

ドローン物流システムにおけるモデル検査法を用いた 運行スケジュールの決定手法

辻 光顕 長谷部 浩二

近年の物流業界では、物流量の増加に伴うドライバー数の不足や交通渋滞の発生などが課題となっている。このため、ドローン（無人航空機）を用いて荷物の配送を行うシステムが注目されている。ドローンを用いた効率的な配送を実現するためには、各ドローンの飛行時間が最短になるようなスケジューリング（荷物の割り当てや飛行経路の決定）を行う必要がある。また、複数のドローンが限られた空間内に同時に存在することによる衝突の危険の排除や、定期的な充電などの制約条件を満たす必要もある。本論文では、こうした効率的かつ安全なドローンのスケジューリングを行うための手法を提案する。特にここでは、実時間システムの状態を網羅的に探索するモデル検査器である Uppaal を用いて、反例からスケジューリングを行う手法を用いる。また、本論文では、いくつかの具体例を用いて提案手法の有用性を示す。

1 研究の背景と目的

近年、物流業界では、通信販売の普及などによる物流量の急増に伴い、ドライバー数の不足や交通渋滞の発生などが大きな問題になっている。こうした問題に対処する方法として、ドローン（もしくは無人航空機：Unmanned Aerial Vehicles, UAV）を利用した物流システムに注目が集まっている。また、ドローンを用いた物流は、既存の輸送手段と比べて細かい移動が可能となることから、商用のサービスに限らず災害時の支援物資の輸送手段としても期待が高まっている。こうした背景から、ドローンによる物流システムの研究開発が数多く行われてきた。

例として、海外では米国企業の Amazon 社が「Prime Air」と呼ばれる商品の配送計画を進めている [2]。これは、従来のトラックによる配送のコストや時間を削減することを目的として、ドローンを

用いて商品を保管している配送センターからそれらを直接顧客の下に届けるというものである。さらに、Amazon 社は複数のドローンが隊列編成を行うことで、単一のドローンによる配送に比べ、より重量のある荷物を配送する方法も提案している。

また、国内では KDDI 社が「スマートドローン」と呼ばれるプロジェクトを進めている [1]。このプロジェクトは、災害時の物資支援やその状況の把握などを行うことを目的としてドローンを運用するというものである。このプロジェクトでは、KDDI 社が保有する携帯無線基地局を利用することによって、自動かつ広域でのドローン運用を提案している。

ドローンを用いた物流システムにおいて、効率的な配送を実現するためには、複数あるドローンの飛行時間が最短となるようなスケジューリング（輸送する荷物の割り当てや飛行経路の決定）が求められる。一方で、複数のドローンが限られた空間内に同時に存在することによる衝突の危険の排除や、定期的な充電、さらには隊列の編成のためのドローン間の同期など、さまざまな制約条件を満たす必要もある。

このような衝突回避に関する先行研究として、例えば論文 [15] では、ドローン同士による衝突回避をヒューリスティクスによってスケジューリングする手

Collision-free Scheduling of Drone Delivery Systems Using Model Checking.

Mitsuaki Tsuji, 筑波大学情報学群情報科学類, College of Information Science, University of Tsukuba.

Koji Hasebe, 筑波大学システム情報系情報工学域, Faculty of Engineering, Information and Systems, University of Tsukuba.

法を提案している。また論文[8]では、複数のドローンが充電を繰り返しながら運行するシステムを対象に、メタヒューリスティクスを用いてドローンの飛行距離を最適化する手法について述べている。また、本論文と同じ手法を用いた先行研究として、論文[5][6]ではタスクのスケジューリングを Uppaal[3] と呼ばれるモデル検査器によって行う手法について述べている。

一方で、本論文における物流システムでは、複数のドローンが単体あるいは隊列を編成して運行するものとなっている。そのため、ドローン同士による衝突の危険性がない最適な運行スケジュールを求めめるためには、既存の手法を直接適用することが難しいと考えられる。

このような課題に対して、著者らはモデル検査法を用いてシステムの状態を網羅的に探索することにより、ドローン同士の衝突が発生しない最適な運行スケジュールの決定手法について提案する。特にここでは、複数のドローンが隊列を編成しながら荷物の配送を行うようなシステムについてモデル化を行い、運行スケジュールを決定する手法について述べる。具体的には、論文[5][6]のアイデアを基に、モデル検査器 Uppaal によってドローン同士の衝突が発生しないような反例を求めめることで、各ドローンの飛行時間が最短となる運行スケジュールを求めめる方法について説明する。

