。このパラメータは、時間帯別やフロア別に乗客の発生間隔の期待値を表している。図 4 では、0~MaxTime までの各階の乗客の発生間隔は指数分布に従う。リストを増やしていくことで、より細かい時間帯(朝, 昼, 夕)に別けて各階の乗客を発生させることが可能になる。

各階の乗客の発生間隔の期待値(指数分布)を表す。
左から1階は2、2~16階は30

到着間隔
arriveMean = [(mt, [2,30,30,30,30,30,30,30,30,30,30,30,30,30,30]), # 0~mt秒までの間隔
(maxTime, [2,30,30,30,30,30,30,30,30,30,30,30,30,30,30])] #mt+1~ maxTimeまでの間隔

図 4

3.5 イレギュラーな乗客エージェントの発生アルゴリズム

本研究では、日常的な場面を考慮したセンシング情報を用いたかご割当を考える為、ある一定の確率 **Prob a** でイレギュラーな動きをする乗客エージェントを発生させる。イレギュラーな動きとして、乗客エージェントがセンシング情報で割当てられたかごに乗車後、エレベータ内のボタンで本来とは異なる行先階を選択し、移動するものである。4章で、確率 **Prob a** を変化させていき、シミュレーションを行い、人物センシング情報を用いたかご割当てにどれほどの影響があるのか調べていく。

以下に、イレギュラーな乗客を発生させるアルゴリズムを示す。シミュレーション上では、乗客エージェントが作成された後、各エージェントについて行先階が決定される。この行先階の決定後、確率 **Prob a** で本来の行先階とは異なる行先階のデータを乗客に与える必要がある。

1. 関数 `calcGenerateNum` で、到着間隔のリスト `arriveMean[]` に従い各時刻の乗客エージェントを生成する。
2. 関数 `environmentBase.initAttribute` を呼び出し後、関数 `initAfter` (各乗客エージェントに対して行先階を決定する) に入る。
3. `n` について 0 から `maxFloor` の範囲で、確率 $\frac{ws[i][n]}{\sum ws[i][n]}$ で `n` を返すような乱数ジェネレータを `g` とする。`ws[]` は行先階の設定の為の重み行列であり、`ws[i][n]` は目的階 `n` についての重みである(`i`: 乗客エージェントの乗車階)。
4. 乱数ジェネレータ `g` を `self.goal` に返し、センシング直後の行先階が決定する。

5. 0.0 から 1.0 までの一様分布に従う乱数ジェネレータを b とし, b を z に返す.
6. z が $\text{Prob } a$ よりも大きい場合, self.goal1 (実際の行先階) を self.goal (センシング直後の行先階) とする. その他の場合, self.goal1 と self.goal が異なる値になるまで, 乱数ジェネレータ g を self.goal1 に返す.

これにより, self.goal (センシング直後の行先階) と self.goal1 (実際の行先階) が作成できる. イレギュラーな乗客は, この値が異なるデータをもつエージェントである. また, $\text{Prob } a$ の値の変化の影響を調べるために, かが割当てのアルゴリズムでは self.goal をもとに決定された値を使用する.

3.6 人物センシング情報を用いたかが割当てのアルゴリズム

[2]を基に, 人物センシング情報を用いた割当かが決定に関する提案手法の方針を①～③に示す.

- ① 乗車階の戸開待機時間・乗車時刻間隔・かが待機時間を評価し, 周回時間の削減・単位走行あたりの乗車人数の確保・かが内の待機時間削減の観点から割当候補かごを選定する.
- ② 各かごの停止回数を制限することにより, 周回時間の削減を図る.
- ③ ①かつ②の条件を満たした割当候補かごに対して, 乗客のセンシング位置からエレベータホールまでの移動時間を考慮した評価指標に基づき, 個々の乗客の待ち時間が小さくなるように割当かごを決定する.

以下に, 人物センシング情報を用いたかが割当てのアルゴリズムを示す.

- M : 割当候補かご集合(0,1,2,3)
- $i(\text{self.floorNum})$: 乗り場
- $j(\text{self.goal})$: センシング直後の行先階
- y : 行先階
- ta : エレベータの現在位置と停止予定階か予測したかごの i 階の到着時間と乗客のエレベータホール到着時間との差
- tci : $i(\text{self.floorNum})$ 階までのかごの到着予測時間
- stopFloor : エレベータの停止回数
- TH : センシング位置からエレベータホールまでの距離
- $\text{beta}(\beta)$: $TW0$ と同じ値(定員の 75%が乗り込む時間= $20*0.75$)

- `tw`: 戸開待機時間
- `tw0`: エレベータ定員の 75%が乗り込む時間
- `tw1`: `tw0` に許容時間を加算した値
- `tint`: 乗車予定時刻間隔
- `tint0`: エレベータ定員の 50%が乗り込む時間
- `tint1`: `tint0` の 1/2 の値
- `S`: 停止回数上限値
- `s`: 停止回数上限値の初期値

