Energy Minimization UAV Trajectory Design for Delay-Tolerant Emergency Communication

Y. A. Sambo, P. V. Klaine, J. P. B. Nadas, and M. A. Imran
School of Engineering, University of Glasgow,
Glasgow, Scotland. G12 8QQ.
Email: yusuf.sambo@glasgow.ac.uk

***Abstract*—The increasing cases of wireless communication networks being partly (or even fully) destroyed after the occurrence of natural disasters has made researchers focus on the use of Unmanned Aerial Vehicles (UAVs) to provide quick and efficient backup communication in post-disaster scenarios. However, the performance of UAVs in the provisioning of wireless coverage is known to be constrained by their battery life, which limits their flight times. In this paper, we explore the use of a single UAV to provide backhaul connectivity to truck-mounted Base Stations (BSs) that have been deployed within a disaster zone to provide network coverage to users based on the principle of delay-tolerant communications. We propose a trajectory design that uses genetic algorithm to find the trajectory with the least energy requirement for the UAV to visit all the BSs and return to a central node that acts as a gateway to the core network. Our trajectory design takes into account both the straight-and-level flight and banked-level turns of the UAV in computing the energy requirement. Simulation results show that our proposed design outperforms two approaches in the literature by up to 14% and 40%.**

***Index Terms*—UAV; trajectory; genetic algorithm; emergency communication networks; energy minimization; SON.**

I. Introduction

There has been tremendous improvement of the key performance indicators (KPIs) of wireless communication systems over the past 4 generations. However, despite all the progress made in the past years, the next generation of mobile networks, 5G, promises to stretch and improve these KPIs beyond the targets of current mobile networks [1]. One area that 5G is expected to improve over current cellular systems is in the realm of the quality of resilience, which deals with autonomous reconfiguration of wireless networks. For example, whenever a natural disaster occurs, leading to the destruction of wireless communication infrastructure, it is essential that 5G and beyond systems are able to rapidly adapt and autonomously reconfigure themselves in order to restore coverage as soon as possible [2].

In the aftermath of the 2011 Great Japan earthquake, over 29,000 BSs in the affected areas were shut down and up to 90% of voice calls were blocked in the few BSs that remained active [3]. When Hurricane Maria struck Puerto Rico in 2016, over 95% of the BSs in the island were destroyed, leaving millions of people unconnected [4]. As such, it is paramount to devise quick and efficient post-disaster deployment schemes to restore communication services.

UAV-aided communication is presently receiving a fair share of attention from researchers due to its low-cost, quick and flexible deployment [5]. The use of UAVs as flying BSs for coverage expansion and capacity improvement in terrestrial networks has been explored in [5–7]. The authors in [8] and [9] investigated the use of UAVs as mobile relays between two nodes by operating as data ferries. In [10], reinforcement learning was used to design a trajectory based on the shortest-tour technique for data gathering from ground sensor nodes. Moreover, the authors in [11] used Genetic Algorithm (GA) to design an energy conscious trajectory for UAV based mobile crowd sensing. In [12], the authors proposed an energy-efficient trajectory design whereby a UAV tries to maintain communication with a ground terminal. In terms of post-disaster communication, we used machine learning to determine the optimal positioning of distributed drone BSs, with the objective of maximizing the number of users served within a disaster zone in [13]. Furthermore, the works in [14] and [15] have shown that UAV-BSs can provide relief to users by improving coverage and the throughput of the network when some BSs have been cut-off. The authors in [16] proposed a UAV trajectory design for post-disaster scenarios based on latency and data volume constraints using GA for data delivery to user groups.

Despite the benefits of UAV-aided communication, especially in post-disaster settings [13–16], UAVs have a major drawback in terms of flight time, which is constrained by battery capacity [17]. This limits the flight time of battery-powered UAVs to under a couple of hours, which affects the performance of UAVs in post-disaster scenarios where access to electricity could be challenging. Hence, UAV trajectory design is a key performance criterion in post-disaster UAV-aided communications [17].

In this paper, we propose a quick, efficient and low-cost post-disaster wireless communication deployment whereby a truck-mounted BS is deployed to clusters of users within a disaster zone. In order to ensure quick and easy deployment of the network, it is assumed that the BSs do not have any form of backhaul capability to the core network and, as such, rely on a UAV to periodically come for a fly-by to receive the data from each BS and ferry it to the core network, or vice versa. The UAV starts from a gateway node (which denotes the physical location where data is extracted from the UAVs and transferred to the core network), flies to each BS to receive/transmit data and back to the gateway node to offload the data to the core network. This approach is most suitable for delay-tolerant communication whereby users send and receive text and multimedia messages, emails, update social media status, receive RSS feeds and emergency messages.

