# Distributed Transmit Beamforming for UAV to Base Communications

**Yin Lu, Jun Fang, Zhong Guo and J. Andrew Zhang**

**Abstract:** Distributed transmit beamforming (DTB) is very efficient for extending the communication distance between a swarm of UAVs and the base, particularly when considering the constraints in weight and battery life for payloads on UAVs. In this paper, we review major function modules and potential solutions in realizing DTB in UAV systems, such as timing and carrier synchronization, phase drift tracking and compensation, and beamforming vector generation and updating. We then focus on beamforming vector generation and updating, and introduce a concatenated training scheme, together with a recursive channel estimation and updating algorithm. We also propose three approaches for tracking the variation of channels and updating the vectors. The effectiveness of these approaches is validated by simulation results.
**Keywords:** unmanned aerial vehicle; distributed transmit beamforming; beamforming vector generation and updating; channel prediction

## I. Introduction

We consider a situation where a swarm of UAVs work collaboratively on a task, in an area that is relatively far away from the base. For example, these UAVs are doing formation flight, or patrolling an area. These UAVs are connected to the base via wireless communication links, and form a communication network including these UAVs and the base. Considering the constraints in weight and battery life, we prefer communication systems with small profile, low weight and low power consumption. When the distances among UAVs are much smaller than the distance from UAVs to the base, distributed beamforming can be an excellent solution for achieving long-range and low-power communications. Since the main communication traffic is typically from UAVs to the base and the base can have much higher transmission power, we only consider the realization of distributed transmit beamforming (DTB) in this paper.

DTB [1] is a form of cooperative communication where two or more information sources simultaneously transmit a common message and control the phase (and power) of their transmissions so that the signals are constructively combined at an intended destination. Ideally, DTB with $N$ nodes/antennas can result in an $N$-fold gain in received power, for a given total transmitted power [2]. Hence using DTB, UAVs may significantly reduce the total transmission power, or extend their communication distance to the base.

DTB for conventional sensor networks has been well studied in [2] [3] [4] [5]. There are some specific problems for applying DTB to UAV networks, associated with the signal propagation environment, their movement and the geographical shape of the UAV swarms. There is very limited work on DTB for UAV networks that addresses these problems. An earlier work in this area was published in [6], where the author only reviewed the challenges and preliminary solutions, but provided little detail.

In this paper, we investigate the specific realization of DTB in a swarm of UAVs, where the group of UAVs transmit signals cooperatively to the base via forming distributed transmit beamforming. We review the major function modules and discuss potential solutions to implementing these modules, including timing and frequency synchronization, tracking phase drift, and beamforming vector generation and updating. We show that most of these modules, apart from beamforming vector generation and updating, can be efficiently implemented within UAVs, without requiring the involvement of the base. We then introduce a concatenated training scheme with scattered training symbols for estimating the channels between the base and UAVs and obtaining the DTB vector. In this scheme, UAVs send training sequences scattered over time, and the base estimates the channel, generate the DTB vector and feedback to UAVs. This scheme can efficiently combine discontinuous training symbols within one packet or across multiple packets for channel estimation. Hence training overhead can be significantly reduced to improve spectrum efficiency. We consider channel variation due to both UAV movement and the residual frequency offset, and propose three methods for updating beamforming in possibly fast varying channels. These schemes have varying complexity, and demonstrate different performance in simulation. They can be selectively adopted depending on channel varying speed.

## II. SIGNAL AND SYSTEM FORMULATION

There is no limitations on the formation and number of UAVs for technologies discussed in this paper. To provide concrete results, we consider an exemplified swarm of 16 UAVs, flying in formation (in two rows here). The moving speeds of the UAVs are up to 50 m/s, and the base is static. We consider a 2D geographical setup with horizontal distance and the height to the base only, represented as x- and y- axis respectively. The base is assumed to be at (x,y)=(0,0). The initial relative positions of these UAVs are in two rows, one row with 8 UAVs at y=200 m, and the other row also with 8 UAVs at y=250 m. The distance between two neighbouring UAVs in each row is 100 m. UAVs are travelling horizontally away from the base. The initial horizontal distance between the base and the nearest UAV is 5 km.

The carrier frequency used for wireless communication between UAVs and the base is assumed to be 900MHz, with a bandwidth of 5MHz. Thus the maximal Doppler frequency is 150 Hz. The packet length is assumed to be 200 samples, and hence the packet period is 0.04 ms. So over one packet, the maximum Doppler frequency can cause a phase shift of about 7 degrees, which is insignificant.

