Optimum Design for Drone Highway Network

Masatoshi Hamnanaka
Center for Advanced Intelligence Project
RIKEN
Tokyo, Japan
masatoshi.hamanaka@riken.jp

***Abstract*—This paper describes a design method for drone highway networks to eliminate the risk of conflict between drones and to improve the overall flight efficiency. Many flight path designing methods have been proposed; however, none of them addresses the issue of flight efficiency. We optimize each path using ant colony optimization and optimize the position of the terminal connecting the paths using particle swarm optimization. Experimental results show that the proposed method improves flight efficiency by 15.6% on average.**

Keywords— Air space map; path planning; ant colony optimization (ACO); particle swarm optimization (PSO)

I. Introduction

Multi-rotor unmanned aircraft systems (UASs) can fly and land in narrow places and can therefore be used for transporting things to mountainous areas where trees grow. However, in order for a drone to fly autonomously in the mountainous areas, the following three problems have to be resolved.

- Weak or lost GPS signal

  Position detection is extremely important for controlling the UAS and for preventing collisions. At present, most drones use the global positioning system (GPS) for position detection. However, GPS sometimes cannot indicate the correct position when signals from the satellites are shielded by buildings and/or mountains. Furthermore, GPS signals are very weak and hence are easily affected by a variety of interference [1, 2].

  We equipped the UAS with a compact, lightweight, two-dimensional LiDAR that was originally developed for automated driving of automobiles; we used it to acquire a two-dimensional surface cross-section (Fig. 1). Machine learning techniques such as those using support vector machines (SVMs) [7] do not work well for learning the relationship between a 2D surface vector and a label of the flight area because the amount of data obtained from a 3D map is enormous. SVMs would require exponentially increasing resources for constructing a model on such large datasets, which is infeasible.

  To resolve this issue, we used deep learning. Using our method based on deep learning we were able to estimate the flight area with 98.4% accuracy in a field experiment.

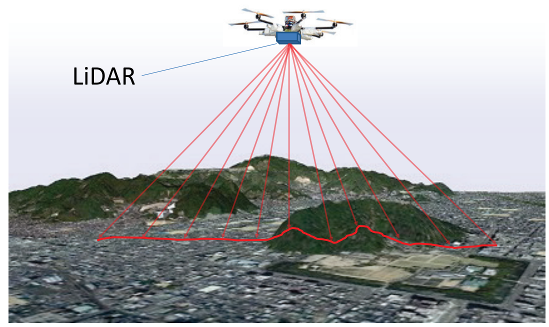


Fig. 1. Acquisition of 2D surface vector by 2D LiDAR.

- Increased collision risk

  With more drones flying in the sky, the risk of collision will increase and flying at high speed will be difficult. According to the Road Traffic Law, if two drones intersect in the air, the drone entering the intersection from the left takes precedence, so the drone entering from the right needs to wait while hovering and consuming energy.

  To resolve this issue, we proposed a drone highway network [6].

- Low energy efficiency

  Multi-rotor drones are energy-inefficient because they need to support their own weight by the rotation of the rotor. In mountainous areas, it is necessary to fly the drone at high altitudes, which requires high energy consumption.

  For the drone highway network, an optimum design method was proposed to combine the ant colony optimization (ACO) and the particle swarm optimization (PSO) for achieving the best energy-efficient route [6]. However, as optimizations require high calculation costs, it was not possible to create a fully optimized route.

In this paper, we report the results of a large-scale simulation by implementing ACO calculation on general-purpose computing on graphics processing units (GPGPU). Section II of this paper discusses related work; Section III gives an outline of the proposed method; Sections 4 and 5 describe ACO and PSO, and Section VI explains how we evaluate the proposed method. We conclude with a brief summary and future plans in Section VII.

II. Related Works

Various methods for creating flight paths for drones have been proposed. Li et al. tried to optimize the route between two points using ACO [7]. Because ACO takes time to converge, Bhatt et al. proposed a Sniff-Dog algorithm that converges at high speed [8]. Shi et al. formulated the problem of creating a flight path as a constraint satisfaction problem and found the shortest path without collision on the dynamic graph [9]. These studies optimize one route from the starting point to the arrival point [7-9].