Although being simple by design, GA has been proven to find solutions to both complex and non-deterministic problems [18]. Hence, we propose a GA solution with the objective of finding a trajectory that minimizes the energy requirement of a UAV to get data from all the truck-mounted BSs. Note that this approach is a generalization of the well-known Traveling Salesperson Problem (TSP), which finds the route with the shortest distance for a salesperson to visit all the cities within a given set of cities once and return to the origin [19]. By modeling the total energy consumption of the UAV for any given trajectory, which takes into account both the energy requirements of straight line flight, as well as the energy required to change the heading of the UAV by making banked-level turns towards the next BS, we show that

This research was funded by EPSRC Global Challenges Research Fund - the DARE project - EP/P028764/1.

excluding the turn energy requirements significantly underestimates the UAV energy requirements. Hence, the scenario we consider differs from the traditional TSP given that the previous and future nodes visited, turn angle, influence the energy cost of a branch. Simulation results show that our proposed algorithm outperforms a trajectory algorithm based on [11] and [16] by up to 40% and 14%, respectively.

The remainder of this paper is organized as follows. Section II describes the system model. In Section III, we propose our GA-based energy minimization trajectory design algorithm. Section IV presents the performance analysis of the algorithm, while Section V concludes the paper.

II. System Model

Consider a post-disaster scenario in which the network is fully down and all communication links have been cutoff. We assume that N truck-mounted BSs are deployed to serve distinct areas where users might be clustered. Furthermore, the BSs rely on a UAV to periodically come for a fly-by to receive the data from each BS and transport it to the core network. One of the BSs is assumed to have a reliable connection to the Internet and, thus, serves as the gateway to the core network.

A. Flight Altitude

Assuming that all the BSs are on ground level, the maximum altitude that maximizes the communication link between the UAV and a truck-mounted BS, subject to a path-loss threshold constraint is given as [20]

$$h = r\tan(\vartheta), \tag{1}$$

where ϑ denotes the optimal elevation angle[1], and r represents the maximum coverage radius, which can be found from solving [20]

$$l = \frac{\eta_{\text{LoS}} - \eta_{\text{NLoS}}}{1 + \epsilon\exp\left(-\varsigma(\frac{180}{\pi}\vartheta - \epsilon)\right)} + 20\log\left(\frac{r}{\cos\vartheta}\right) + \beta. \tag{2}$$

In (2), l denotes the path-loss threshold, ϵ and ς are environment dependent constants, while η_{LoS} and η_{NLoS} represent the line-of-sight and non-line-of-sight losses that are added to the free space propagation loss, which also depend on the environment; finally, $\beta = 20\log(4\pi/\lambda) + \eta_{\text{NLoS}}$, where λ represents the carrier wavelength.

In this paper, we consider the N BSs to be randomly placed over an area, with a minimum separation of $2r$ between any two BSs and a UAV altitude of h.

B. UAV Energy Consumption Model

The energy consumption model of a UAV considers the energy requirements of straight-and-level flight and also that of changing the heading of the UAV through banked turns.

1) Straight-and-level flight: this refers to a flight that has a fixed heading and altitude. The following conditions hold for straight-and-level flights with constant speed [12]:

- Lift (L) is equal to its weight (W), i.e.,

$$L = W. \tag{3}$$

- Thrust (Γ) is equal to the drag (D), and is given as

$$\Gamma = D = \frac{1}{2}v^2 C_{D_0} S\rho(h) + \frac{2L^2}{\pi e_0 A_R S v^2 \rho(h)}, \tag{4}$$

where $v, C_{D_0}, S, \rho(h), e_0$ and A_R represent the UAV speed, zero lift drag coefficient, wing area, air density at altitude h, Oswald efficiency and aspect ratio of the wing, respectively[2].