1. `eNum = self.simulator.aSynchronousAgentSet.n` とする.
2. 割当候補かご集合 $M(0,1,2,3)$ のリスト `MM` の各かご `[i]` について, 関数 `getArriveTime` から引数 `(tci, stopFloor)` を呼び出す. かごの到着時間と乗客のエレベータホール到着時間との差が定員の 75%が乗車する時間よりも長い場合 (`tci - TH + beta >= 0`), そのかご `[i]` について割当候補かご集合から除外する. もし, すべてのかごが除外対象の場合, すべてのかごについて除外しない.
3. 割当候補かご集合 M のリスト `MM` の各かご `[i]` について, 以下の条件①, 又は②のどちらかを満たさないかごを割当候補かご集合から除外する. もし, すべてのかごが除外対象の場合, すべてのかごについて除外しない.
 - ① 戸開待機時間より定員の 75%が乗車する時間の方が長い, かつ乗車予定時刻間隔より定員の 50%が乗車する時間が長い.
(`tw <= tw0 and tint <= tint0`)
 - ② 戸開待機時間より定員の 75%が乗車する時間に許容時間を加算した値の方が長い, かつ乗車予定時刻間隔より定員の 50%が乗車する時間の 1/2 の値が長い.
(`tw <= tw1 and tint <= tint1`)

①・②について (`tw0 < tw1 and tint0 > tint1`) である.
4. 停止回数上限値の初期値 `s` を `s = self.agentset.env.maxFloor / eNum + 2` と定める. 停止回数上限値 `S` の緩和回数を 100 と設定し, 割当候補かご集合 M のリスト `MM` の各かご `[i]` について, `ta` の値が負の場合, かがが先に到着したした際の割引を考慮した係数として $\alpha=0.5$ を適用する. 停止回数上限値 `S` 以下のかごがある場合, `ta` の二乗した値を `v[i]` のリストに入れる. もし `v[]` のリストに数値がある場合, `v[]` の中の最小値を割当かご `c` と決定し, 割当かご `c` を返す. ない場合, 停止回数上限値 `S` を 1 増やして, 4 の初めから決定するまで繰り返す.

3.7 出力結果例

シミュレーション実験の評価における出力指標としてエレベータ群管理の主要な評価項目である以下の2つの指標を用いる。

- 平均待ち時間 (Average Waiting Time) :
乗客がエレベータホールに到着後かごに乗車するまでの時間の平均値。エレベータホールまでの移動時間および乗車後のかご内の待機時間は含まない。
- サービス完了時間 (Total Journey Time) :
乗客がエレベータホールに到着後エレベータに乗車して行先階にて降車するまでの時間の平均値。エレベータホールまでの移動時間は含まない。

以下に、シミュレーション実験の出力結果を示す。

① 出力データ(乗車・降車)

表1・2はS4 Simulation Systemを用いてシミュレーションを行った際の乗車時・降車時の出力データである。表1・2では、シミュレーション開始から乗車・降車順に、それぞれ乗車時刻、降車時刻、エージェントID、エレベータID、行先階、乗車階、ホール待ち時間、サービス完了時間等を出力した。このデータをエージェントID毎に照らし合わせることで、分析を行っていく。

表 1

乗車	乗車時刻	エレベータID	エージェントID	ホール待ち時間	センシング直後の行先階	実際の行先階	乗車階
1	11	0	5	0	14	14	7
2	14	1	6	0	5	5	3
3	21	0	4	11	2	2	12
4	27	1	12	1	3	3	13
5	29	3	13	0	15	15	6
6	32	2	14	0	0	0	10
7	34	0	9	9	8	8	0
8	34	2	8	11	4	4	8
9	35	1	15	2	2	2	5
10	38	0	7	18	11	11	4
11	41	2	17	0	0	0	6
12	41	3	16	0	11	11	15
13	42	2	18	0	1	1	6
14	49	2	10	22	0	0	1
15	49	3	11	21	5	5	8
16	49	3	20	3	0	0	8
17	51	3	21	2	2	2	6
18	52	1	24	0	10	10	5
19	54	2	22	3	10	10	1
20	59	2	25	3	8	8	6

表 2

降車	降車時刻	エレベータID	エージェントID	サービス完了時間	乗車時間
1	16	1	6	2	2
2	18	0	5	7	7
3	31	0	4	21	10
4	37	1	12	11	10
5	38	3	13	9	9
6	39	1	15	6	4
7	42	0	9	17	8
8	44	2	8	21	10
9	45	3	16	4	4
10	46	0	7	26	8
11	48	2	18	6	6
12	50	2	14	18	18
13	51	2	17	10	10
14	52	2	10	25	3
15	52	3	11	24	3
16	56	3	21	7	5
17	59	3	20	13	10
18	61	2	25	5	2
19	63	1	24	11	11
20	64	2	22	13	10

② アニメーションモデル

図 5 は S4 Simulation System を用いてシミュレーションを行った際のアニメーションモデルの出力例である。幅 14, 高さ 16 の格子グラフ (x 軸と y 軸) からなっており, $x=0$ から乗客●が発生し, x 軸方向に水平に移動する。乗客は割当てられたかご■の地点 ($x=10,11,12,13$) まで移動し, かごが到着するまで待機する。かごが到着した後, 乗客は乗車し行先階へと移動する(乗降時にかかる時間は 1 秒)。図 5 の乗客●の値は行先階, かご■の値は乗車人数を表している。