On the other hand, the works of Balachandran et al. [10] and Galea et al. [11] are close to our approach on the premise that they controls multiple drones. Balachandran et al. proposed a method in which one drone selected from a myriad of drones approaching an intersection calculates the schedule of all the drones in the vicinity and synchronizes all the drones with information on that schedule. Galea et al. proposed a method to create a route network using Voronoi diagram. However, in these studies, no suggestions for improving the efficiency of the entire route network were proposed.

We aim to improve flight efficiency of the whole route in the network by proposing an algorithm that combines ACO and PSO.

III. Drone Highway Network Optimization

In this section, we present an outline of a method for constructing a drone highway network that enables drones to fly far and fast with high energy efficiency.

Figure 1 is a map of Kyoto City in Japan; the dark color belt (green) extending from the top to the bottom is a mountain at the prefectural border of Shiga prefecture and Kyoto prefecture. When the lattice in Fig. 1 is adopted as a drone highway network, the drones can fly in various directions. However, because of the route passing near the summit of the mountain, it takes a lot of energy to get over the mountain, resulting in poor energy efficiency.

In order for the drones to fly at high speed with high energy efficiency, it is important that 1) the distance is short, 2) the altitude change is small, and 3) the rute is straight. In this research, we introduce a combination of two techniques, ACO and PSO, as a method of searching for routes that simultaneously meet these three conditions.

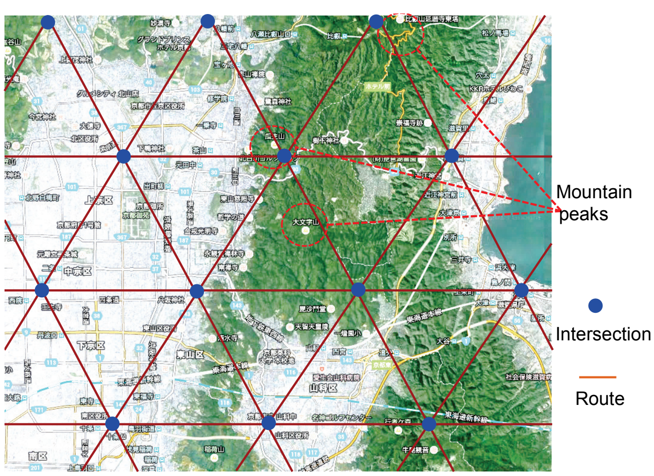


Fig. 2. Example of drone highway by lattice.

A. Ant colony optimization (ACO)

Optimization of each route of the drone highway network is performed by ACO. ACO is a stochastic model of ants efficiently moving from a colony to feeding stations using pheromones. An ant goes to pick up food from the colony. When the ant finds a bait, he brings back a part of it, releasing pheromones at that time. When other ants searching for food find pheromones, they move their paths in search of food ahead of them and strengthen pheromones on their way back. Since pheromones evaporate over time, the more suitable route is selected.

B. Particle swarm optimization (PSO)

Optimization of colony position is performed by PSO. PSO is intended for simulating social behaviour, as a stylized representation of the movement of organisms in a flock of birds or a school of fish. When trying to find the optimal solution in the multidimensional space, the particle group having the position and speed moves around. Each particle's movement is influenced by its local best known position. It is also guided toward the best known positions in the search space, which are updated as better positions are found by other particles.

C. Drone highway network optimization algorithm

We optimize the flight route network by combining ACO and PSO. The algorithm is as follows (Fig. 3).

a) Select one intersection (terminal) of the drone highway network at random. Set the terminal as a colony and put the bait on the surrounding terminals (Fig. 3a).

b) Calculate the optimal route from each colony to the bait by ACO and calculate their average movement costs (Fig 3b).

c) Place colony in various positions and use PSO to find a colony position with the lowest average movement cost (Fig. 3c).

d) Optimize the entire drone highway network by repeating steps (a) to (c) as one step (Fig. 3d).

IV. ACO for Routes Optimization

ACO is known as a method for efficiently obtaining a good approximate solution for a traveling salesman problem (TSP). TSP is a combination optimal NP hard problem that finds the lowest total cost (time and distance) of the tour among the routes that pass through all cities only once and return to the first city.