Knowing that power is equal to the product of force and speed, the power for straight-and-level flight, P_s, is given as

$$\begin{aligned} P_s = \Gamma v &= \frac{1}{2}v^3 C_{D_0} S\rho(h) + \frac{2W^2}{\pi e_0 A_R S v\rho(h)}, \\ &= c_1 v^3 + c_2/v, \end{aligned} \tag{5}$$

where

$$c_1 \triangleq \frac{1}{2}C_{D_0} S\rho(h), \text{ and } c_2 \triangleq \frac{2W^2}{\pi e_0 A_R S\rho(h)}.$$

In (5), c_1 is known as the parasitic drag, which determines the power required to overcome friction drag and pressure on the UAV, while c_2 is known as the induced drag, which gives the power required to overcome lift-induced drag [12]. Given the relationships between c_1, c_2 and v, we can conclude that c_1 predominates at high speed, while c_2 predominates at low speed. Lastly, the energy required for straight-and-level flight is given as

$$E^s = t_s P_s = \left(c_1 v^3 + c_2/v\right) t_s, \tag{6}$$

where t_s is straight-and-level flight time.

2) Banked level turn: In order to change the heading of the UAV, the aircraft must bank at an angle so that the lift creates an acceleration component that is normal to the velocity of the UAV. The relationship between L and W for banked level turns is given as [21]

$$L = nW. \tag{7}$$

In (7), n denotes the load factor of the UAV and it can be expressed as

$$n = \sqrt{1 + \left(\frac{v\omega}{g}\right)^2}, \tag{8}$$

where ω and g denote the turn rate of the UAV and gravitational acceleration, respectively. For banked level turn, the relationship between Γ and D is given as

$$\Gamma = D + \hat{m}a, \tag{9}$$

where $\hat{m}$ and a denote the mass of the UAV and acceleration, respectively. By inserting (7) and (9) into (5), the power required for banked level turns can be expressed as

$$P_t = c_1 v^3 + c_2 n^2/v + \hat{m}av. \tag{10}$$

Hence, the energy required for a banked-level turn is given as

$$E^t = t_t P_t = \left(c_1 v^3 + \frac{c_2}{v}n^2 + \hat{m}av\right)t_t, \tag{11}$$

where $t_t = \phi/\omega$ is the time it takes for the UAV to complete the turn from point A to point B in Fig. 1, such that ϕ and ω denote the turn angle and turn rate, respectively. The radius of turn r_t, as shown in Fig. 1, can be found from [21]

$$r_t = \frac{v^2}{g\sqrt{n^2+1}}, \tag{12}$$

where g denotes gravitational acceleration.

III. Proposed Algorithm

In this Section, we formulate our problem and propose our trajectory design that minimizes the flight energy requirement of the UAV to visit all the BSs and fly back to the origin. Note

[1]The optimal elevation angle has been shown to be $42.44°$ for urban areas [20]

[2]Note that in this paper, we assume a zero wind speed.

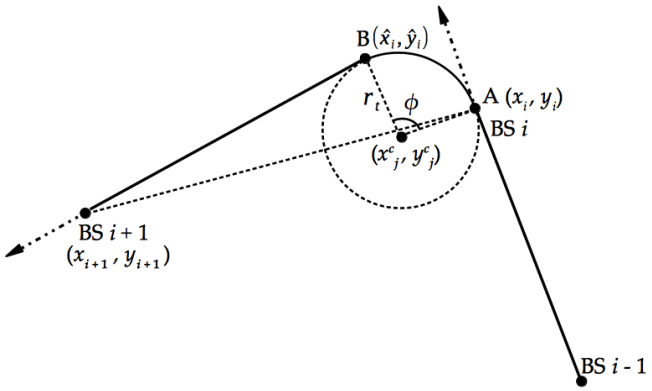

Figure 1. Turning performance of UAV.

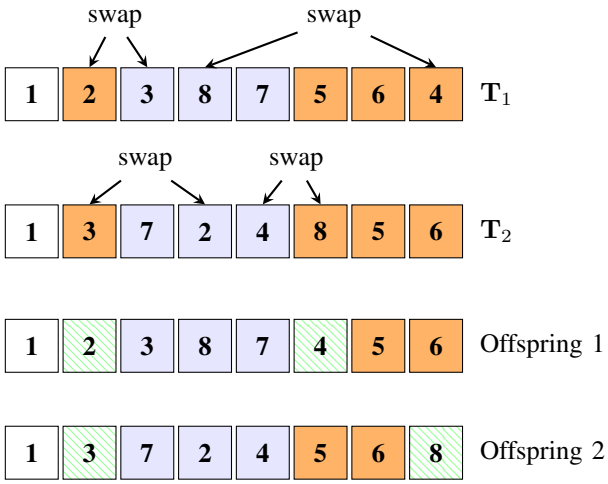

Figure 2. Partially-Matched crossover operation.

that this problem is proven to be combinatorial and NP-hard, hence, intractable to solve for a large number of BSs [16]. Accordingly, we adopt GA, an evolutionary type of algorithm that uses a small sample of possible combinations of the solution (termed *population*) and iteratively adapts them over Δ *generations* to find a good solution [22].

