

緊急度と網羅性を考慮した効率的な空撮経路の生成

中村 優太[†] 井ノ口宗成^{††} 北原 格[†] 森嶋 厚行[†]

[†] 筑波大学 〒305-8571 茨城県つくば市天王台1丁目1-1

^{††} 静岡大学 〒422-8017 静岡県静岡市駿河区大谷836

E-mail: [†]s1621630@u.tsukuba.ac.jp, ^{††}inoguchi@inf.shizuoka.ac.jp, ^{†††}kitahara@iit.tsukuba.ac.jp,
^{††††}mori@slis.tsukuba.ac.jp

あらまし 自然災害時の被災地の情報の入手などにおいて、空撮画像は重要な情報源である。被災地の空撮を行う際、まず緊急に撮影すべき箇所を回る必要がある。一方、どの地点が被災しているかという情報を全て集めることは難しいため、網羅的な撮影も求められる。しかし、緊急度を優先した効率的な経路と、網羅性を優先した効率的な経路は必ずしも両立しない。そこで、本論文ではA*探索アルゴリズムを利用し、緊急度が高いものを優先して回りつつ、残りの箇所を回る際には既に通った箇所をできるだけ通らない経路の生成手法を提案する。シミュレーション実験によって、本手法が単純に緊急度の高い箇所から回る手法と比べ経路の重複を半分以上削減し、あるケースでは空撮経路を19%短くできることを確認した。

キーワード 無人航空機、災害、経路探索

1. はじめに

地震や津波などといった自然災害においては、迅速な復旧・復興活動が重要となる。このとき、どこかの建物が被災しているかという情報を収集する必要がある。このような被災箇所を特定しなければならない状況において、空撮画像は重要な情報源である。特に、近年の無人航空機の普及により、容易に空撮画像が入手可能になり、注目を集めている。

被災地には主要な道路や発電所などの重要箇所や、被災したという情報が得られた地点のような緊急に状況を確認すべき箇所が存在する。そのため、空撮を行う際には緊急度によって重み付けられた領域に対しての経路探索を行い、緊急度が高い箇所から撮影して回ることが求められる。

一方、経路決定の際に利用する情報が全て正確で揃っているとは限らないため、領域を網羅するように撮影することも求められる。これは、情報が集まらない場所でも被災している状況や、被災地において予想外の場所が被災している状況が考えられるためである。

しかし、緊急度を優先した情報入手のための効率的な経路と、網羅的な情報収集のための効率的な経路は必ずしも両立しない。単純に緊急度が高い順に撮影を行うと、残りの未探索箇所を網羅的に撮影する際には重複が多い無駄な経路になる可能性がある。無人航空機での空撮を考えた場合では、バッテリー制約が厳しいために重複が多いと経路が長くなり領域を回りきれない。一方、網羅的に探索することを重視した経路だと緊急度が高い箇所を優先して回ることができない。したがって、図1のように、緊急度を優先しつつも後に網羅的に回ることを考慮した経路を生成する必要がある。図1において経路A・Bは同じ長さの経路であるが、経路Bはまだ通っていない箇所を分断するような経路となっており、後で網羅的に回る際に既に通った箇所を必ず通らなければならないため、経路Aと比べて