Because the start point and the goal point are decided, dynamic programming (DP) matching can be used to find the optimal on-the-ground route, so the optimum route can be efficiently searched by the Viterbi algorithm. However, it is difficult to optimize multi-rotor drone routes with DP matching because the energy efficiency of a drone flying in a three-dimensional path depends not only on distance and time but also on height change and curvature. We solve this problem by using ACO.

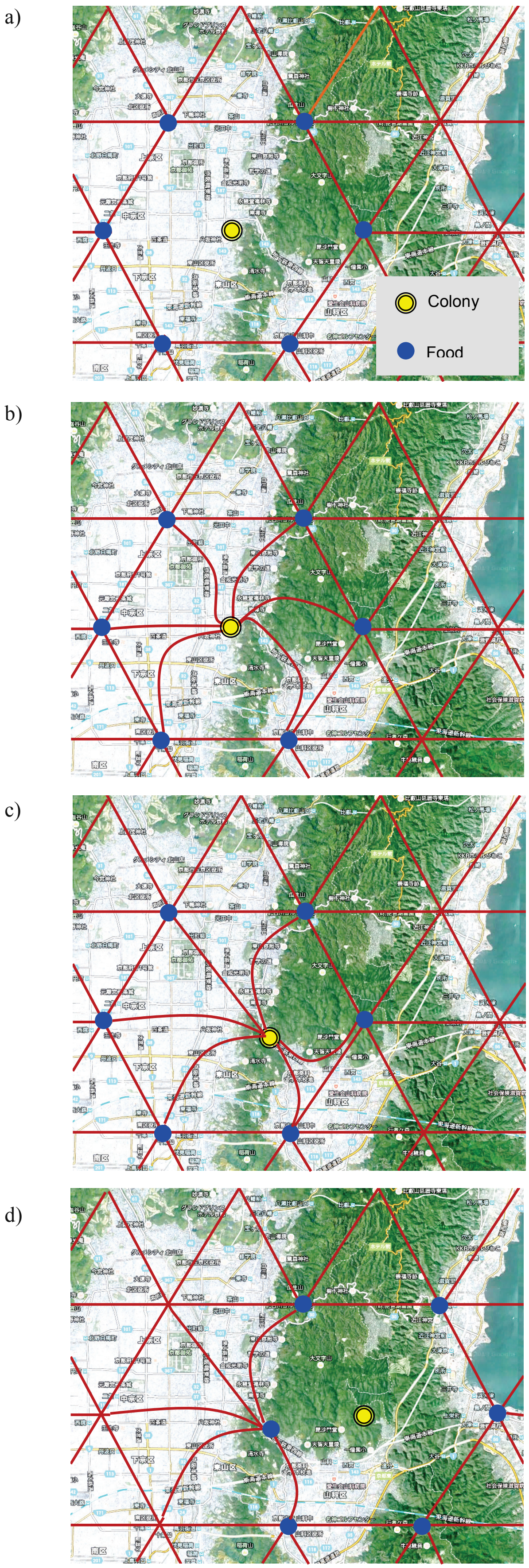


Fig. 3. Drone highway network optimization algorithm.

Whether the ant moves in one of the six directions of upward, downward, rightward, leftward, forward, and backward is determined by the ε-greedy method using the values of the pheromones in six directions.

The curvature is a reciprocal of the radius of the circumscribed circle of the triangle which is made up of the point where the ant is now, the point where the ant was 5 steps before, and the point where the ant will be after 5 steps.

When all the ants reach the goal or exceed the number of search termination steps, more pheromone is added to the route that the ant has passed. In order to shorten the search time by ants, we performed the following five procedures.

- Initial values of pheromone

  The pheromone was set as the initial value in a route which rises vertically from the start point and moves horizontally when reaching the highest point, so that the Euclidean distance to the goal is minimized. Setting this way converges very quickly compared to setting the pheromone's initial value to random. It is confirmed that the final convergence results are almost the same.

- Pheromone in each direction

  Pheromones were recorded and reflected in six directions of up and down, left and right, front and back, so that the ants following the pheromone did not flow back to the starting point.

- Search termination

  Ants are restricted in movement if there are walls nearby, so pheromone concentration increases along the walls. When there are ants searching for a place very far from the goal, the search time becomes too long, so we decided to terminate the search by squaring the shortest distance from the start to the goal. We also set the search terminate step number.