With DTB, the received baseband signal, in the absence of noise, at the base can be represented as

$$y(t) = \exp(j2\pi f_s\, t) \sum_n x_n(t) h_n(t) = s \sum_n a_n(t) \exp(j\phi_n(t))\, \mathrm{w_n(t)} \exp(j2\pi(f_s - f_{c,n})\, t),$$

where $n$ is the index of the UAVs, $f_s$ is the receiver's carrier frequency; $x_n(t) = s\, w_{\mathrm{n}}(\mathrm{t})\exp(j2\pi f_{c,n}\, t)$ is the signal transmitted from the n-th UAV, with the transmitted data symbol $s$, beamforming weight $w_n(t)$ and carrier frequency $f_{c,n}$; and $h_n(t) = a_n(t) \exp(j\phi_n(t))$ is the complex channel between the $n$-th UAV and the base with magnitude $a_n(t)$ and phase $\phi_n(t)$. Here, $\phi_n(t)$ has incorporated phase shifts caused by propagation delay, initial phase difference between transmitters, phase drift, and Doppler phase shift. Since line-of-sight multipath is dominating in the UAV–to-base connection, $a_n(t)$ mainly depends on the path loss and changes slowly. Therefore it can be assumed to be fixed for each UAV in this formulation, and the variable $t$ can be dropped. The term $\phi_n(t)$ contains both fast and slow time-varying components, and needs to be treated separately, as will be detailed later. In the above expression, we have assumed that the difference between signal arrival times is small enough so that no resolvable multipath signal is caused. This assumption is based on the fact that timing in DTB for UAVs is a less challenging problem and can be achieved with well-known technologies, such as through locking to the GPS timing.

From the above equation, we can see that in order to achieve a robust beamforming, it is necessary to synchronize $f_{c,n}$ for any $n$, estimate $\phi_n(t)$, and track and compensate for their variations over time. These are the main challenges in realizing DTB, in addition to other challenges such as information sharing between beamforming nodes. For more information on DTB, the readers are referred to [1] and [2] for overviews and [7] for MAC and routing design.

Observed over a period of $[k_0T_0, (k_0+K-1)T_0]$ with $K$ samples, the normalized mean DTB beamforming gain, normalized to the ideal one with perfectly known channels, is defined as

$$\gamma = \frac{\sum_{k=k_0}^{k_0+K-1} \left| \sum_n a_n \exp\left(j\phi_n(kT_0)\right) \mathrm{w_n}(kT_0) \exp\left(j2\pi\left(f_s - f_{c,n}\right) kT_0\right) \right|^2}{K a_n^2}.$$

## III. Main Function Modules and Solutions

Successfully implementing DTB in a swarm of UAVs relies on the following operations, as shown in Fig. 1 in the order of processing:

- Synchronize UAVs' transmission time so that their signals arrive at the base receiver approximately at the same time;
- Synchronize UAVs' carrier frequencies so that $(f_s - f_{c,n})$ becomes the same for all UAVs;
- Get rid of fast varying phase components in $\phi_n(t)$;
- Generate the beamforming vector based on the observed signals at the base, and feedback it to UAVs;
- Track the changes and repeat the above steps when necessary.

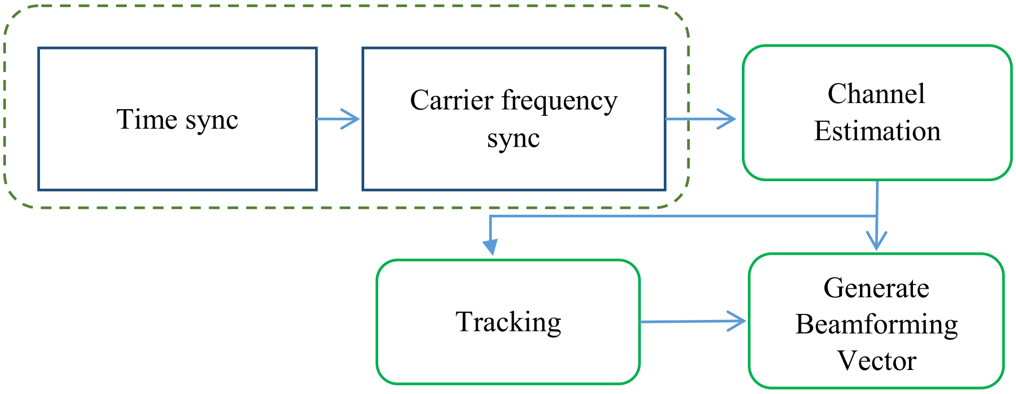


**Fig.1** *Major process in forming distributed transmit beamforming. Operations in square blocks and round-corner blocks are done in UAVs and the base, respectively.*