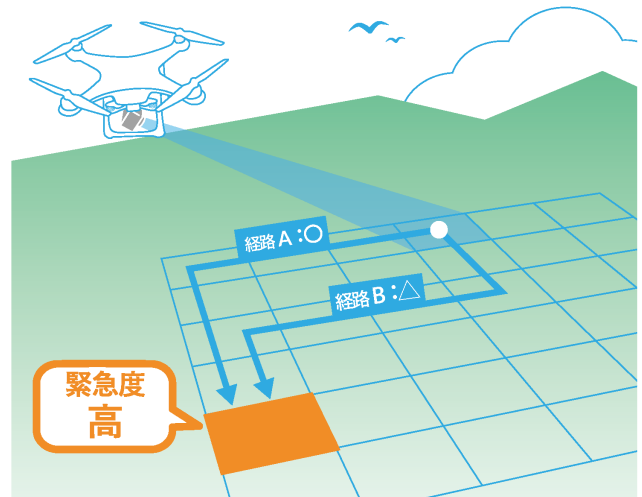


図1: 緊急度だけでなく網羅性も考慮した経路の例（経路Aと経路Bは同じ長さの経路だが、網羅のことを考えると経路Aの方が望ましい）

望ましくない経路といえる。

本稿では、A*探索アルゴリズム [1] を利用して、緊急度と網羅性を共に考慮した空撮経路の生成手法を提案する。提案する手法では、必ずしも最短経路ではなくとも緊急度が高いものを優先しつつ、後で残りの未探索箇所を回る際にできるだけ既に通った箇所を通らなくて済むような経路の生成を行う。このような経路にするために、できるだけ未探索箇所がひとまとまりになるようなヒューリスティクスを用いている。加えて、単純に緊急度の高い箇所から回る手法（以下、単純な手法）と本手法との比較シミュレーションを行い、実際にどの程度経路を短くできるのかを確認した。

本研究の貢献は次の2点である。

- 緊急度と網羅性を共に考慮した空撮経路生成手法の提案
- 単純な手法と提案手法の比較シミュレーションを行い、

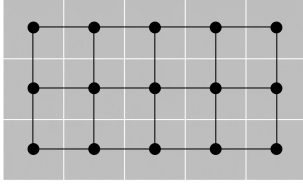


図 2: セルに分割された領域へのグラフの配置

実際にどの程度経路を短くできるのかを確認

本稿の構成は次の通りである。2. 節で取り組む問題を定義する。3. 節で関連研究を整理し、本研究との差異を示す。4. 節で提案する手法について述べる。5. 節でシミュレーションとその結果について述べる。6. 節でまとめを述べる。

2. 問題定義

本節では、まず本研究で取り組む問題について定義し、その後、利用する評価関数について述べる。

情報収集を行う領域として矩形の領域を想定する。この領域は正方形のセルで分割され、各セルに緊急度による重み付けがなされているものとする。このとき、領域上に、図 2 のようなセルの中央に格子点としてノードがある格子状のグラフ構造を考える。各ノードはセルに対応した重みを持つ。この重み付きグラフを $G = (V, E, w)$ とおく。頂点の集合を $V = \{v_s, v_1, v_2, \dots, v_n\}$ 、辺の集合を $E = \{e_1, e_2, \dots, e_m\}$ 、各頂点に対しその重みを与える関数を $w : V \rightarrow [0..1]$ とする。

上記のようなグラフ G とスタートノード v_s ($1 \leq s \leq n$) とパラメータ α ($0 \leq \alpha \leq 1$) が与えられたとき、 V 中の頂点を全て含み、かつ $\alpha e_1(R) + (1 - \alpha)e_2(R)$ が最大となるような長さ l の歩道 $R = [(v_{i1}, v_{i2}), (v_{i2}, v_{i3}), \dots, (v_{il}, v_{il+1})]$ を求める。ここで、 $e_1(R)$ は重みの大きいノードをどれだけ優先して通ったかを評価する関数、 $e_2(R)$ はどれだけ重複を避けることができたかを評価する関数として、 α は 2 つの評価関数のどちらを優先するかを調節するパラメータとして定義する。前述の通り、 $e_1(R)$ と $e_2(R)$ はトレードオフの関係になる場合があるため、パラメータ α によってどちらを優先するかを調節する。各評価関数について次に述べる。

評価関数 $e_1(R)$ は次のように定義する。

$$e_1(R) = 1 + w(v_1) + w(v_2)^2 + \dots + w(v_n)^l \quad (1)$$

重みは $0 \leq w(v) \leq 1$ の範囲をとるため、評価関数 $e_1(R)$ は、先に回ったノードの重みの影響は大きく、後から回ったノードの重みの影響は小さく評価する。したがって、重みが大きいノードをより優先して先に回ることによって返り値は大きくなり、後から回れば返り値は小さくなる。