- Constraints on ants movement

  As there were many ants that were close to the goal but could not reach it, we decided to descend when ants flew over the goal. The depression can only move upward so that the ant does not descend. The ants did not go through same place again and did not make U-turns.

- Parallel processing with GPGPU

  In ACO, if the number of ants is small, the number of trials until convergence increases. Conversely, increasing the number of ants converges with fewer trials, but the trial time per one convergence increases. Therefore, we calculated ACO on GPGPU in parallel for each ant. As it can be done in parallel for speed up the ACO, except for calculation of pheromone update.

V. PSO FOR COLONY POSITION OPTIMIZATION

Particle swarm optimization (PSO) is a population-based stochastic optimization technique inspired by social behavior of bird flocking or fish schooling. In PSO method, the particles that represent potential solutions move around in the phase space with a velocity updated by the particle's own experience and the experience of the particle's neighbors or the experience of the whole swarm.

Although PSO is included in one of evolutional computation methods, convergence is better than genetic algorithm. However, since evaluation of solution candidates is repeated, it is difficult to increase the speed in the case of time-consuming processing for evaluating solution candidates. The evaluation value of PSO that we use is the average cost of the ACO. Since ACO takes time to converge, further speeding up is necessary. For speeding up, we did the following.

- Recording evaluation values

  In some cases, a particle once again obtains an evaluation value where a certain other particle has found an evaluation value before. Because the place for putting colonies is finite, record the evaluation value obtained first after searching for the colony position, and when the colony comes again to the same place, use the recorded evaluation value to speed up.

  The routes are obtained from the colonies to the surrounding six baits by using ACO. We speed up the calculation by performing six calculations in parallel. Parallelization techniques in a single ACO and six parallelizations of ACOs are different. The former is a software parallelization within one GPGPU, the latter is a hardware parallelization with six GPGPUs.

VI. Experimental Results

Simulation experiments are conducted using the drone highway network construction method that combines ACO and PSO.

A. Simulation area of ACO

We randomly select the start and goal points within a 200 square meter area from 36.6217 to 36.619900 of north latitude and from 139.1090 to 139.111240 of east longitude where we conducted the experiment before and find the optimal route. Figure 4 shows a three-dimensional topographic map of the area of 200 square meters with a resolution of 1 meter. The three-dimensional topographic map was obtained by flying 10 times and averaging it with DJI's Drone Matrice 600 equipped with SICK's LiDAR LD-MRS (Fig. 5). There is an altitude difference of about 90 meters between the deepest and the highest place. In the area of 200 square meters, an ant anticipates a goal point where there is food and proceeds. Ants advance one meter in one step up and down, right and left, forward and backward.

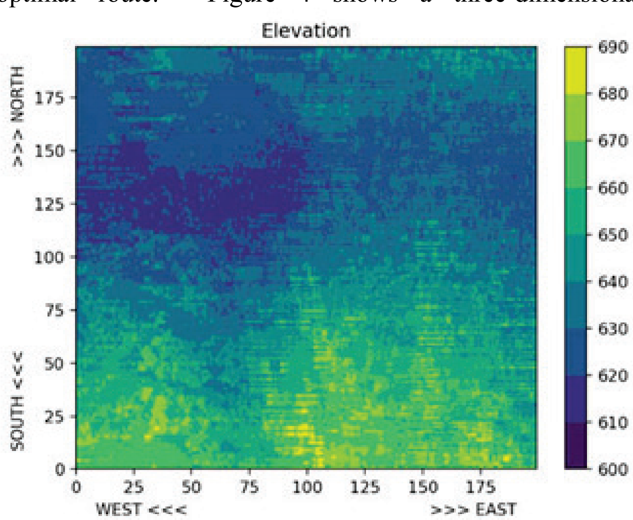


Fig. 4. Three-dimentional topographic map.

Fig. 5. *Matrice 600* equipped with *LD-MRS.*

B. Results of ACO

The following formula is defined as a cost function to be minimized when the cumulative distance moved by the ant is x meters, the cumulative height difference is y meters, and the average curvature of the route is z meters. The curvature is the reciprocal of the radius of the circumscribed circle of the triangle made up of the current position of the ant, the start point, and the goal point.

$$f = ax + by + cz \tag{1}$$

In ACO, there are many hyperparameters, which we set as follows.