### 3.1 Time Synchronization between UAVs

Ideally, the arrival time of signals from different UAVs should be the same. However, this will require complex interaction between UAVs and the base. In DTB, we mainly concern large timing difference between UAVs that will lead to misalignment between the symbols from different UAVs, and cause large inter-symbol interference (ISI) at the receiver. Accumulated timing offset also needs to be compensated, as it will be translated as large ones. The beamforming vector can generally absorb small timing difference, which only cause some phase shift of the received signal.

The propagation time difference between UAVs and the base is typically small and hence is not a concern here. Since the distances between UAVs are much smaller than the distance between them and the base, their travel distances to the base only vary insignificantly. For example, when the UAVs are 5 kilometres away from the base, a relative distance of 50 metres between UAVs will only lead to a difference of about 1.25m in the propagation distance. For a data rate 10Mbps, this corresponds to about 5% of the bit period, which can be ignored. Hence once UAVs' transmission time is synchronized, we can assume that propagation delay causes little degradation to beamforming performance.

Time synchronization can thus be limited to be within UAVs, which is required to ensure that all of the cooperating UAVs start to transmit the same symbol at the same time. Being simplified as a conventional time synchronization problem in a network, various well-developed methods can be applied [8], for example, synchronizing to the GPS time is easy to achieve in UAVs.

## 3.2 Carrier Synchronization between Distributed UAVs

The more critical and challenging problem is carrier synchronization [9] [10]. The phases of the signals from different nodes may vary with time quickly and diversely if their carrier frequencies are different. We call them *carrier frequency offset* (CFO) here. Thus the beamforming gain will vary with time rapidly and randomly. Large CFO can result in complete failure of DTB and hence must be compensated. For phase shift caused by smaller CFO, it is shown in [4] that beamforming gains are quite robust to moderate errors in phase alignment. For example, 90 percent of an ideal two-antenna beamforming gain is attained even with phase offsets on the order of 30° [4]. The phase shift caused by small CFO can also be compensated by tracking its variation and updating the beamforming vector, as will be shown in Section IV. Thus practically, carrier synchronization will become solved if we can maintain the carrier frequencies' stability to several ppm (parts per million) and achieve similar synchronization accuracy.

Carrier synchronization can be implemented in either analog circuit or digitally. The core components of the analog circuit are phase looped lock (PLL) and a voltage controlled oscillator (VCO) [4] [8]. The unknown carrier frequency is generally accompanied by an unknown phase shift, which may be caused by propagation delay and the different phase response of hardware. The analog implementation requires phase offset to be known and compensated before correcting the frequency offset [4]. Estimation of the phase offset is a complex process, requiring generally closed-loop between transmit and receive nodes. The digital implementation can estimate the frequency offset and phase shift independently [8].

The digital implementation, resembling a digital PLL, is typically based on a maximum likelihood (ML) or maximum a posterior probability (MAP) formulation of parameter estimation [8]. The estimation of frequency offset is independent of the phase offset in the digital implementation. Hence we can ignore the phase offset during frequency offset estimation. Actually, the phase offset, which typically changes slowly, does not need to be estimated separately here as it can be incorporated to the channel estimation later. In this case, we can use a simple algorithm for frequency offset estimation, based on computing the one-lag autocorrelation of the baseband signal, as discussed in Chapter 8 in [8]. Assign any UAV as a master and let it transmit a beacon/training signal, other UAVs can implement this autocorrelation algorithm and work out their frequency offsets to the master UAV readily. The estimation can be done without involving the base.

After getting the carrier offset estimation, it can be either used to adjust the carrier frequency of the transmitter or inject a time-varying digital phase shift term to the beamforming weight.