評価関数 $e_2(R)$ は次のように定義する。

$$e_2(R) = 1 - \frac{l - n}{l} \quad (2)$$

評価関数 $e_2(R)$ では、経路中の重複していないノードの割合を求めることによって、どれだけ重複を避けることができたかを評価する。

3. 関連研究

本節では、まず本研究で扱う問題に関連のある既存の問題を取り上げ、その差異について述べる。その後、情報収集のための無人航空機の航空経路生成という点において関連した研究について整理し、それぞれの差異について述べる。

本研究で扱う問題に近いものとして、配送計画問題 (Vehicle Routing Problem : VRP) [3] の中でも時間枠付き運搬経路問題 (Vehicle Routing Problem with Time Windows : VRPTW)、および制約条件を緩めた Vehicle Routing Problem with Soft Time Windows (以下、VRPSTW) がある。VRP は、特定の施設に待機する運搬車が全ての顧客を巡回し、戻ってくる上で距離の総和が最短のものを求める問題である。VRPTW は、加えて各顧客に Time Window (訪れることが可能な時間帯) が設定されており、その時間帯に必ず訪れなければならない。VRPSTW は、設定された Time Window に遅れてもよいが、その分のペナルティを課すように VRPTW の制約条件を緩めた問題である。これらの問題は NP 困難であること [2] が知られており、これまでにヒューリスティクスによる解法の研究 [4–7] が多くなされている。

これらの問題と本研究で扱う問題の差異に、ノードを訪れる回数がある。VRPTW と VRPSTW において、運搬車は全てノードを丁度 1 回訪れる。一方、今回は前述のようなグラフを作成したため、本研究で扱う問題では同じノードを複数回通過することを許す。

情報収集のための無人航空機の航空経路生成については多くの研究がおこなわれているものの、緊急度の高い箇所を優先して回りつつ、網羅性も考慮した経路生成に関する研究は、我々の知る限り初めてである。他の各手法について、次から述べる。

無人航空機の航空経路生成の代表的な手法としてまず、HAWK [10] が挙げられる。この手法では、要救助者を発見するための経路生成を行うが、このとき経路生成に空間充填曲線であるムーア曲線を用いることで、位置特定の精度への対応と網羅性の確保を行っている。生成される経路は、要救助者がいる可能性の低い箇所は大雑把に、高い箇所は細かく回るような経路となる。対して、本研究では重要な箇所を細かく回るのではなく、先に回るような経路を生成する。

Liu らの研究 [9] では、情報収集を行う領域を 2D-tree で分割し、分割された場所の中で重要度が高い箇所を多く通る経路を生成する手法を提案している。この手法は、情報収集を行う領域に対してあらかじめ特定のパターンで重要度が設定されているという前提がある。この前提の上で、重要度の高いノードを多く持つ区域を複数回通過するような経路を生成する手法を提案している。対して、本研究では、重要な場所を何回も通るような経路ではなく、重要な場所を早く通るような経路の生成を行う。また、本手法では情報収集を行う領域に対しての上記のような前提は持たない。

Suijit らの研究 [8] では、情報収集を行う領域に対し優先度を考慮した経路設定を行う方法を提案している。この手法においても、あらかじめ情報収集を行う領域に重み付けを行う。こ

の重み付けされた場所同士の隣接関係をグラフとみなして経路制御を行う。この手法では、優先度の高い箇所を多く含む経路を生成しているが、情報収集を行う領域を網羅的に回るわけではない。対して、本研究では情報収集を行う領域の優先度のみでなく網羅性も考慮した経路生成を行う。

4. 提案手法

本節ではまず、提案手法の概要を述べる。次に、利用するA*探索アルゴリズムの説明を行う。その後、本手法におけるヒューリスティクスについて述べる。最後に、提案手法のアルゴリズムの詳細について述べる。

4.1 概要

本手法は、緊急度の高い箇所から順番に回りつつも重複が少なくなるような経路を生成する。まず、最も緊急度の高い箇所を目標地点として、現在地点からの経路探索を行う。次に、その目標地点の次に緊急度が高い箇所を新たな目標地点として設定し、さらに経路探索を行う。これを通過していない箇所がなくなるまで繰り返すのが、本手法の主な流れとなる。