- Number of ants MAX_ANTS 8192
- Constant of pheromone update Q 3
- Constant of dependence on pheromone ALPHA 1.4
- Constant of pheromone evaporation RHO 0.5
- Pheromone initial value IP 0
- Randomness of behavior EPSILON 0.05
- Maximum number of searches MAX_STEP 30,000
- Enforcement frequency MAX_Time 100
- Weight for distance a 0.08
- Weight for height b 0.8
- Weight for curvature c 0.01

As the ant population number MAX_ANTS increases, the degree of parallelism increases and converges at high speed, but it may cause instability due to insufficient memory and the like. 8192 is the upper limit that can operate stably in our environment when GPU is used. The pheromone update constant Q needs to be changed according to the distance from the start to the goal, but in this case 3 was appropriate. Randomness of behavior selection EPSILON acts randomly at a probability of 5% when 0.05.

As a result of the search, the minimum cost has decreased, and it has been confirmed that the optimization has advanced (Fig. 6). Fig. 7 shows the results of performing ACO with the white x as a start point and the red x as a goal point. The heat map is the amount of pheromone, and the shortest path is the white line. In the tenth trial (Fig. 7a), there is a meandering part, and the optimization has not progressed sufficiently, but it has advanced in the 100th trial (Fig. 7b).

The calculation speed was 1 hour 21 minutes 30 seconds in CPU (Mac mini Corei 7-8700 B 3.2 G / 4.6 GHz 6 core 12 threads) in 100 trials and 82.3 seconds in GPGPU (Nvidia GTX 1080).