A. Fitness Function

We define the fitness value of any chromosome k to be the inverse of the total energy required for the UAV to travel from the gateway node to all the BSs and back to the gateway node, i.e., $\mathbf{T}_{k,1}$ through $\mathbf{T}_{k,2}, \ldots, \mathbf{T}_{k,N}$ and back to $\mathbf{T}_{k,1}$, for any trajectory $\mathbf{T}_k$, $\forall k = 1, 2, \ldots, K$, where K denotes the number of chromosomes. The fitness value, F_k, of chromosome k can be expressed as

$$\begin{aligned} F_k &= \frac{1}{\sum_{i=1}^{N}\left(E_i^s + E_i^t\right)} \\ &= \frac{1}{\sum_{i=1}^{N}\left(\frac{d_{k,i}}{v}P_s + \frac{\phi_{k,i}}{\omega}P_t\right)}. \end{aligned} \tag{13}$$

The parameter $d_{k,i}$ in (13) denotes the straight-and-level flight distance that the UAV travels between BS i and BS $i+1$, which is given as

$$d_{k,i} = \sqrt{(x_{k,i+1} - \hat{x}_{k,i})^2 + (y_{k,i+1} - \hat{y}_{k,i})^2}, \tag{14}$$

where $x_{k,i+1}$ and $y_{k,i+1}$ represent the x and y coordinates of the $(i+1)^{\text{th}}$ BS in $\mathbf{T}_k$, respectively; while $\hat{x}_{k,i}$ and $\hat{y}_{k,i}$ denote the coordinates of the UAV's position after the turn at BS i, represented as point B in Fig. 1.

The turn angle, $\phi_{k,i}$, can be found from

$$\phi_{k,i} = 2\sin^{-1}\left(\frac{\sqrt{(x_{k,i} - \hat{x}_{k,i})^2 + (y_{k,i} - \hat{y}_{k,i})^2}}{2r_t}\right), \tag{15}$$

where $\hat{x}_{k,i} = x^c_{k,i} - s_t r_t \sin(\psi)$ and $\hat{y}_{k,i} = y^c_{k,i} - s_t r_t \cos(\psi)$, such that $\psi = \alpha_{k,i} + s_t\theta_{k,i}$. Moreover, $\alpha_{k,i} = \tan^{-1}\left(\frac{y^c_{k,i} - y_{k,i+1}}{x^c_{k,i} - x_{k,i+1}}\right)$ and $\theta_{k,i} = \sin^{-1}\left(r_t / \left(\sqrt{(x_{k,i+1} - x^c_{k,i})^2 + (y_{k,i+1} - y^c_{k,i})^2}\right)\right)$, where, $x^c_{k,i}$ and $y^c_{k,i}$ denote the x and y coordinates of the center of turn the UAV makes, respectively, as shown in Fig. 1. Furthermore, the parameter s_t determines the turn direction such that $s_t \in \{-1, +1\}$, where $s_t = -1$ if the UAV is making a right turn or $+1$, otherwise.

B. Selection

Roulette wheel selection is used to choose the chromosomes that will undergo crossover, whereby the parents are chosen based on their fitness values [22] for mating to produce offsprings that would replace them in the next generation. Consider a roulette wheel containing all the chromosomes of the population and the pocket size of each chromosome is relative to its fitness value. This implies that if a ball is thrown in, the chromosomes with higher fitness values will have a better chance of being selected. We assume a fraction of the population, $U = \gamma_c \times K$, is selected in each generation to undergo crossover, where γ_c denotes the percentage of the chromosomes that will be selected.

We also perform *elitism*, which involves copying the best chromosome found in previous generations to the next generation so as to make the best solution survive to the end of the run [22].