ここでもし、同じ大きさの緊急度をもつ地点が複数あった場合は、目標地点は次の手順で決める。

- (1) 現在地点からの距離が短い箇所を目標地点とする
 - (2) 現在地点からの距離が同じだった場合、次に大きい緊急度を持つ場所への距離が遠いもの为目标地点とする
 - (3) 次の緊急度を持つ場所への距離が同じだった場合、経路 r を計算し、評価関数 $e_1(r)$ が返すの値が大きい方を目標地点とする
- 例えば、現在地点のノード v_c と次のような条件を満たすノード v_1, v_2, v_3 があるとする。

$$w(v_1) = w(v_2) \quad (3)$$

$$w(v_1) > w(v_3) > w(v_j) \quad (j \neq 1, 2, 3) \quad (4)$$

このとき、目標地点は次の手順で決定する。ここで、 r_1, r_2 は現在地点から v_1, v_2 へのそれぞれの経路、 $d(v_c, v_t)$ はノード v_c, v_t 間の距離を返す関数とする。

- (1) $d(v_c, v_1)$ と $d(v_c, v_2)$ を比較し、小さい方を優先
- (2) $d(v_c, v_1) = d(v_c, v_2)$ であれば、 $d(v_1, v_3)$ と $d(v_2, v_3)$ を比較し、大きい方を優先
- (3) $d(v_1, v_3) = d(v_2, v_3)$ であれば、 r_1, r_2 を求めた後、 $e_1(r_1)$ と $e_1(r_2)$ を比較し、大きい方を優先

本手法では、現在地点から目標地点への経路探索には最短経路探索アルゴリズムのひとつであるA*探索アルゴリズムを利用する。本手法では、A*探索アルゴリズムのコスト計算に重複をできるだけ避けるような指標を組み込むことで、効率的な経路を探索する。

4.2 A*探索アルゴリズム

A*探索アルゴリズムは、グラフにおける開始地点から目標地点までの最短経路を見つける問題において、スタートからの距離とゴールまでの距離の和の比較を繰り返すことで最短経路を探索するアルゴリズムである。

A*探索アルゴリズムの考え方は、次の通りである。まず、現在地点 v_c に対して開始地点からの距離を $g(v_c)$ 、目標地点までの距離を $h(v_c)$ として、これらの和を総距離 $f(v_c)$ とする。

$$f(v_c) = g(v_c) + h(v_c) \quad (5)$$

ここで距離をコストとして扱えば、最短経路を求める際には総コスト $f(v_c)$ が最小となり続けるような経路を選択すればよい。

しかし、開始地点からの距離 $g(v_c)$ とは違い、目標地点までの距離 $h(v_c)$ は既知ではない。そのため、目標地点までの距離の推定値 $h^*(v_c)$ を与えるものとする。この推定値 $h^*(v_c)$ はヒューリスティック関数と呼ばれる。総コストは $f^*(v_c)$ として次のように与えられる。

$$f^*(v_c) = g(v_c) + h^*(v_c) \quad (6)$$

この $f^*(v_c)$ の比較を繰り返すことで最短経路を探索するアルゴリズムがA*探索アルゴリズムである。

4.3 経路決定のヒューリスティクス

本手法におけるヒューリスティック関数 $h^*(v_c)$ は、次のように定義する。

$$h^*(v_c) = \alpha c_d + (1 - \alpha)(c_o + \hat{c}_o) \quad (7)$$

ここで、 c_d は現在地点から目標地点までの距離、 c_o はその経路をとることによってこれまでの経路と重複するノードの数、 $\hat{c}_o$ はその経路をとることによってさらにその後生じると推定される重複のノードの数とする。距離 c_d は経路がなるべく最短経路になるような値であるのに対して、重複ノード数 c_o と推定重複ノード数 $\hat{c}_o$ はできるだけ重複を避けるような経路をとるための値である。そのため、パラメータ α により、どちらを優先するかを重み付ける。