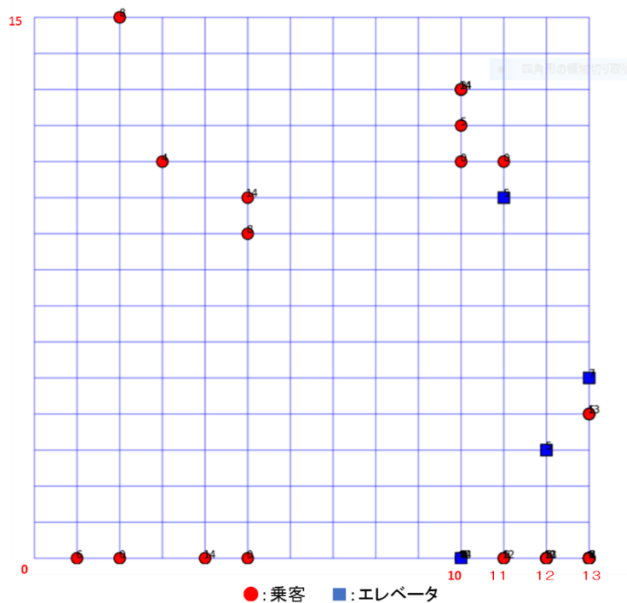


図 5

第 4 章

シミュレーション実験

4.1 シミュレーションモデルの設定

3 章で示したように、今回は 16 階床のオフィスビルを想定する。かごの台数は 4 台であり、1 階でのセンシング位置からエレベータホールまでの長さは、10[m]又は 20[m]を設定している。シミュレーション上では、センシング位置と乗客発生のポイントは同じであり、乗客がエレベータの乗降りにかかる時間は 1 秒とする。また、本研究では登録された階に移動する一般的な乗客だけでなく、登録された階とは異なる階に移動するイレギュラーな乗客を発生させる。乗客の発生・行先階の設定は 3 章で述べたとおりである。

シミュレーション時間(maxTime)を 10000 と設定し、イレギュラーな乗客の発生確率(Prob a)、到着間隔(arriveMean)、行先階の重み行列(ws)のパラメータを変更しながら、それぞれに対してシミュレーションを行い、平均待ち時間・サービス完了時間を計測する。

4.2 想定環境

以下に、シミュレーションの評価実験の基本仕様を示す。

- ① かご 4 台、かご定員 20 人
- ② 16 階床ビル (1F~16F, 1F: ロビー階) ※オフィスビルを想定。
- ③ 交通流パターン: OD (Origin and Destination) は、Up Traffic (朝方), Down Traffic (夕方), UpDown Traffic (昼) の 3 つの代表的な交通流パターンで評価を行う。
- ④ 乗客到着はポアソン到着 (ランダム到着) を想定。

乗客・エレベータのセンシング位置とエレベータの距離 x と歩行速度 v についての条件を以下に示す。一般のオフィスビルでは、1F (ロビー階) のエレベータホール近辺はセンシング機器の設置自由度が高く、一方上方のオフィス階はセンシング機器の設置スペースに余裕がない為、1F のセンシング位置 x に複数の値を適用し、上方階は 5m と設定した (図 6)。

- 1F: ホールからの距離 $x[\text{m}] = \{10\}$
- 2~16F: ホールからの距離 $x[\text{m}] = \{5\}$
- 乗客歩行速度 $v[\text{m/s}] = 1.0$

・イレギュラーな乗客の発生確率 $\text{Prob } a [\%] = \{0, 10, 20, 50\}$

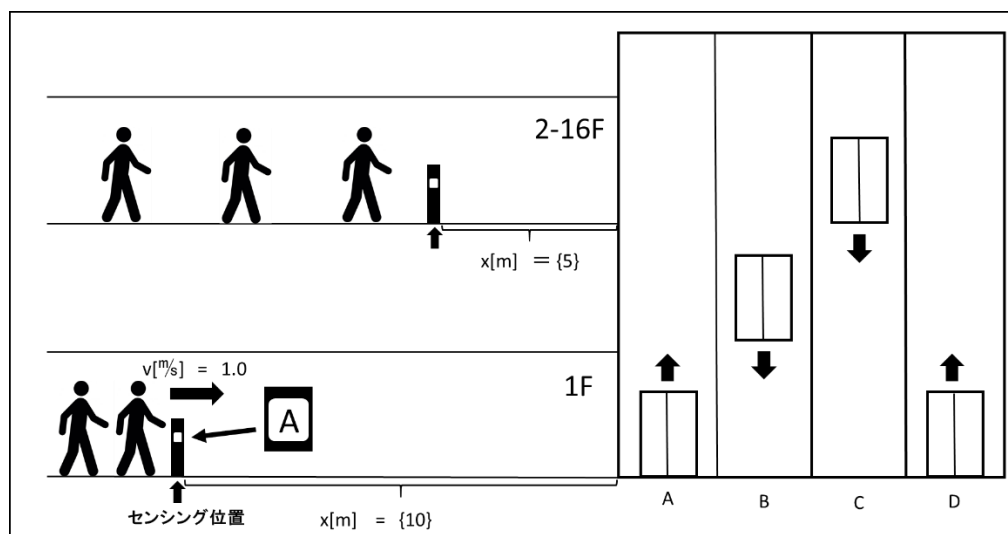


図 6

4.3 一般的な時間のシミュレーション結果比較

イレギュラーな乗客の発生確率 $\text{Prob } a [\%]$ と乗客の到着間隔について各パラメータを設定して比較する。各パラメータについて maxTime を 10000, センシング位置からホールまでの距離 x を 10(m), 行先階の重み行列 ws の値を 1 階が 3、2~16 階を 1(※以降, $ws(3.1)$ と表現する。)と固定し, $\text{Prob } a = [0\%, 10\%, 20\%]$ について, 平均待ち時間・サービス完了時間をそれぞれ比較したものが図 7、図 8 である。又, 乗客の到着間隔の値を 1 階を 10, 2~16 階を 30(※以降, $am(10.30)$ と表現する。), $am(20.30)$, $am(30.30)$ について, 平均待ち時間・サービス完了時間をそれぞれ比較したものが図 9, 図 10 である。今回は使用したパラメータは **UpDown Traffic(昼)**に相当する。なお, シミュレーションソフトは信頼区間 95%の区間推定も行っている。これらは付録に載せる。