## 3.3 Tracking Phase Drift during Beamforming

There are three types of phase drifts that may be of concern, caused by oscillator instability, residual frequency offset, and the movement of UAVs (Doppler frequency), respectively. The oscillator phase drift is random, and may represent an irreducible phase error if the stability period of the oscillator is too short. The last two generally change slowly, and can be compensated by updating beamforming vector, which will be considered in detail in Section 4.

Several studies have reported that the oscillator phase drift is generally not a significant issue for distributed beamforming design. In [11], using Brownian motion to model the oscillator drift, a Cramer-Rao bound is derived for the performance of estimating phase and frequency in the presence of the random phase drift. It is shown that estimation performance can be improved by increasing the number of observations, increasing the sampling frequency, and applying a Kalman filter [11] [12]. In [4], a statistical model is applied to analyze the effect of the oscillator phase drift on the beamforming gain. The results demonstrate that beamforming gain is robust to phase errors under some typical phase noise parameters. In average beamforming gains of at least 91% are achievable and 81% of the maximum for an extraordinary 35° phase drift is obtained.

### 3.4 Impact of UAV formation on Beamforming

The relative location of UAVs has some impact on the shape of beamforming. Although the "antenna geometry" of a distributed beamformer may be random, the beamformer pattern may be characterized statistically based on some statistical approximation of the geometrical distribution of UAVs. The probability distribution of the far-field beam pattern of a distributed beamformer with node locations uniformly distributed on a two-dimensional disk of radius *R* is analyzed in [13]. It is demonstrated that very narrow beamwidths can be achieved for a reasonable number of nodes. For example, a network with 10 randomly placed source nodes on a disk of radius 25 meters will, on average, achieve a 3dB beamwidth of less than half a degree if the sources transmit with 900MHz carriers. In [3], the beam pattern is analysed based on an assumption of Gaussian distributed instead of uniformly distributed nodes, and it is shown that Gaussian deployment gives wider mainlobe and has lower chance of large sidelobes.

Figures 2 and 3 present the beam pattern of the DTB formed by the 16 UAVs at their initial locations. The figures show clear main and sub lobes of the beam. The size of the mainlobe is geographically large in both horizontal and vertical domains.

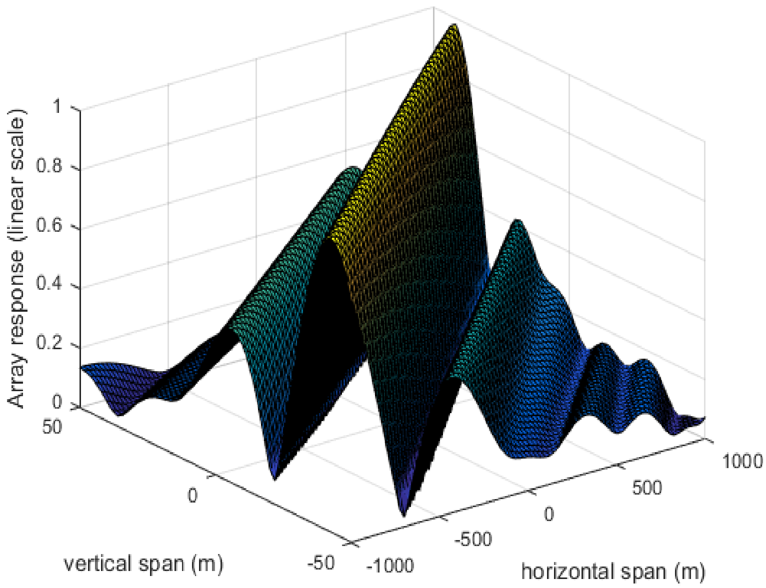


**Fig.2** *3D plot of the beamforming pattern seen around the base. The base locates at (horizontal span, vertical span) = (0,0).*