ヒューリスティック関数 $h^*(v_c)$ で利用するこれら3つの値について、次の(1)~(3)にて詳しく述べる。

(1) 距離 c_d

距離 c_d は目標地点までの最短経路をとるための値である。現在地点 v_c と目標地点 v_t の間の距離として次のように定義する。

$$c_d = d(v_c, v_t) \quad (8)$$

(2) 重複ノード数 c_o

本手法は、繰り返しA*探索アルゴリズムによって出力される道 (path) を繋げたものを経路としている。重複ノード数 c_o は、A*探索アルゴリズムによって新たに出力された道に含まれる、それまでの経路で既に通ったノードの数である。本手法では経路に重複を含むことを許すが、重複が多い経路は望ましくないため、この値を考慮する。既に通ったノードの列を経路 $R = [v_1, v_2, \dots, v_c]$ 、道を列 $P = [v_c, v_{c+1}, \dots, v_t]$ と置けば、重複ノード数 c_o は次のように表される。

$$c_o = |\text{set}(R) \cap \text{set}(P)| \quad (9)$$

(3) 推定重複ノード数 $\hat{c}_o$

推定重複ノード数 $\hat{c}_o$ は現時点では重複はしていないが、後々

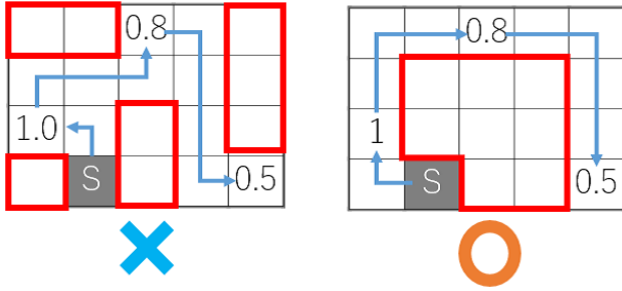


図 3: 未探索箇所のみとまり (右の経路と比べて左の経路は未探索箇所を回る際に既に回った箇所を複数通らなければならない)

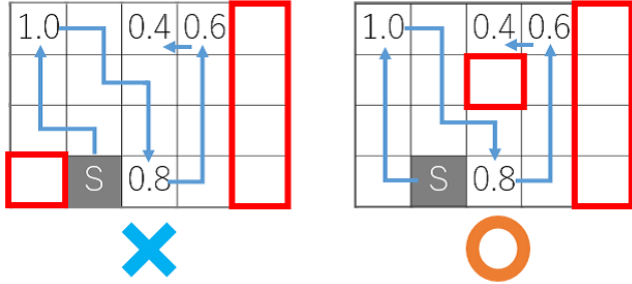


図 4: 未探索箇所同士の回りやすさ (右とくらべ左の経路は未探索箇所同士の距離が遠いために重複ノード数が増えてしまう)

の経路を考えると重複すると推定されるノードの数を表す。後々の重複を避けるために次の 3 つ指標を考える。

i. 未探索箇所のみとまり

重複を避けるために、未探索箇所のみとまりを評価する。後々の重複を避けるためには、未探索箇所ができるだけひとまとまりになることが望ましい。例えば、図 3 のように、未探索箇所をできるだけひとまとまりにするような経路をとることで後々の重複を避けることができる。

未探索箇所のみとまりを評価するために、未探索箇所の連結成分数を計算する。未探索箇所の連結成分数が増えた場合、少なくとも 1 つは既に通った頂点を通らなければ網羅的に探索することができない。