① イレギュラーな乗客の発生確率 $\text{Prob } a [\%]$ についての比較

図 7 では, $\text{Prob } a$ の確率を 0,10,20 と上げていくにつれて, 全体的に平均待ち時間の数値は悪化する。Max=10000,x=10,ws(3.1),am(20.30)の 0%と 20%を比較した場合, 20%は 0%と比べて最大で 28%, 数値が悪化している。人物センシング情報を用いたかご割当ての際に, 乗客自身の情報だけではなく他階で待っている乗客のホールの到着時刻や行先階などの情報によって, かごの停止回数や待機時間が決定されていく。しかし, イレギュラーな乗客の発生数が増えれば増えるほど, 決定された停止回数や待機時間に誤差が生じて

しまい、エレベータの制御が悪くなるといえる。これは、他のどのパラメータを設定しても言えることであり、イレギュラーな乗客の発生確率 $\text{Prob } a(\%)$ を上げることは全体的に平均待ち時間の悪化につながる。

図 8 から、0%と 20%を比べた場合、全てのパラメータにおいてサービス完了時間の数値は悪化している。しかし、0%と 10%、10%と 20%を比べた場合、大小関係のバラつきが大きい結果となった。サービス完了時間の場合、イレギュラーな乗客の発生確率 $\text{Prob } a(\%)$ を上げてても数値の大小関係にバラつきがあり、必ずしも数値の悪化につながるとは言いきれない。

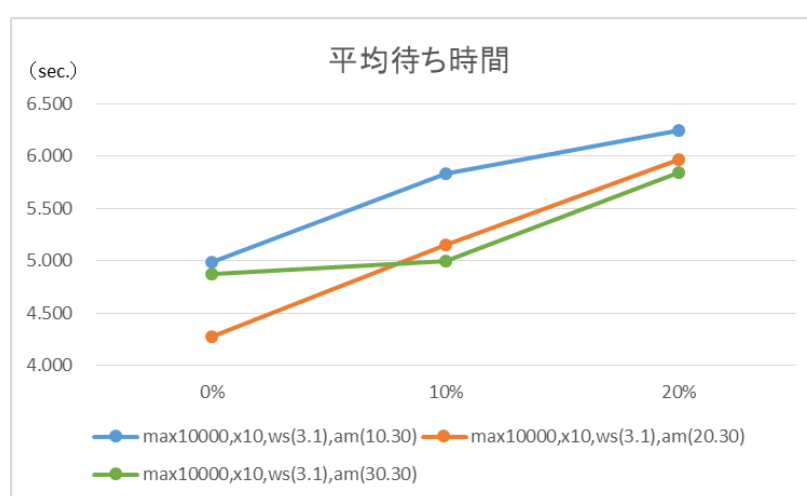


図 7

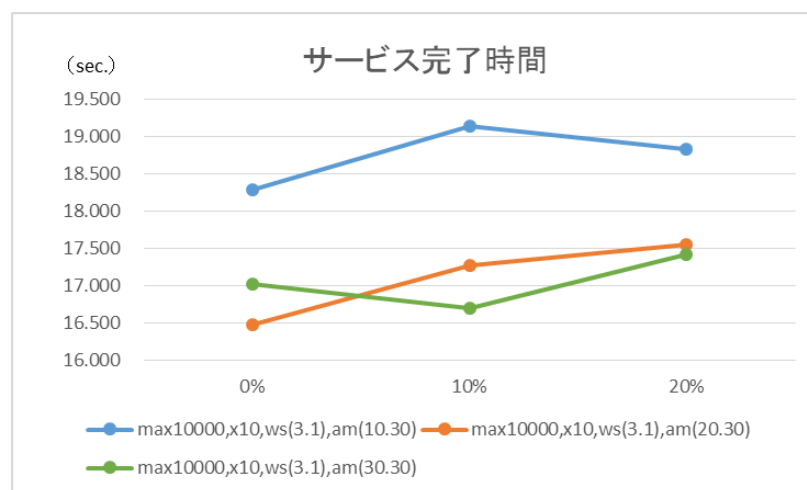


図 8

② 乗客の到着間隔についての比較

図9では、乗客の到着間隔を $am(10.30)$, $am(20.30)$, $am(30.30)$ にしていくにつれて、全体的に平均待ち時間の数値は小さくなる。これは1階の乗客の到着の発生頻度を少なくしていくことで、1回のシミュレーション当たりの乗客エージェントの発生数が少なくなったことが考えられる。これにより、エレベータの混雑度が下がり、待ち時間は少なくなる。また、乗客の発生数が少なくなるということは、それだけイレギュラーな乗客の発生数が少なくなるということであり、エレベータ制御に対して影響が少なくなる。これは、図10のサービス完了時間についても同じことが言える。