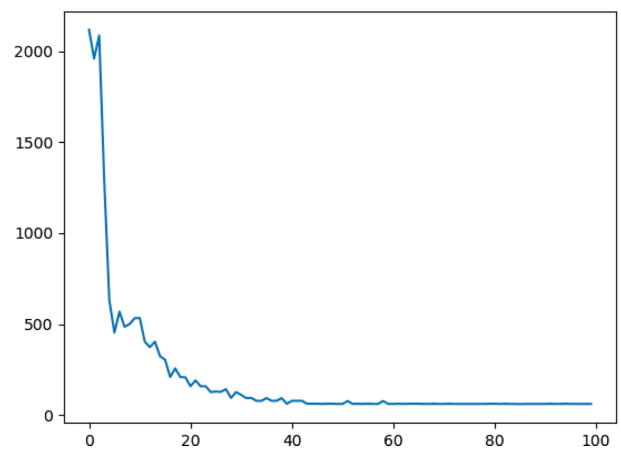


Fig. 6. Change of minimum cost by ACO

(a) 1 Epoch

(b) 100 Epochs

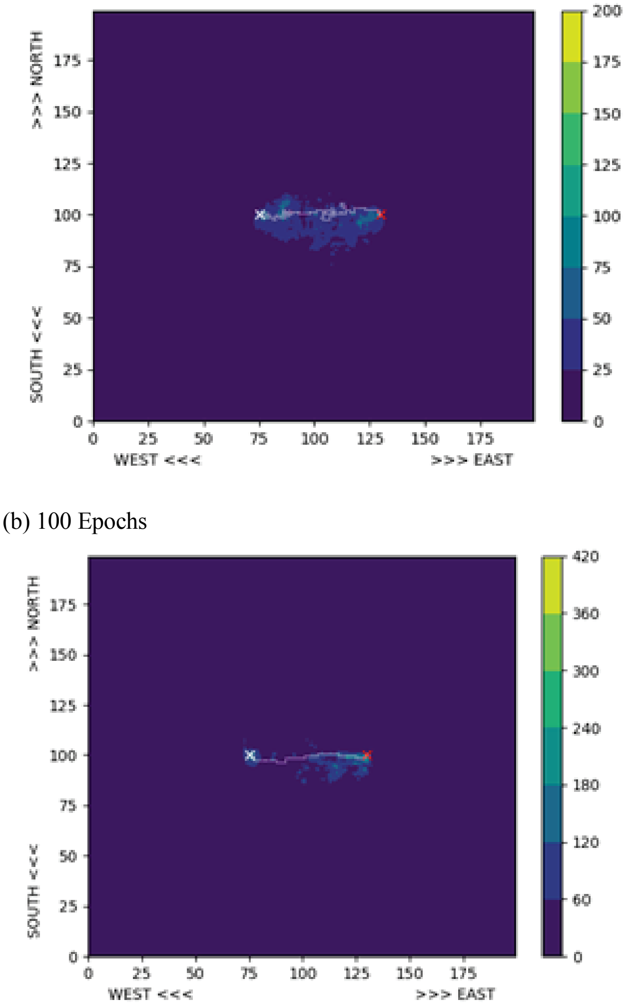


Fig. 7. Heat map of pheromone

C. Simulation area of PSO

We use a wide range of satellite 3D topographic maps from Advanced Land Observing Satellite (ALOS) for PSO. ALOS data can be downloaded with GeoTIFF. The range we used is an area of 2.5 Km x 3.0 Km, north latitude 34.3408-34.3986 degrees, east longitude 133.7561-133.7838 degrees. Altitudes are given for each mesh of about 25 m east to west and about 30 m north to south. The lengths of the east-west and north-south sides are different because ALOS's GeoTIFF is mercator projection. We optimized by PSO in this 100 × 100 mesh areas.

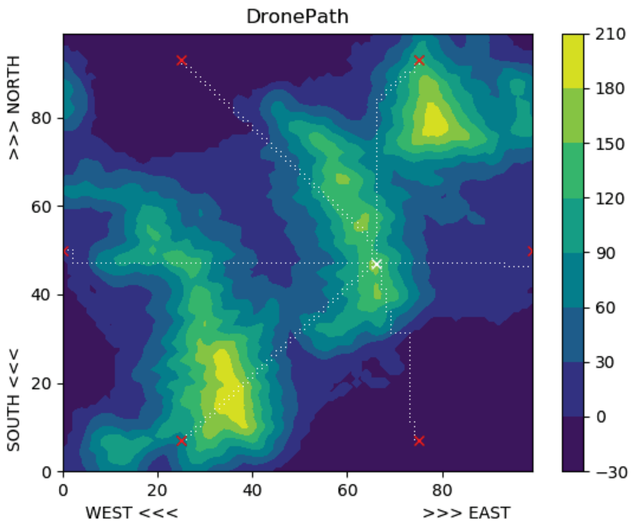
D. Results of PSO

In PSO, there are also many hyperparameters, which we set as follows.

- Size of Swarm SWARM_SIZE 30
- Inertia coefficient INERTIA 0.9
- Personal best of acceleration factor ACCEL_P 0.8
- Global best of acceleration factor ACCEL_G 0.8
- Minimum of standard partial partal refression coefficient COEF_MIN -1
- Maximum of standard partial partal refression coefficient COEF_MAX 1

Figure 8 shows the drone highway network obtained as a result of PSO with 50 steps and 100 steps. The white x is a start point, the red x is a goal point, and the white line represents the optimized network. The background heat map represents altitude. Each route is made to avoid high altitude places. Optimization reduced the average cost from the start point to all six goals by 15.6% (Fig. 9). The calculation time for the PSO was 12 hours.

(a) 50 Steps



(b) 100 steps

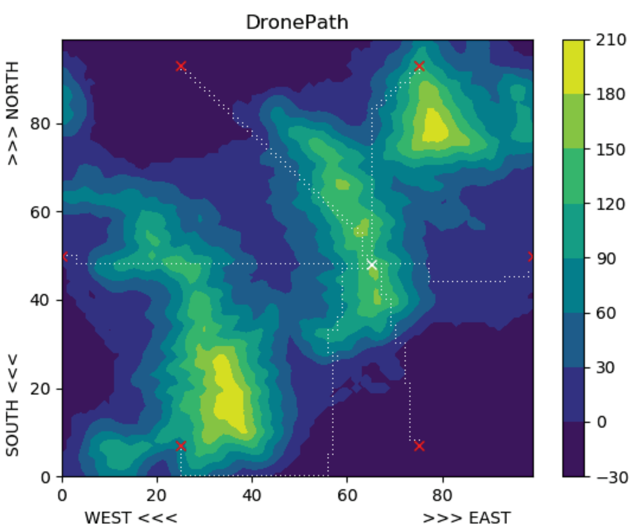


Fig. 8. Optimized drone highway network.