頂点の集合 V のうち、未探索である頂点の集合を U とする。未探索箇所のみとまりは、 U のうち 1-連結グラフが構成可能な頂点の集合を要素に持つグラフ T を指す。したがって未探索箇所のみとまりの数 q は、 U のうち 1-連結グラフが構成可能な頂点の集合を要素に持つグラフの個数となる。

ii. 未探索箇所同士の回りやすさ

重複をできるだけ少なくするためには、未探索箇所同士の距離が短い方が望ましい。図 4 から、未探索箇所同士の距離が後々の重複に影響することがわかる。そこで、未探索箇所のみとまりである U に含まれる 1-連結グラフ同士を分離している頂点の最小個数 s を評価する。

iii. 現在地点からの回りやすさ

未探索箇所がいくらまとまっても、目標地点から遠ざがるところに固まっていたり、重複が増えてしまう。そのため、目標地点 v_t と経路として通る箇所を除いた目標地点から最も

Algorithm 1 Efficient Route Generating Algorithm

Input: v_s, G, α

Output: R

```

1:  $v_c \leftarrow v_s$ 
2:  $U \leftarrow G.node$ 
3: while  $U \neq \text{empty}$  do
4:    $goal \leftarrow U.priority.max()$ 
5:    $path \leftarrow A\_star(v_c, goal, \alpha)$ 
6:    $R \leftarrow R + path$ 
7:    $Move(R)$ 
8:    $v_c \leftarrow goal$ 
9:    $U \leftarrow U - R.all$ 
10: end while

```

近い未探索箇所 v_r とのノード間距離が短い方が望ましい。目標地点 v_t からの回りやすさをノード間距離 $d(v_t, v_r)$ として評価する。

以上、i~iii までの指標を用いて、推定重複ノード数 $\hat{c}_o$ は次のように求める。

$$\hat{c}_o = \begin{cases} d(v_t, v_r) & q = 0 \\ s + d(v_t, v_r) & q \geq 1 \end{cases} \quad (10)$$

4.4 アルゴリズム

提案手法の疑似コードをアルゴリズム 1 に示す。

まず、入力として開始地点 v_s と、領域のグラフ G を与える。1, 2 行目では初期化として、現在地点 v_c に v_s を、未探索ノードの集合 U に $G.node$ を代入する。4 行目では、 $goal$ に優先順位が最大のセルを代入する。5 行目において、現在地点 v_c から $goal$ までの経路探索を A*探索アルゴリズムを用いて行い、経路 $path$ を得る。6 行目で、 R に $path$ を追加する。7, 8 行目で、経路にもとづいて移動を行い、 v_c を $goal$ に更新する。9 行目で、未探索ノードの集合 U から通過したノードを削除する。未探索ノードがなくなるまで、この 4~9 行目までの処理を繰り返す。

5. シミュレーションによる評価実験

本節では、まずシミュレーションを行う準備として使用するデータやパラメータなどの設定について述べる。その後、単純な手法と提案手法それぞれのシミュレーション結果を示し、評価を行う。

5.1 シミュレーションの設定

シミュレーションを行うにあたり、国土地理院が提供する平成 28 年熊本地震のデータ [11] を利用する。使用する写真を図 5 に示す。これは、平成 28 年熊本地震発災直後の熊本県阿蘇郡南阿蘇村立野の阿蘇大橋付近の空中写真である。まず、この空中写真から、被害の大きさに対応した値の割当てを人手で行い、図 6 のような重み付け領域を作成する。次に、この領域から、 5×9 の正方形格子状に並んだ頂点の集合 V と隣り合う頂点の四方をつなぐ辺 E 、頂点と重みを対応付ける関数 w からなるグラフ G を作成する。このグラフ G とスタートノード、パラメータ α を入力として与える。ここで、今回のシミュレーショ