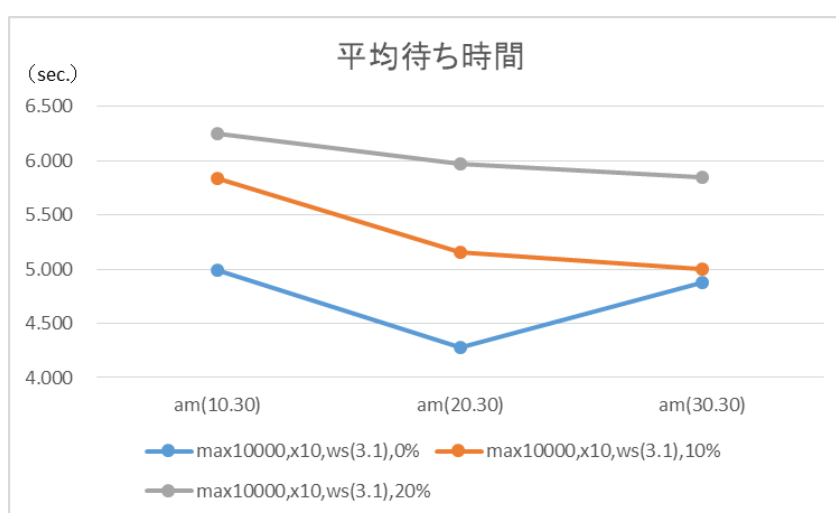


図9

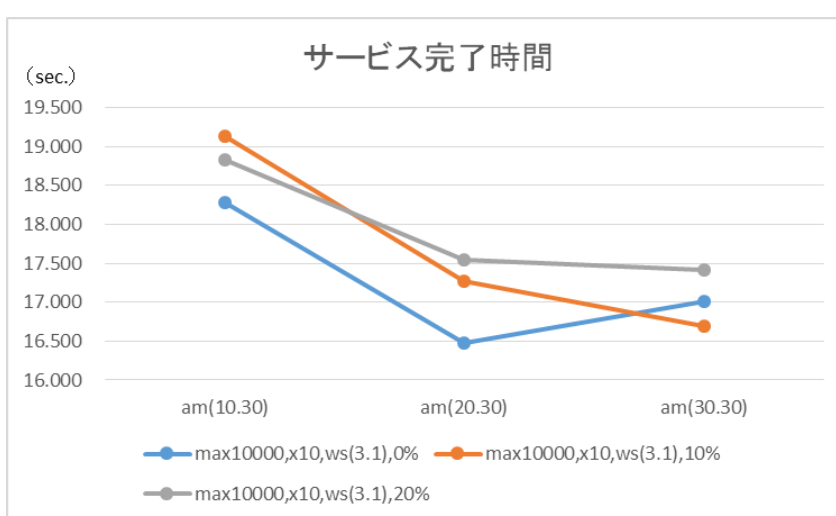


図10

4.4 時間帯別（朝、夕）によるシミュレーション結果比較

以下に時間帯別(Up Traffic(朝方)・Down Traffic(夕方))のシミュレーション結果を示す。時間帯別毎の結果に対して、イレギュラーな乗客の発生確率 Prob a(%)と乗客の到着間隔を軸として平均待ち時間・サービス完了時間の比較を行う。

(a) Up Traffic(朝方)の場合

Up Traffic の場合、朝方を想定している為、1階で発生する乗客の数が多く、1階から2~16階に移動する乗客の割合を高く、1階に下がる割合を少なくする。よって、各パラメータについて行先階の重み行列 ws の値を $ws(1.5)$ と固定し、Prob a = [0%, 10%, 20%] について、平均待ち時間・サービス完了時間をそれぞれ比較したものが図 11、図 12 である。又、乗客の到着間隔の値を $am(5.30)$, $am(10.30)$, $am(20.30)$ について、平均待ち時間・サービス完了時間をそれぞれ比較したものが図 13, 図 14 である。

① イレギュラーな乗客の発生確率 Prob a[%]についての比較

4.2 と同様にイレギュラーな乗客の発生確率 Prob a[%]が上がるにつれて、平均待ち時間の値は全体的に悪化した。UpDown Traffic(昼)に比べて、Up Traffic(朝方)の方が平均待ち時間・サービス完了時間の値は大きくなった。サービス完了時間についてはイレギュラーな乗客の発生確率 Prob a(%)を上げてても数値の大小関係にバラつきがあり、必ずしも数値の悪化につながるとは言いきれない結果となった。