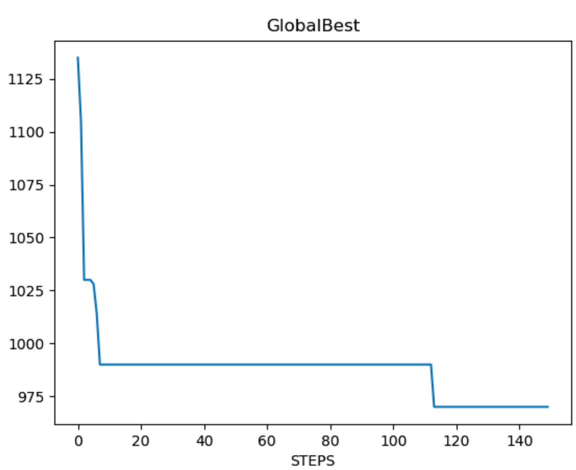


Fig. 9. Change of average cost by PSO

VII. Conclusion

We proposed a high efficiency drone highway network design method. We optimized each route using ACO and optimized the position of the intersection points of the route using PSO.

Software parallelization was performed on GPGPU to speed up ACO. In order to speed up PSO, hardware parallelization was performed using six GPGPUs. As a result of the optimization, the average cost decreased by 15.6%.

We plan to flight multi-rotor drones and verify flight efficiency. Figure shows results of a large-scale simulation. We plan to perform more large-scale simulation.

Acknowledgments

This work was supported by the Japan Society for the Promotion of Science (JSPS KAKENHI Grant Number 17K19972).

References

[1] Andrew J. Kerns, Daniel P. Shepard, Jahshan A. Bhatti, and Todd E. Humphreys, “Unmanned Aircraft Capture and Control Via GPS Spoofing,” Journal of Field Robotics, Vol. 31, Issue 4, pp. 617–636, 2014.

[2] David I. Urbina, Jairo A. Giraldo, Alvato A. Cardenas, Nils Ole Tippenhauer, Junia Valente, Mustafa Faisal, Justin Ruths, Richard Candell, and Henrik Sandberg, “Limiting the Impact of Stealthy Attacks on Industrial Control Systems,” Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 1092–1105, 2016.

[3] Yann LeCun, Yoshua Bengio, and Geoffrey E. Hinton, “Deep Learning,” Nature, Vol. 521, Issue 7553, pp. 436–444, 2015.

[4] Shin-ichi Amari, Tomoko Ozeki, Ryo Karakida, Yuki Yoshida, and Masato Okada, “Dynamics of Learning in MLP: Natural Gradient and Singularity Revisited,” Neural Comput, Vol. 30, Issue 1, pp. 1–33, 2018.

[5] Masatoshi Hamanaka, “Deep Learning-based Area Estimation for Unmanned Aircraft Systems using 3D Map,” Proceedings of 2018 International Conference on Unmanned Aircraft Systems (ICUAS2018), pp. 416–423, 2018.

[6] Masatoshi Hamanaka, Hideki Shiomi, “Optimum Design Method for Drone Flight Route Network, Proceedings of the 78th national convention of IPSJ, 1B-04, 2016 (in Japanese).

[7] Zhibin Li, Xiaojun Zhang, Yuing Lu, Yanxiang Cui, and Minghang Li, “Preliminary Research on Remote Planning Methods for Paths of Unmanned Aerial Vehicles in the Safe Aerial Corridor,” Proceedings of 2018 International Conference on Unmanned Aircraft Systems (ICUAS2018), pp. 1288–1294, 2018.

[8] Ashish Bhatt, Arnab Maity, and Kaushik Das, “Path Planning for UAV using Sniff-Dog-Algorithm,” Proceedings of 2018 International Conference on Unmanned Aircraft Systems (ICUAS2018), pp. 692–701, 2018.

[9] Ziji Shi and Wee Keong Ng, “A Collision-Free Path Planning Algorithm for Unmanned Aerial Vehicle Delivery,” Proceedings of 2018 International Conference on Unmanned Aircraft Systems (ICUAS2018), pp. 358–362, 2018.

[10] Swee Balachandran, Cesar Munoz, and Maria Consiglio, “Distributed Consensus to Enable Merging and Spacing of UAS in an Urban Environment,” Proceedings of 2018 International Conference on Unmanned Aircraft Systems (ICUAS2018), pp. 670–675, 2018.