C. Crossover

Given that the UAV can visit each BS only once in each trajectory and, as such, gene repetition in the chromosomes is not permitted, we use the Partially-Matched Crossover (PMX) approach [22] to create new offsprings from the selected parents. Assuming $\mathbf{T}_1$ and $\mathbf{T}_2$ are the selected parent chromosomes, two different crossing points that enclose 2 or more genes are randomly selected. This section is transferred from $\mathbf{T}_1$ to the same position in $\mathbf{T}_2$. In the event that there is a repetition on genes in the new chromosome, the duplicate genes of $\mathbf{T}_2$ that are not part of the transferred section are swapped with the genes of $\mathbf{T}_2$ that have been transferred to $\mathbf{T}_1$ to form the first offspring. The same approach is repeated to create the second offspring. Fig. 2 depicts how the offsprings are created using PMX.

After performing crossover, the fitness of the new offsprings is calculated and they only replace their parents in the next generation if they have a better fitness.

D. Mutation

In order to avoid falling into local optima, a portion of the chromosomes in a population undergo mutation [22]. In our algorithm, two entries of $\mathbf{T}_k$ with the highest gross energy requirements are swapped. Here, we define gross energy of BS i as the sum of the straight-and-level flight energy between

Algorithm 1 Energy Minimization Trajectory Algorithm

Inputs: $\Delta, N, K, \hat{\mathbf{T}}, v, \rho, S, C_{D_0}, W, A_R, e_0, \gamma_c, \gamma_m$ and $x_i, y_i \ \forall i = 1, 2 \ldots, N$
Initialize $\delta = 1$
While $\delta \leq \Delta$
  Roulette Wheel Selection;
  Compute fitness $F_k(\delta)$, $\forall k = 1, 2, \ldots, K$;
  Calculate cumulative sum of the fitness ratio of all chromosomes, $\mathcal{F}$
    For $i = 1, \ldots, U$
      Generate random number, m, from 0 to 100
      Go through $\mathcal{F}$ starting from 0, and select the chromosome at the point where $u_i \in \mathcal{F} > m$, to form $\mathcal{U}$
    End
  Crossover
  While $U > 0$
    Select 2 random chromosomes a and b from $\mathcal{U}$
    Choose 2 random crossing points and swap enclosing genes between chromosomes a and b
    Swap genes if there is repetition as shown in Fig. 2
    Denote new offsprings as $\tilde{a}$ and $\tilde{b}$ and compute their fitness
    Replace a and b by $\tilde{a}$ and $\tilde{b}$, if they have higher fitness
    $U = U - 2$
  End while
  Mutation
  For $k = 1, \ldots, K$
    Generate a random number, n_k
    If $n_k > \gamma_m$,
      Compute $G_i, \forall i = 1, \ldots, N$
      Swap the 2 genes with the highest G_i
    End if
  End
  $T^{\dagger}(\delta) = \max(F(\delta))$
  $\delta = \delta + 1$
End while
Output: $\mathbf{T}^{\star} = \max(T^{\dagger})$

BS i and the previous BS in the trajectory (BS $i-1$), the straight-and-level flight energy between BS i and the next BS in the trajectory (BS $i+1$) and the turn energy at BS i, which is defined as

$$G_i = E^s_{i-1} + E^t_i + E^s_i. \tag{16}$$

In order to determine which chromosomes are selected for mutation, we generate K random numbers between 0 and 1, one for each chromosome in the population. A chromosome is selected for mutation if its value is less than a given threshold, γ_m.

Note that the first entry of $\mathbf{T}_k$ is always untouched during mutation because the UAV always starts and terminates at the origin. The choice of using gross energy in our mutation is because it best defines the energy cost of visiting any given BS in the trajectory, with reference to the previous BS and the next one.

Our proposed GA approach is summarized in Algorithm I.

Table I
SIMULATION PARAMETERS [6, 20, 21, 23]

Parameter	Value	Parameter	Value
l (dB)	90	ϵ	11.95
η_{LoS} (dB)	1	ς	0.136
η_{NLoS} (dB)	20	λ (m)	0.15
c_1	9.26 $\times 10^{-4}$	c_2	2250
m (kg)	10	g (m/s^2)	9.81
γ_c (%)	80	γ_m	0.2

IV. NUMERICAL RESULTS

In this section, we compare the performance of our proposed algorithm against the algorithms of [11] and [16]. Instead of using the fitness function of [16] in our implementation of their algorithm, which takes into account data volume for each BS and latency, we use our fitness function. On the other hand, the fitness function of [11] is maintained as it is based on the straight-and-level energy requirement of the UAV during the flight. The crossover and mutation designs of the benchmark algorithms are maintained. In this paper, we employed 1000 runs of Monte Carlo simulation, whereby new BS locations are randomly placed over an area of 1600 m $\times$ 1600 m in each iteration and the results are averaged out. We assume that the first BS in each trajectory acts as the gateway and is located at the point (1,1) on the grid. The simulation parameters used in this paper have been summarized in Table I.