本論文の構成は次の通りである。第2章では著者らが対象とする物流システムの概要について説明し、第3章で物流システムのモデル化について述べる。また、第4章で物流システムの運行スケジュールを決定する手法について説明し、第5章でモデル検査法による検証結果を示す。第6章では関連研究について述べ、最後に第7章で結論および今後の課題について述べる。

2 物流システムの概要

この章では、本研究で対象とする物流システムの概要について説明する。

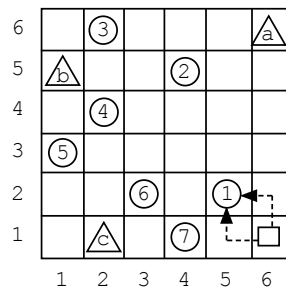


図1 運行エリアの具体例1

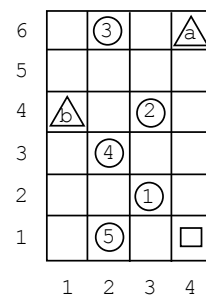


図2 運行エリアの具体例2

2.1 ドローン

ここで扱う物流システムは、人による操作なしで飛行可能なドローンによって構成される。ドローンは配送する荷物の重量に応じて、単体あるいは複数のドローンが隊列を編成することによって飛行を行う。また、これらのドローンは内蔵されたバッテリーによって一定時間連続して飛行することが可能であり、随時充電を行うことで配送を継続することができる。

2.2 運行エリア

この物流システムでは、あらかじめドローンが配送を行うためのエリアが存在し、これを運行エリアと呼ぶ。図1と図2に運行エリアの具体例を示す。

運行エリアは格子状に区切られており、各マスの中には荷物を輸送するための拠点である基地や、それらを届ける場所である配送先が存在する。さらに、運行エリアには充電スペースが複数設置されており、これを利用して随時ドローンのバッテリーを充電する

ことができる。

例えば、図 1 においてマスの中に示された四角と丸はそれぞれ基地、配送先を表し、三角は充電スペースを表している。各配送先に描かれた数字は基地を出発したドローンが移動するための順番を表している。また、充電スペース上の文字はこれらを区別するためのものである。図 1 において、ドローンは配送先 1 から配送先 2 へ移動する場合に充電スペース a を通過することが可能である。同様に、配送先 3 から配送先 4 へ移動する場合に充電スペース b を通過することが可能である。

2.3 ドローンの運行

第 2.2 節で述べたように、ドローンは通過する配送先の順番が決まっており、その順番に応じてマスを縦横移動することが可能である。例として、図 1 の矢印で示すような 2 つの経路が存在する。なお、マスを移動するために必要な距離は全て等しいものとする。また、ドローンは配送できる荷物の重量に制約があるため、2 地点間を単体で移動する場合と、隊列を編成して移動する場合がある。例えば図 1 の例では、配送先 1 と配送先 2 の間は複数のドローンが隊列を編成して移動し、それ以外の地点間では単体のドローンによって移動を行うというようなものである。

2.4 隊列の編成

この物流システムでは、複数のドローンが隊列を編成して移動することができる。隊列は、行先を同一とする複数のドローンによって、基地あるいは配送先に存在する地上のスペースを利用して構成される。そのため、ドローンは隊列を編成する際に一度着陸する必要がある。着陸しているドローンと飛行しているドローンが衝突する可能性は低いため、隊列を編成する場合には、一時的に複数のドローンが同じマスに存在することが可能である。ドローンは配送時間をできるだけ短くするため、隊列を編成可能であるものから順に編成を行うものとする。

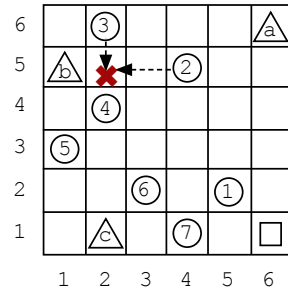


図 3 衝突による危険性の例

2.5 衝突の危険性

本論文における物流システムでは、同時に複数のドローンが移動する。そのため、運行スケジュールによっては、同じ時刻に複数のドローンが狭い領域に存在することによって衝突が起きてしまう可能性がある。その状況を図 3 に示す。