図 5: 阿蘇大橋付近の空中写真

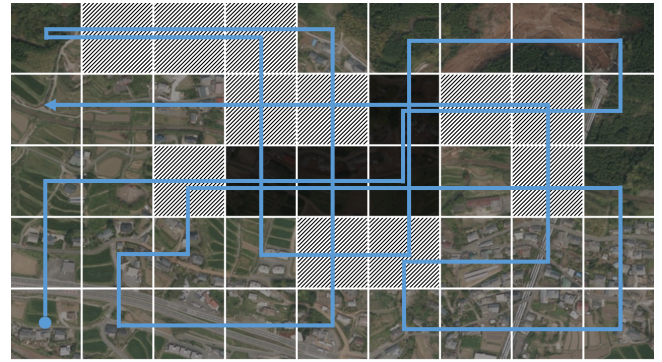


図 7: 単純な手法による生成経路

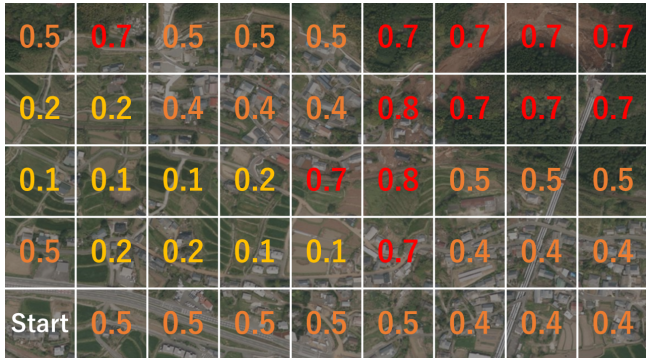


図 6: 空中写真の分割と重みづけ

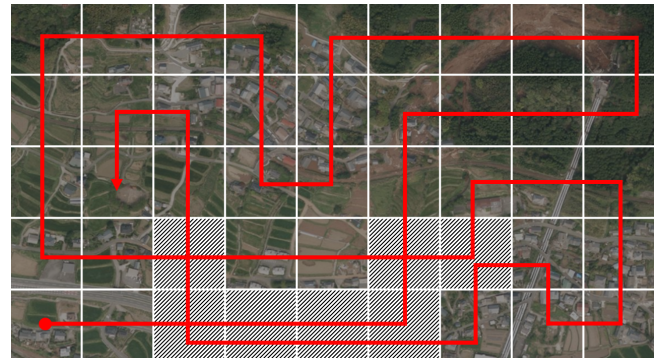


図 8: 提案手法による生成経路

表 1: 評価関数と経路長, 重複ノード数の比較

	単純な手法	提案手法
経路長	63	51
重複ノード数	19	7
評価関数 $e_1(R)$	2.13175345678535	2.58923906487727
評価関数 $e_2(R)$	0.6984126984126984	0.8627450980392157

ンではパラメータ $\alpha = 0.5$ とする.

5.2 シミュレーション結果

単純な手法と提案手法のシミュレーション結果として出力された経路をそれぞれ図 7, 図 8 に示す. これらの図において, 1 度重複して通ったセルを右上がりの斜線で網掛けし, 2 度重複して通ったセルを黒で塗りつぶしている. また, それぞれのシミュレーション結果の経路長と重みの関係を図 9 に, 評価関数と経路長, 重複ノード数の比較を表 1 に示す.

図 9 より, 単純な手法と比べて提案手法は緊急度の高い場所を優先して回りつつ, 経路長を短縮できていることが分かる. 表 1 からは, 単純な手法と比べて提案手法では重複ノード数を半分以上に抑え全体の経路長が 19%ほど短くなっていることが分かる. また, 提案手法の経路に対する 2 つの評価関数の値は両方とも単純な手法に対する評価関数の値を上回った. これらのことから, 単純な手法と比べたときの本手法の有効性が示唆された.