In Fig. 3, we compare the energy requirement of our trajectory design versus UAV speed, against the benchmark algorithms based on [11] and [16] for $N = 16$ BSs, $K = 520$ chromosomes and $\Delta = 180$ generations. It can be observed that as the UAV speed increases, the power requirement of all the algorithms decreases to a minimum speed of 30 m/s and then starts to increase again. This behavior is consistent with the power profile of aircrafts and it has to do with the power required to overcome the parasitic and lift-induced drag. It can be seen that our design outperforms the algorithms based on [11] and [16] by over 38% and 14%, respectively, at high UAV speed. This is due to the superiority of our GA, which performs crossover on a given percentage of the population in each generation and replaces the new offprings only if they have better fitness than their parents. Moreover, our algorithm also takes into account the energy cost of visiting each BS in the trajectory during mutation, compared to both benchmark algorithms. It is obvious that when the complete energy (CE) requirement is taken into account, whereby the turn energy requirement is also considered in the fitness function, the resultant energy requirement is lower than when only the energy straight-line flight is considered by about 1% to 7% as the UAV speed increases, as shown in the algorithm of [11] with CE. However, our algorithm still outperforms the algorithm of [11] with CE by over 30%. This confirms the superiority of our design over then algorithms of [11] and [16].

Fig. 4 shows the energy requirement of our trajectory design versus increasing number of BSs against the benchmark schemes for $v = 30$ m/s, $K = [250, 350, 430, 520, 650, 800]$ and $\Delta = [50, 100, 140, 180, 300, 450]$. Given that the search space of the trajectories increases with the number of BSs, K

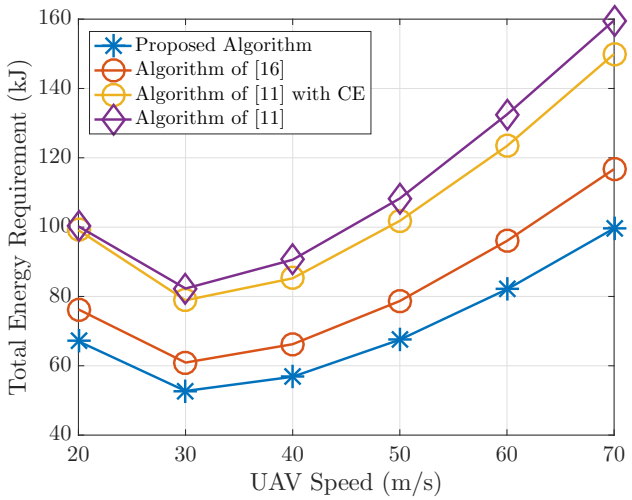

Figure 3. UAV energy requirement comparison of our proposed trajectory design against benchmark methods for increasing UAV speed.

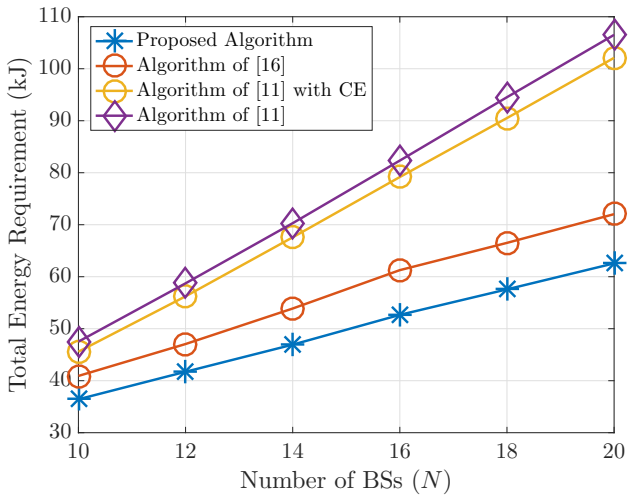

Figure 4. UAV energy requirement comparison of our proposed trajectory design against benchmark methods for increasing number of BSs.

and Δ in our proposed design and the benchmark algorithms have been set to vary according to the number of BSs. As expected, the energy requirement of all the algorithms increases with the number of BSs. This has to do with the increased flight time and the number of turns that the UAV has to make to get to all the BSs, as the number of BSs increases. It can be observed that the performance gap between our proposed design and the algorithm of [16] on one hand, and the algorithm of [11] on the other hand, increases with the number of BSs. This is due to our crossover selection criteria and our mutation design that is dependent on the gross-energy requirement of each BS, and the fact that the algorithm of [16] chosses the 2 fittest chromosomes in each iteration for crossover and mutation, and their offsprings replace the weakest ones in the next. This results in our algorithm outperforming the algorithm of [11] by between 23% to 40% at low and high number of BSs, while achieving up to 12% lower energy requirement when compared to the algorithm of [16]. Again, there is a difference of about 4% when the complete energy requirement is taken into account in the algorithm of [11].