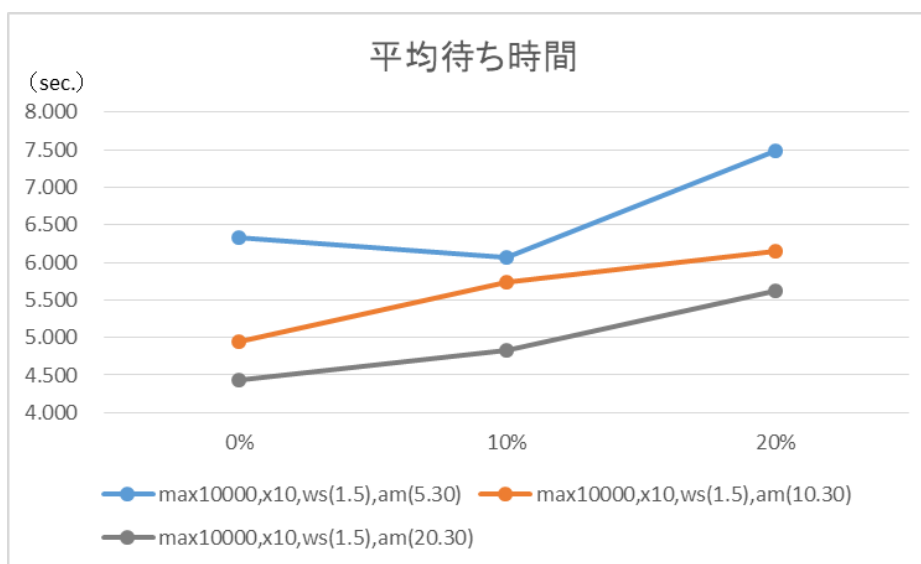


図 11

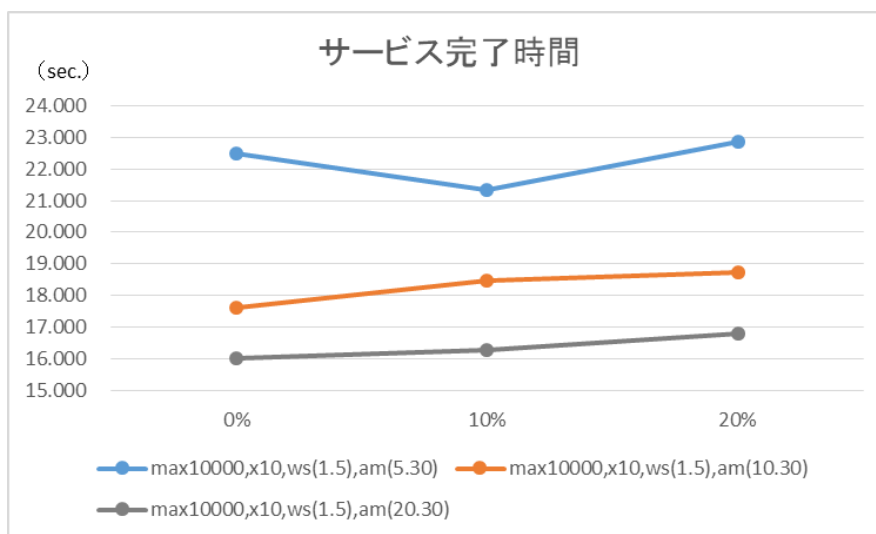


図 12

② 乗客の到着間隔についての比較

図 13 では、乗客の到着間隔を am(5.30),am(10.30),am(20.30)にしていくにつれて、全体的に平均待ち時間・サービス完了時間の数値は小さくなる。4.2 で述べたように乗客の発生頻度が少なくなるということは、イレギュラーな乗客の発生数が少なくなるということであり、エレベータ制御に対して影響が少なくなる。これは、図 14 のサービス完了時間についても同じことが言える。また、①と同様に平均待ち時間・サービス完了時間の値は UpDown Traffic(昼)に比べて、大きくなった。

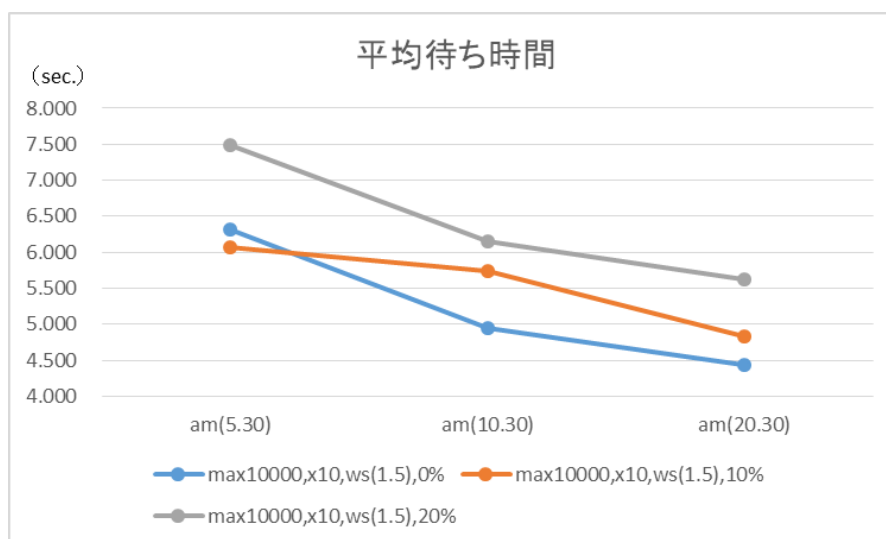


図 13

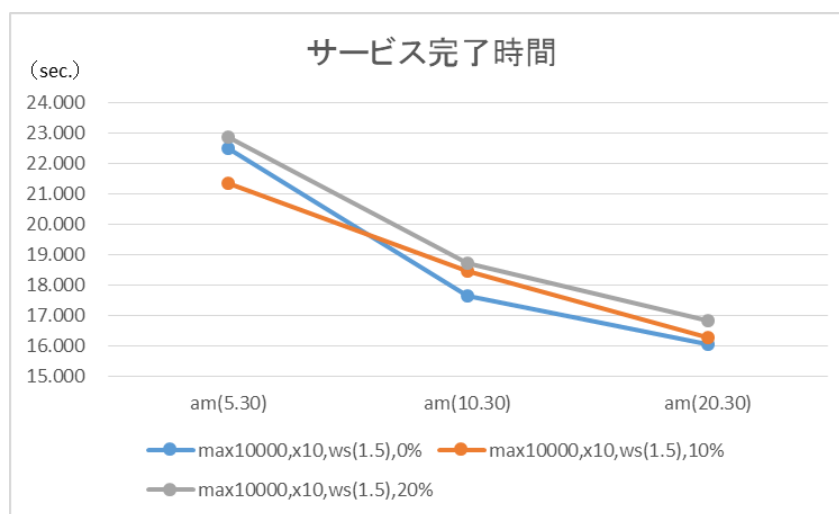


図 14

(b) Down Traffic(夕方)