図 3 では、バツ印で示した位置に複数のドローンが同時に存在しており、衝突の危険性があることを示している。これは配送先 2 から配送先 3 へ飛行するドローンと、配送先 3 から配送先 4 へ飛行するドローンが同時に同じマスに存在していることを表している。この例では 1 つのマスに 2 つ以上のドローンが存在することを衝突と見なしている。しかし、どの程度の接近を衝突と見なすか、あるいはドローンの大きさといった物流システムの特徴に依存してマスの大きさは変化する。また、各マスを細かくしていくことによってモデルの精度を高めることができる。一方で、精度を高めた場合にはモデルの状態数が多くなるため、運行スケジュールを決定することが困難になる可能性がある。したがって、モデル化を行う場合には実際の物流システムに応じた判断を行う必要がある。

3 ドローン物流システムのモデル

この章では、2.5 節で示した衝突を防ぐスケジュールを決定するため、物流システムをモデル化する方法について述べる。

3.1 ドローンのモデル

ドローン全体の集合を D で表す。全てのドローンは隊列を編成することが可能であるが、その上限を表す関数を $max_fleet : D \rightarrow \mathbb{N}$ と定義する。(ただし、 \mathbb{N} は自然数全体の集合である。)

また、ドローンはバッテリーを使用して動作するため、連続して飛行可能な距離に上限がある。そのため、その上限を超えない範囲でドローンは充電スペースを訪れる必要がある。ドローンが運行エリア上のマス連続して移動できる回数の上限を表す関数を $max_range : D \rightarrow \mathbb{N}$ と定義する。

3.2 運行エリアのモデル

本論文では、運行エリアを図1や図2のような格子状の m 行 n 列からなる矩形を用いて表現し、その集合を $A \subseteq \mathbb{N} \times \mathbb{N}$ で表す。運行エリア上の各マスをセルと呼び、その位置を順序対 $\langle x, y \rangle$ (ただし、 $1 \leq x \leq n, 1 \leq y \leq m$) で表現する。運行エリアには、ドローンのバッテリーを充電することが可能な複数の充電スペースが存在し、その集合を $E \subseteq A$ と表す。

また、各マスには衝突の危険を回避するため、同時に飛行可能なドローンの数に対する上限が定められている。そこで、その上限を表す関数を $cap : A \rightarrow \mathbb{N}$ と定義する。ただし、ドローンは隊列を編成および解除する際に着陸するため、その場合は同じセル上にドローンが存在しても衝突の危険性はないものとする。同様に、隊列飛行しているドローンは単一のドローンと見なし、同じセル上に存在している場合でも衝突の危険性がないものとする。

3.3 飛行ルートのモデル

ドローンの飛行ルートを $a_1, \dots, a_n \in A$ のように、運行エリア A 上におけるセルの列を用いて表す。ただし、各 $a_i = \langle x_i, y_i \rangle, (1 \leq i \leq n)$ について、 $x_i = x_{i+1}$ かつ $|y_i - y_{i+1}| = 1$ 、または、 $y_i = y_{i+1}$ かつ $|x_i - x_{i+1}| = 1$ であるとする。また、全てのドローンは基地を起点として運行を行ってから再び基地に戻るため、 $a_1 = a_n$ である。ドローンは連続して飛行可能な距離に制限があるため、それを超えない範囲で充電スペースを訪れる必要がある。す

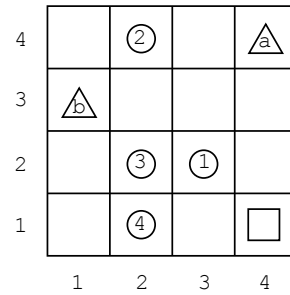


図4 対象とする運行エリア

なわち、運行ルートにおける任意の部分列に対し $\forall i, j, k [(a_i, a_j \in E) \wedge \forall k (i < k < j \rightarrow (a_k \notin E)) \rightarrow j - i \leq max_range]$ である。

4 モデル検査法によるスケジュールの決定

4.1 対象とする物流システムのモデル

2.2節において具体的な運行エリアを示したが、ここでは図4のような簡単なモデルを対象とする。

また、このモデルでは複数のドローンが配送先1から配送先2の間を隊列によって移動するものとする。さらに、充電スペース a はドローンが配送先1から配送先2を移動する途中に訪れることができるものとする。同様に、充電スペース b は配送先2から配送先3の間を移動する途中に訪れることが可能であるとする。ここでは、全ての $a \in A$ および $d \in D$ において、 $max_fleet(d) = 2, max_range(d) = 8, cap(a) = 2$ である。