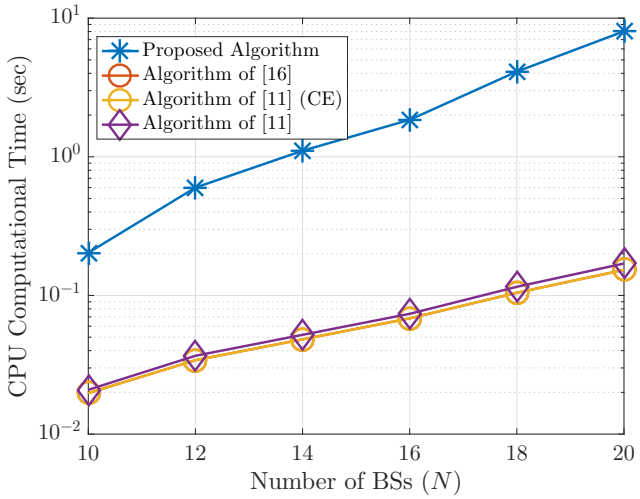

Figure 5. CPU computational time comparison of our proposed trajectory design against benchmark methods for increasing number of BSs.

In Fig. 5, we show the CPU computational time comparison of our proposed trajectory design against the benchmark schemes for increasing number of BSs. It is obvious that the CPU computational time of all algorithms increases with increasing number of BSs. It is evident that our trajectory design has a higher CPU computational time when compared to the benchmark algorithms. Our algorithm has about 10 to 50 times the computational complexity of the algorithms of [11] and [16] at low and high number of BSs. This is because γ_c and γ_m of the population undergoes crossover mutation, respectively, in each generation, whereas only 2 parent chromosomes undergo crossover and mutation in each generation in the benchmark algorithms. Moreover, our proposed mutation involves finding the BSs with the highest gross energy requirements and swapping their positions, against randomly choosing BSs to swap in the benchmark algorithms. This approach results in performance gains of up to 40% and 12%, when compared to the algorithms of [11] and [16], respectively, as shown in Fig 4. It is worthy of note that although our proposed trajectory algorithm has a higher complexity when compared to the benchmark algorithms, it is expected that optimization would be performed offline, which results in the computational time being less of an issue.

V. Conclusion

In this paper, we have proposed a GA-based trajectory design that minimizes the energy requirement of a UAV to travel to and receive data from BSs deployed around a disaster zone and return to a gateway BS to offload the data. The trajectory design takes into account both the straight-and-level flight and banked-level turns of the UAV in computing its energy requirement. Simulation results show that the proposed solution outperforms a trajectory algorithm based on [16] and Nearest Neighbor algorithm, with respect to total energy requirement. In the future, we plan to design an intelligent BS positioning scheme by using machine learning and also evaluate the energy efficiency of our trajectory design, by considering throughput and data transmission time at each BS in the trajectory.

References

[1] N. Cardona *et. al*, "Scientific Challenges Towards 5G Mobile Communications," COST IC1004 White paper, Tech. Rep., Dec. 2013.

[2] S. A. R. Naqvi, S. A. Hassan, H. Pervaiz, and Q. Ni, "Drone-Aided Communication as a Key Enabler for 5G and Resilient Public Safety

Networks," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 36–42, Jan. 2018.

[3] Ministry of Internal Affairs and Communications (Japan), "Maintaining Communications Capabilities during Major Natural Disasters and other Emergency Situations," Tech. Rep., 2011. [Online]. Available: http://www.soumu.go.jp/main_content/000146938.pdf

[4] D. Giel, "Puerto Rico, Ravaged by Hurricane Maria and Still Without Power, Receives Aid from NY Delegation," https://www.cnbc.com/2017/09/22/puerto-rico-still-without-power-receives-aid-from-ny-delegation.html, Sept. 2017. [Online]. Available: {https://www.cnbc.com/2017/09/22/puerto-rico-still-without-power-receives-aid-from-ny-delegation.html}

[5] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless Communications with Unmanned Aerial Vehicles: opportunities and challenges," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 36–42, May 2016.