[11] Marlon Galea, Brian Zammit, and Jason Gauci, “Design of a Multi-Layer UAV Path Planner for Cluttered Environments,” Proceedings of 2018 International Conference on Unmanned Aircraft Systems (ICUAS2018), pp. 914–923, 2018.

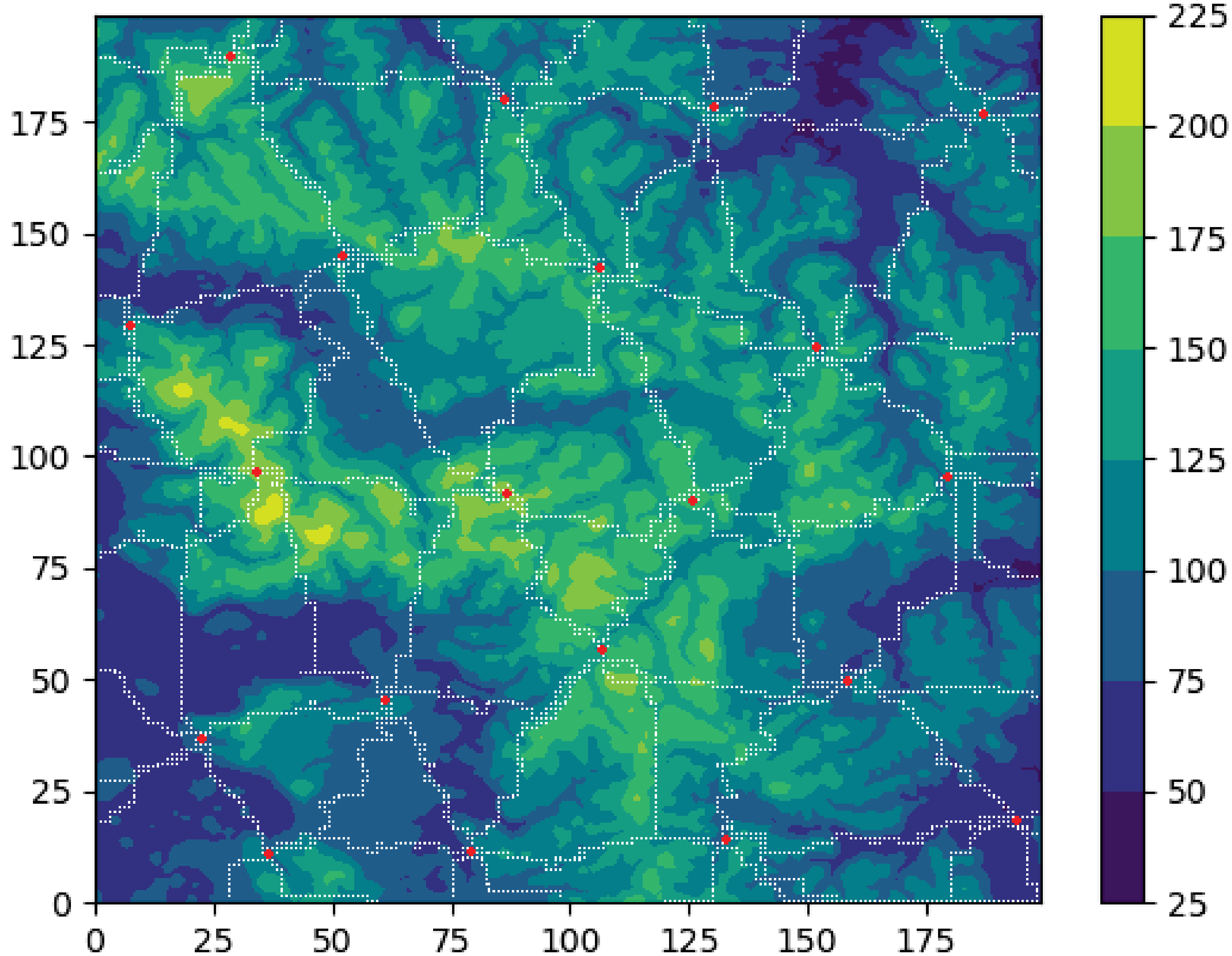


Fig. 10. Results of a large-scale simulation