4.2 Uppaal による物流システムのモデル

第3章で用いたモデルを、モデル検査器 Uppaal によって記述する方法について説明する。Uppaal では対象のシステムを GUI を用いて記述し、システムの各状態をロケーション、状態間の遷移をエッジとして表現する。前節で述べたドローンのモデルを Uppaal によって記述したものを図5に示す。

このモデルでは、基地や配送先などのドローンが存在できる位置をロケーションで表し、その間を接続する運行経路をエッジで表現している。また、それぞれのドローンにとって固有な時間変数 lc および、モデル全体の時間変数 gc を定義している。ここで扱うモ

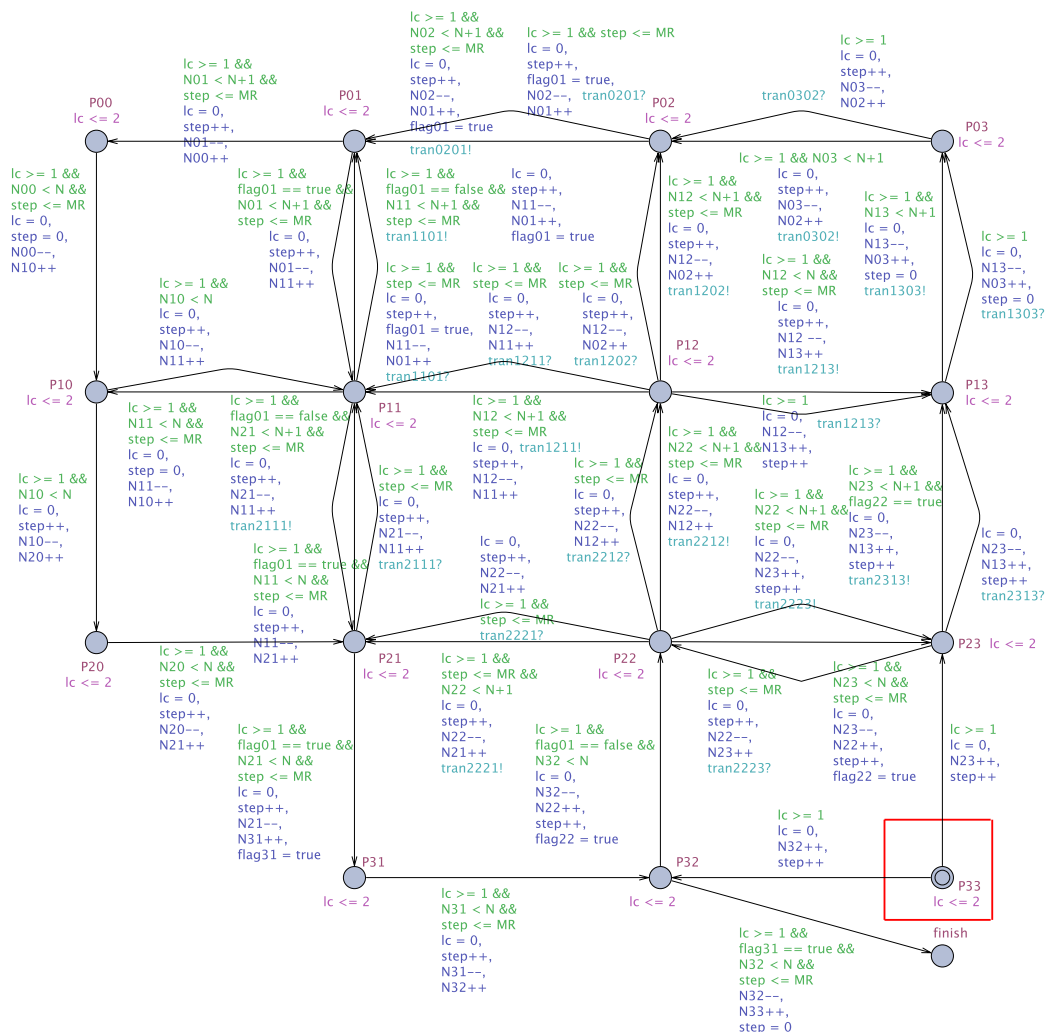


図5 Uppaalによるドローンのモデル

デルにおいて、ドローンは隣接する地点間を1単位時間あるいは2単位時間の範囲で遷移し、連続して最大MR個のロケーションを移動することができる。なお、この値は関数 max_range によって与えられるが、図5の例では全て8としている。ドローンは図5の赤い四角で囲まれたロケーションであるP33を起点とし、最終的にロケーション finish に遷移する。また、隊列の編成はUppaalにおけるバイナリチャネルを用いて表現され、これによって複数のドローンが通信を行って協調動作することができる。