[6] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Unmanned Aerial Vehicle With Underlaid Device-to-Device Communications: Performance and Tradeoffs," *IEEE Transactions on Wireless Communications*, vol. 15, no. 6, pp. 3949–3963, Jun. 2016.

[7] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP Altitude for Maximum Coverage," *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, Dec. 2014.

[8] Y. Zeng, R. Zhang, and T. J. Lim, "Throughput Maximization for UAV-Enabled Mobile Relaying Systems," *IEEE Transactions on Communications*, vol. 64, no. 12, pp. 4983–4996, Dec. 2016.

[9] J. Zhang, Y. Zeng, and R. Zhang, "Spectrum and Energy Efficiency Maximization in UAV-Enabled Mobile Relaying," in *2017 IEEE International Conference on Communications (ICC)*, Paris, May 2017, pp. 1–6.

[10] B. Pearre and T. X. Brown, "Model-free Trajectory Optimization for Wireless Data Ferries Among Multiple Sources," in *2010 IEEE Globecom Workshops*, Miami, FL, Dec. 2010, pp. 1793–1798.

[11] Z. Zhou, J. Feng, B. Gu, B. Ai, S. Mumtaz, J. Rodriguez, and M. Guizani, "When Mobile Crowd Sensing Meets UAV: Energy-Efficient Task Assignment and Route Planning," *IEEE Transactions on Communications*, vol. 66, no. 11, Nov. 2018.

[12] Y. Zeng and R. Zhang, "Energy-Efficient UAV Communication With Trajectory Optimization," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3747–3760, Jun. 2017.

[13] P. V. Klaine, J. P. B. Nadas, R. D. Souza, and M. A. Imran, "Distributed Drone Base Station Positioning for Emergency Cellular Networks Using Reinforcement Learning," *Cognitive Computation*, May 2018.

[14] A. Merwaday and I. Guvenc, "UAV Assisted Heterogeneous Networks for Public Safety Communications," in *2015 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, New Orleans, Mar. 2015, pp. 329–334.

[15] A. Kumbhar, I. Güvenç, S. Singh, and A. Tuncer, "Exploiting LTE-Advanced HetNets and FeICIC for UAV-Assisted Public Safety Communications," *IEEE Access*, vol. 6, pp. 783–796, 2018.

[16] K. Anazawa, P. Li, T. Miyazaki, and S. Guo, "Trajectory and Data Planning for Mobile Relay to Enable Efficient Internet Access after Disasters," in *2015 IEEE Global Communications Conference (GLOBECOM)*, San Diego, CA, Dec. 2015, pp. 1–6.

[17] S. A. R. Naqvi, S. A. Hassan, H. Pervaiz, and Q. Ni, "Drone-Aided Communication as a Key Enabler for 5G and Resilient Public Safety Networks," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 36–42, Jan. 2018.

[18] P. V. Klaine, M. A. Imran, O. Onireti, and R. D. Souza, "A Survey of Machine Learning Techniques Applied to Self-Organizing Cellular Networks," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2392–2431, Jul. 2017.

[19] E. W. Weisstein, "Traveling salesman problem." [Online]. Available: http://mathworld.wolfram.com/TravelingSalesmanProblem.html

[20] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3-D Placement of an Unmanned Aerial Vehicle Base Station (UAV-BS) for Energy-Efficient Maximal Coverage," *IEEE Wireless Communications Letters*, vol. 6, no. 4, pp. 434–437, Aug. 2017.

[21] J. N. Ostler, W. J. Bowman, D. O. Snyder, and T. W. McLain, "Performance Flight Testing of Small, Electric Powered Unmanned Aerial Vehicles," *International Journal of Micro Air Vehicles*, vol. 1, no. 3, pp. 155–171, 2009.

[22] S. N. Sivanandam and S. N. Deepa, *Introduction to Genetic Algorithms*. Springer, Berlin, Heidelberg, 2008.

[23] B. Wang, Z. Hou, Z. Liu, Q. Chen, and X. Zhu, "Preliminary Design of a Small Unmanned Battery Powered Tailsitter," *International Journal of Aerospace Engineering*, vol. 2016, pp. 1–11, 2016.