Down Traffic の場合、夕方を想定している為、2~16 階で発生する乗客の数が多く、2~16 階から 1 階に移動する乗客の割合を高く、2~16 階に上がる割合を少なくする。よって、各パラメータについて行先階の重み行列 ws の値を $ws(5,1)$ と固定し、 $Prob a = [0\%, 10\%, 20\%]$ について、平均待ち時間・サービス完了時間をそれぞれ比較したものが図 15、図 16 である。又、乗客の到着間隔の値を $am(100.30)$, $am(100.25)$, $am(100.20)$ について、平均待ち時間・サービス完了時間をそれぞれ比較したものが図 17, 図 18 である。

① イレギュラーな乗客の発生確率 $Prob a[\%]$ についての比較

4.2 と同様にイレギュラーな乗客の発生確率 $Prob a[\%]$ が上がるにつれて、平均待ち時間・サービス完了時間の値は全体的に悪化した。UpDown Traffic(昼)に比べて、Down Traffic(夕方)の方が平均待ち時間・サービス完了時間の値は大きくなった。

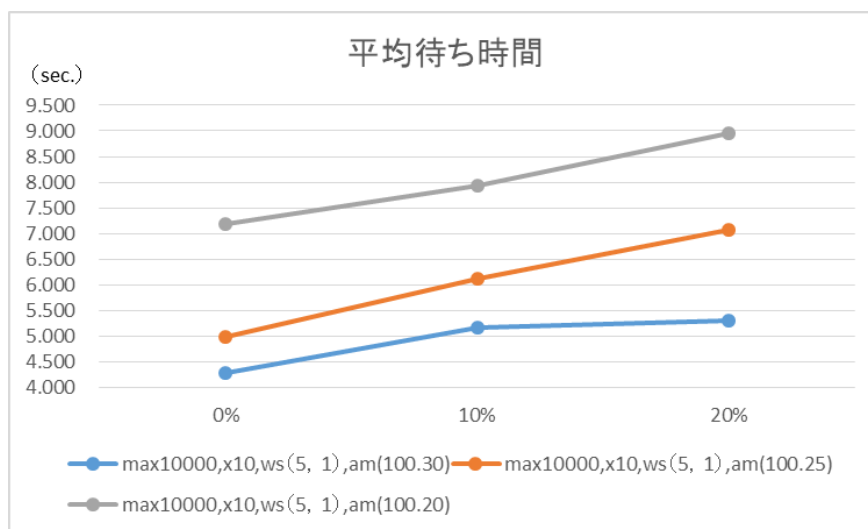


図 15

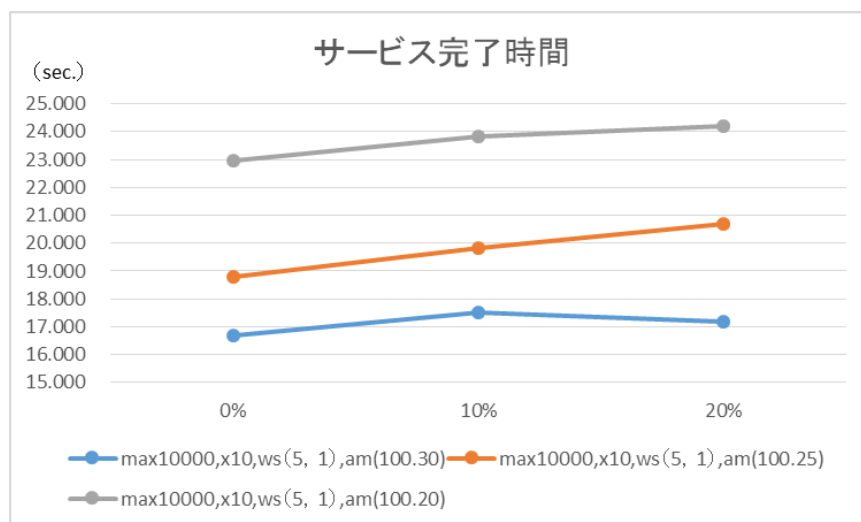


図 16

② 乗客の到着間隔についての比較

図 17 では、乗客の到着間隔を $am(100.20)$, $am(100.25)$, $am(100.30)$ にしていくにつれて、全体的に平均待ち時間・サービス完了時間の数値は小さくなる。4.2 で述べたように乗客の発生頻度が少なくなるということは、イレギュラーな乗客の発生数が少なくなるということであり、エレベータ制御に対して影響が少なくなる。これは、図 18 のサービス完了時間についても同じことが言える。また、①と同様に平均待ち時間・サービス完了時間の値は UpDown Traffic(昼)に比べて、大きくなった。