6. おわりに

本論文では, 自然災害における無人航空機の空撮を想定し, 緊急度と網羅性を考慮した効率的な空撮経路の生成を行う手法

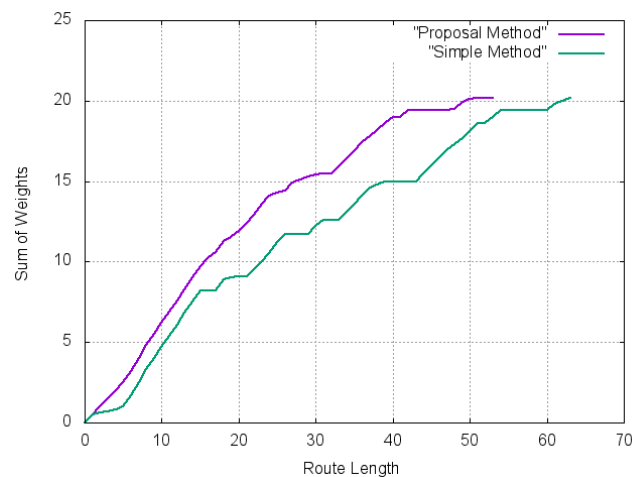


図 9: 経路長と重みの関係

を提案した. この手法では, 緊急度が高いものを優先しつつ, 後で残りの未探索箇所を回る際に重複した経路ををできるだけ通らなくて済むような経路を生成する. そのために, 経路探索に利用する A*探索アルゴリズムのヒューリスティック関数において, 現在地点から目標地点までの距離だけでなく, 経路中の重複ノード数と後々の経路を考えると重複することになるであろう推定重複ノード数を評価した.

加えて, 本手法と単純な手法との比較シミュレーションを行い, その結果, 提案手法は単純な手法と比べて, あるケースにおいては重複ノード数を半分以上削減し, 経路長を約 19%短くすることが確認できた. 今後は, 複数の無人航空機による空撮

シミュレーションおよび実験を行う予定である。

謝 辞

理論的側面に関する助言を頂いた筑波大学准教授鈴木伸崇先生に感謝します。本研究の一部は JST CREST, 科研費 (#25240012), 科学技術振興機構 SIP「レジリエントな防災・減災機能の強化」, および文部科学省「実社会ビックデータ利活用のためのデータ統合・解析技術の研究開発」の支援による。

文 献

- [1] Peter E. Hart, Nils J. Nilsson, Bertram Raphael. “A Formal Basis for the Heuristic Determination of Minimal Cost Paths”, IEEE Transactions on Systems Science and Cybernetics, 1968, Vol.4, No.2, p.100–107.
- [2] J. K. Lenstra, A.H.G. Rinnooy Kan. “Complexity of vehicle routing and scheduling problems”, Networks, 1981, Vol.11, p.221–227.
- [3] Dantzig, George Bernard; Ramser, John Hubert. “The Truck Dispatching Problem” Management Science, 1959, Vol.6, No.1 p.80–91.
- [4] Cordone Roberto, Roberto Wolfler Calvo. “A heuristic for the vehicle routing problem with time windows” Journal of Heuristics, 2001, Vol.7, No.2, p.107–129.
- [5] Martin Desrochers, Desrosiers Jacques, Solomon Marius. “A new optimization algorithm for the vehicle routing problem with time windows” Operations research, 1992, Vol.40, No.2, p.342–354.
- [6] Eric Taillard, Philippe Badeau, Michel Gendreau, Francois Guertin, Jean Yves Potvin. “A tabu search heuristic for the vehicle routing problem with soft time windows” Transportation science, 1997, Vol.31, No.2, p.170–186.
- [7] Nagraj Balakrishnan. “Simple heuristics for the vehicle routing problem with soft time windows” Journal of the Operational Research Society, 1993, Vol.44, No.3, p.279–287.
- [8] Sujit, P. B., and D. Ghose. “Search using multiple UAVs with flight time constraints”, IEEE Transactions on Aerospace and Electronic Systems, 2004, Vol. 40, No. 2, p.491–509.
- [9] Qilie Liu, Xiang Pang, Yong Wang, Yingjun Pan, Leilei Li. “An Improved Path Management Policy for the Ferry in Opportunistic Networks”, Journal of Networks, 2012, Vol. 7, No. 10, pp. 1568–1575.
- [10] Zhongli Liu, Yinjie Chen, Benyuan Liu, Chengyu Cao, Xinwen Fu. “HAWK: An Unmanned Mini Helicopter-based Aerial Wireless Kit for Localization”, IEEE Transactions on Mobile Computing, 2014, Vol.13, No.2, p.287–298.
- [11] 国土地理院, “平成 28 年熊本地震に関する情報”, <http://www.gsi.go.jp/BOUSAI/H27-kumamoto-earthquake-index.html>, accessed 2017–01–09.