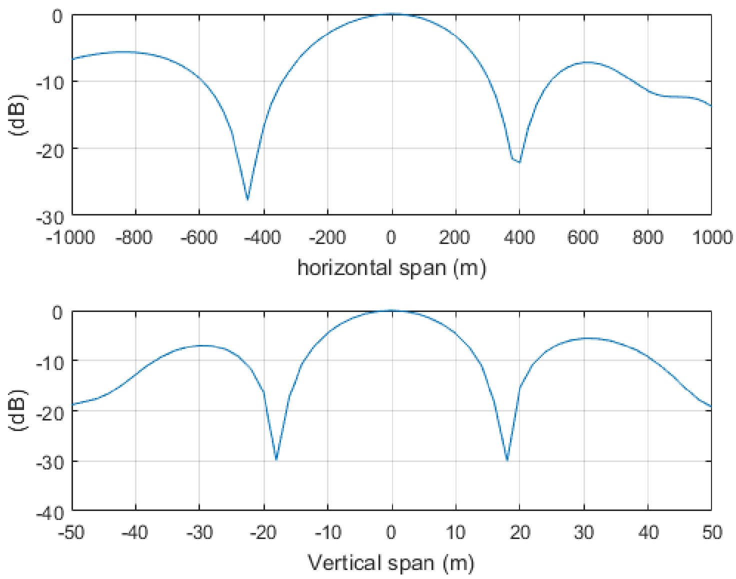


**Fig.3** *Sample 2D-plot of the beam pattern at vertical span=0 (top) and horizontal span=0 (bottom).*

## IV. Generation and Updating of Beamforming Vector

Once time and frequency synchronization are completed, beamforming vector can be determined through channel estimation at the base, and then feedback to UAVs. There are several known channel estimation and feedback schemes, such as the 1-bit feedback scheme [14], and nonfeedback scheme using spatial-temporal extraction [15]. The 1-bit feedback scheme is a close-loop implementation requiring least feedback, but it takes very long time to converge. The nonfeedback scheme does not require information feedback, but has high computational complexity and requires other information for processing, apart from a complex training scheme.

In this section, we introduce a concatenated training scheme for estimating the channels and obtaining the DTB vector, considering channel variation due to both UAV movement and the residual frequency offset.

In [16], we proposed a concatenated training scheme for channel estimation in DTB systems, and derived the optimal training signals, particularly for spatially correlated channels and for the case when the number of consecutive training symbols, $N$, are less than the number of distributed nodes, $M$. The scheme distributes a block of complete training signals to multiple sub-blocks, each being applied to one packet or a fraction of the packet. Training signals over different packets can be seamlessly combined for channel estimation. This scheme can be effectively applied to the UAV setup here. Using scattered training signals are helpful for tracking channel variation, and training overhead can be reduced to improve spectrum efficiency.

Fig. 4 shows an example of the frame structure, where the training signals are placed in the middle and the end of the packet to reduce the delay in applying the estimated DTB vector to the data signal in the next packet. These training symbols can be flexibly placed in many different forms, such as being divided into more and shorter sub-blocks, even as a single pilot, and uniformly scattered over the packet. The preamble in the beginning of a packet may contain additional training signals for estimating the combined channels that are weighted by the DTB vector.