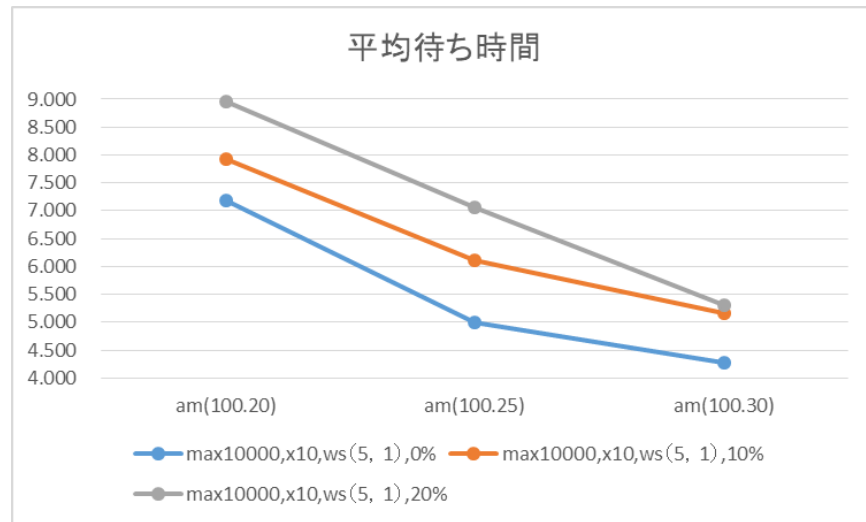


図 17

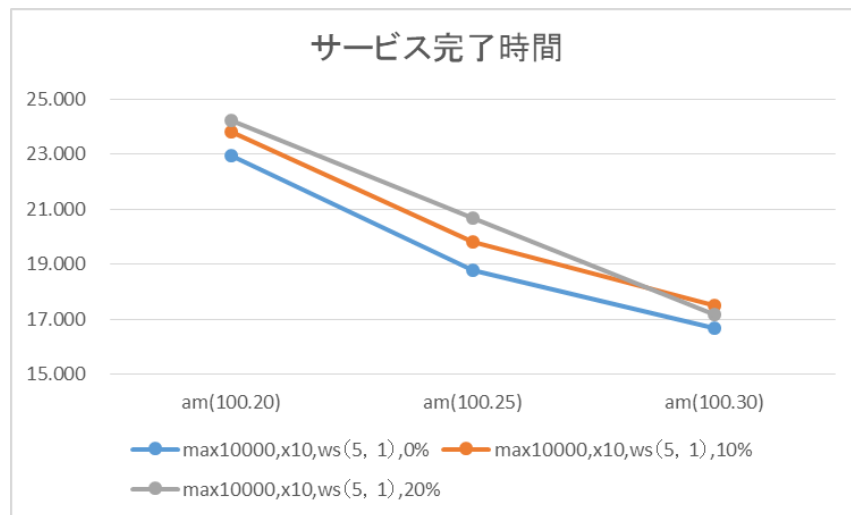


図 18

第 5 章

結論

本論文では、人物センシング情報を用いたかご割当に関して、日常的な場面を考慮する為センシング情報で割当てられたかごに乗車後、エレベータ内のボタンで本来とは異なる行先階を選択し、移動するイレギュラーな乗客を発生させ、どれほどの影響があるのか、シミュレーション解析を行い、評価を行った。イレギュラーな乗客の発生確率 $\text{Prob } a$ を上げると平均待ち時間の値は悪化した。しかし、サービス完了時間は各パラメータの設定によって悪化するものとししないものがあり、すべてのパラメータに必ずしも悪くなるとは言いきれない結果となった。また、乗客の各階の到着間隔別に比較をおこなったところ、乗客の到着(発生)頻度が多いパラメータ程、平均待ち時間・サービス完了時間の値は悪くなった。これは、単純に乗客の発生数が多くなり、ホールで待つ乗客が増えたことにより、エレベータの周回時間が長くなったのが原因だと考えられる。また、イレギュラーな乗客の発生確率が高いもの程、評価は悪かった。

本論文では人物センシング情報を用いたかご割当とエレベータの制御を別々のものと考えてシミュレーションを行った為、すべてのパラメータに関連性を持たせることができなかった。よって今後の課題として、人物センシング情報を用いたかご割当とエレベータの制御の両方を考慮したより優れたエレベータ管理方式を求めることが挙げられる。

参考論文

[1] 匹田士郎・阿部茂(2001)

エレベータ群管理制御における AI 技術の応用, 人工知能学会誌, Vol. 17, No. 1, pp. 57-62

[2] 鈴木直彦・小堀真吾・岩田雅史・山下桜子(2013)

人物センシング情報を用いたエレベータ群管理方式, 電気学会論文誌 C, Vol. 133, No. 7, pp. 1392-1401

[3] 小堀真吾・鈴木直彦・岩田雅史・山下桜子(2012)

総走行距離を短縮するエレベータ群管理割当方式, 電気学会論文誌 D, Vol. 132, No. 11, pp. 1016-1023

[4] 匹田士郎・駒谷喜代俊(1989)

ファジィ・ルールベースを用いた新しいエレベータ群管理システム, 計測自動制御学会誌, Vol. 25, No. 1, pp. 99-104

[5] M. Amano, M. Yamazaki, and H. Ikejima (1995)

The Latest Elevator Group Control System, Elevator Technology 6, Proc. of ELEVCON '95, pp. 88-95

[6] 匹田士郎・阿部茂・岩田雅史(2004)

行先階登録と適応制御によるエレベータ群管理, 電気学会論文誌 C, Vol. 124, No. 7, pp. 1471-1477

[7] J. Kim and B. Moon (2001)

Adaptive Elevator Group Control with Cameras, IEEE Trans. on Industrial Electronics, Vol. 48, No. 2, pp. 377-382

[8] T. Strang and C. Bauer (2007)

Context-Aware Elevator Scheduling, 21st International Conference on Advanced Information Networking and Application Workshops (AINAW '07), pp. 276-281