4.3 検証すべき性質

物流システムが最低限満たされなければならないことは、「全てのドローンがいつかは必ず全ての目的地を通過して戻ってくる」ような運行スケジュールが存在することである。すなわち、以下の性質が満たされる必要がある。

(P1) 全てのドローンは、全ての配送先を通過してから出発地点に到達可能である。

この性質が満たされれば、全てのドローンが配送を行ってから出発地に戻るようなスケジュールを決定することができる。反対に満たされなければ、運行スケ

スケジュールを決定することができない問題であることが分かる。この性質を検証することによって、対象とする物流システムにおける運行スケジュールがあらかじめ決定されている場合に、衝突が発生することなく運行が可能か否かを確認することができる。性質 (P1) が満たされる場合は、論文 [5] の手法と同様に、性質 (P1) の否定となる性質を検証することによって最小時間かつ衝突が発生しないような運行スケジュールを求めることができる。すなわち、以下の性質を検証すればよい。

(P2) 全てのドローンは、出発してから再び出発地点に到達することはない。

性質 (P2) を検証することによって、Uppaal はこの性質が満たされないような反例を検出して出力する。また、Uppaal ではモデル全体の経過時間 gc が最小となるような反例を出力することができる。これによって、衝突の発生がなくドローン全体の運行時間が最小となるような運行スケジュールを得ることができる。

なお、ドローンの衝突が発生しないような運行スケジュールを求めたい場合には、性質 (P2) のみを検証してもよい。なぜなら、性質 (P2) が満たされる場合はドローンが衝突を回避できないことを意味しており、これは性質 (P1) が満たされないことと同値なためである。同様に、性質 (P2) が満たされない場合は、性質 (P1) が満たされることと同値である。

4.4 TCTL 式による検証

Uppaal における検証式は、時間付計算木論理 TCTL の部分論理を使って記述することができる [4]。まず、性質 (P1) については、以下のように記述することができる。

(L1) $E\Diamond (d1.finish \text{ and } d2.finish)$

論理式 (L1) において、 E はシステムの実行列におけるあるパスの存在を表し、 \Diamond はある命題がいつかは成立することを表す。また、 $d1.finish$ はドローン d_1 がロケーション $finish$ に存在することを示す。す

なわち論理式 (L1) は、ドローン d_1 およびドローン d_2 が、いつかはロケーション $finish$ に到達するような実行列が存在することを意味する。

また、性質 (P2) を検証する論理式は以下のように記述することができる。

(L2) $A\Box \text{ not } (d1.finish \text{ and } d2.finish)$

論理式 (L2) において、 A はシステムの実行列において全てのパスを表し、 \Box はある命題が常に成立することを表す。すなわち論理式 (L2) は、 d_1 および d_2 が、ロケーション $finish$ に到達することがないことを意味している。

5 検証結果

この章では、第 4 章で説明したモデルおよび論理式を用いて実際に検証した結果について述べる。

5.1 2つのドローンにおける検証結果

4.1 節に示した運行エリアに対し、2つのドローンが運行するモデルについて検証した結果、性質 (P1) は満たされる結果となった。また、性質 (P2) について検証すると反例が得られ、衝突が発生しないような最小時間の運行スケジュールを検出することができた。なお、検証にかかった時間は 0.002 秒であり、最小時間は 13 単位時間である。

5.2 4つのドローンにおける検証結果

また本研究では、前節の例よりもドローンの数を増やして検証を行った。4つのドローンが存在する場合でも、性質 (P2) を検証した結果、先の例と同様に、衝突が発生しないような最小時間の運行スケジュールを得ることができた。なお、検証にかかった時間は 8.6 秒であり、最小時間は 15 単位時間である。

5.3 6つのドローンにおける検証結果

これまでに述べた例に対し、さらにドローンの数を増やして検証を行った結果について述べる。モデルにおけるドローンの数を 6 つにして検証を行うと、検証に必要なメモリ量が不足してしまうことによって検

証ができないという結果になった。これはモデル検査法を用いる場合によく見られる問題であり、状態爆発と呼ばれる。このような問題の対策としてはモデルにおける変数を減らしたり、より小規模な問題に切り分けて考えるなどの工夫が必要となる。

6 関連研究

これまで、数多くの物流システムや交通システムを対象に、船舶や車両などの衝突回避に関する研究がなされてきた。例として、論文[10]では船舶の運行における衝突の問題に注目し、蟻コロニー最適化を用いて衝突回避を行う手法について述べている。また、論文[13]では、災害時に複数のドローンを活用するという目的で運用した場合に、全てのドローンが要求される動作を行うか否かについて Uppaal を用いた研究を行っている。本論文と同様の手法を用いたものとしては、論文[5]が工場の機械に対する最適な動作スケジュールを求める手法について提案している。

しかし、本研究で扱ったような隊列編成可能な複数のドローンが移動する物流システムを対象に、その運行スケジュールをモデル検査法によって決定する手法を提案した研究はこれまでに行われていない。一般的に、このような複雑な振る舞いを持つ物流システムの運行スケジュールを決定することは容易ではないため、本研究の提案した手法は有益なものであると考えられる。

7 結論と今後の課題

本論文では隊列編成可能な複数のドローンが動作する物流システムを対象に、その運行スケジュールをモデル検査法によって決定する手法について提案した。複雑な振る舞いを持つシステムに対して、最短時間となる運行スケジュールを決定することができた。本論文では、ドローン物流システムを対象にその運行スケジュールを決定する手法について述べたが、この手法は他の物流システムに対しても適用することが可能であると考えられる。今後は、より複雑な問題を扱った場合に起こる状態爆発問題に対応するため、モデルを小さな問題に分割することによりスケジューリ

ングを可能とするような課題に取り組む予定である。
参考文献

- [1] <http://news.kddi.com/kddi/corporate/newsrelease/2017/05/09/2422.html>
- [2] <http://www.amazon.com/primeair/>
- [3] <http://www.uppaal.org/>
- [4] E. M. Clarke, O. Grumberg, and D. Peled. Model Checking, The MIT Press, 1999.
- [5] A. Fehnker, Scheduling a Steel Plant with Timed Automata. *Real-Time Computing Systems and Applications(RTCSA1999)*, pp.280–286, 1999.
- [6] G. Behrmann, K. G. Larsen, J. I. Rasmussen, Optimal Scheduling Using Priced Timed Automata. *SIGMETRICS Perform. Eval. Rev.*, pp.34–40, 2005.
- [7] G. Sirigineedi, A. Tsourdos, R. Żbikowski, B. A. White, Modeling and Verification of Multiple UAV Mission Using SMV. *arXiv e-prints*, 2010.
- [8] J. Kim, B. D. Song, J. R. Morrison, On the Scheduling of Systems of UAVs and Fuel Service Stations for Long-Term Mission Fulfillment, *Journal of Intelligent & Robotic Systems*, vol.70, no.1, pp.347–359, 2013.
- [9] J. S. Bellingham, M. Tillerson, M. Alighanbari, J. P. How, Cooperative Path Planning for Multiple UAVs in Dynamic and Uncertain Environments. *41st IEEE Conference on Decision and Control(CDC2002)*, pp.2816–2822, 2002.
- [10] M. C. Tsou, C. K. Kuang, The Study of Ship Collision Avoidance Route Planning by Ant Colony Algorithm. *Journal of Marine Science and Technology*, vol.18, No.5, pp. 746–756, 2010.
- [11] M. M. Quottrup, T. Bek, R. I. Zamanabadi, Multi-Robot Planning: A Timed Automata Approach. *International Conference on Robotics and Automation(ICRA2004)*, pp.4417–4422, 2004.
- [12] N. Agatz, P. Bouman, M. Schmidt, Optimization Approaches for the Traveling Salesman Problem with Drone. *ERIM Report Series Reference*, No.ERS-2015-011-LIS, 2015.
- [13] N. Li, D. Bai, W. Jiao, Z. Yang, Verifying Stochastic Behaviors of Decentralized Self-Adaptive Systems: A Formal Modeling and Simulation Based Approach. *arXiv e-prints*, 2017.
- [14] T. Yu, J. Tang, L. Bai, S. Lao, Collision Avoidance for Cooperative UAVs with Rolling Optimization Algorithm Based on Predictive State Space. *Applied Science*, vol.7, No.4, 2017.
- [15] Y. Khosiawan, I. Nielsen, Indoor UAV Scheduling with Restful Task Assignment Algorithm, *arXiv e-prints*, 2017.
- [16] X. Zou, R. Alexander, J. McDermid, On the Validation of a UAV Collision Avoidance System Developed by Model-Based Optimization: Challenges and a Tentative Partial Solution. *International Conference on Dependable Systems and Networks Workshop (DSN-W2016)*, pp.192–199, 2016.