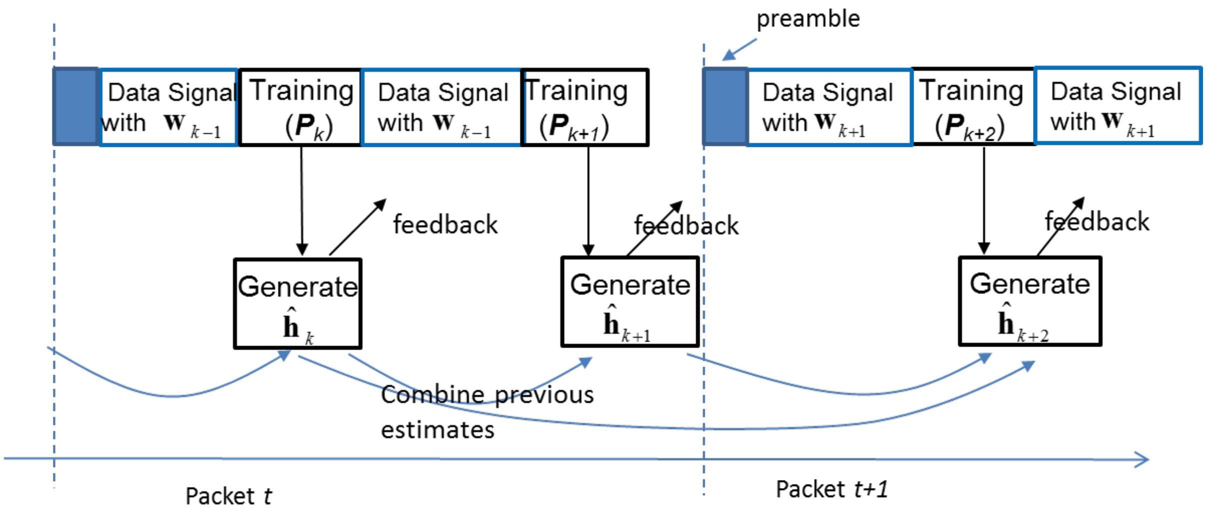


**Fig.4** *Frame structure and signal flow of DTB vector generation.*

Ignoring the correlation between the pathloss of the UAV channels, we can use an orthogonal matrix as the basis of constructing these training signals. Given an $M \times M$ matrix $\boldsymbol{T}$, we can generate $\boldsymbol{P}_k$ from $\boldsymbol{T}$ cyclically. For a $N_k \times M$ matrix $\boldsymbol{P}_k$, mathematically, the $q$-th row of $\boldsymbol{P}_k$ , $q \in [1, N_k]$ is the $(mod(q+p, M))$-th row of $\boldsymbol{T}$, where $p$ is the index of the next row in $\boldsymbol{T}$ after obtaining $\boldsymbol{P}_{k-1,}$ and $mod(x, y)$ denotes the operation of $x$ modulo $y$.

A recursive algorithm can then be conveniently applied to combine signals from these sub-blocks to get the channel estimation at the receiver. Typically, at least $K \geq M/N$ sub-blocks are combined to get the channel estimates, unless channels change too rapidly. Referring to (35) in [16], the recursive equation is given by

$$\widehat{\boldsymbol{h}}_k = \widehat{\boldsymbol{h}}_{k-1} + \boldsymbol{P}_k^H y_k - \boldsymbol{P}_{k-K}^H \boldsymbol{y}_{k-K} \,,$$

where $\boldsymbol{P}_k^H$ is the conjugate transpose of $\boldsymbol{P}_k$, $\boldsymbol{y}_k$ is the corresponding received signals, $K$ is the number of subblocks selected for combination, and $\widehat{\boldsymbol{h}}_k$ denotes the channel estimate obtained at the $k$-th subblock (but with previous signals combined). The DTB vector is then determined through $\widehat{\boldsymbol{h}}_k$, through e.g. $\boldsymbol{w}_k = \mathrm{conj}(\widehat{\boldsymbol{h}}_k)$, where $\mathrm{conj}(x)$ denotes the conjugate of $x$. Without noted otherwise, we will assume that $K \geq M/N$ is used in the following discussion and simulation.

When both Doppler frequency and residual CFO are small enough, the recursive equation (3) works quite reliably. But its performance will deteriorate significantly when channels change rapidly. We propose the following three improved methods to deal with the rapid channel variation. Their performance will be compared in Section 5 through simulation.

## 4.1 Method 1: Improved Recursion with K Adapting to Channel Variation

A simple way for improving the recursive algorithm above is to make $K$ adaptive to the channel variation. When channel is stable, a larger $K$ can lead to higher SNR and hence better channel estimation. However, if it changes rapidly, larger $K$ introduces more mismatch and degrades the estimation performance. Therefore, we want to find a right $K$ that adapts to the channel variation. Our approach to achieving this goal is computing the mean signal power over the data payload period, and comparing it to some thresholds. The selection of these thresholds, however, is not straightforward. UAV channels vary slowly in terms of the pathloss or channel magnitude due to the dominating line-of-sight propagation. This implies that the variation of the computed signal power from DTB will vary consistently with the variation of the channel phase. Therefore, we can ignore the magnitude variation, and focus on the varying channel phase. One approach is then comparing the received signal power obtained at the current sub-block to those at the previous (averaged) ones. If the signal power decreases, it could be an indicator that $K$ shall be decreased. However, this cannot tell us when we should increase $K$.

Our proposed novel approach is to compute and update the signal power for each value of $K$, and use them as benchmarking values for comparison with the signal power. More specifically we compare the signal power obtained at $K$ to the benchmarking values for $K-1$ and $K+1$, as they can tell us clearly what we can expect for different channel variations. Let $p_k(K)$ and $b_k(K)$ denote the measured signal power and the updated benchmarking value at $k$-th block for value $K$, respectively. If channel is stable, $p_k(K)$ is expected to be larger than $b_k(K-1)$ and smaller than $b_k(K+1)$; if channel becomes less stable, $p_k(K)$ is expected to be smaller than $b_k(K)$ and even $b_k(K-1)$. The algorithm is summarized below.

$$\begin{cases} \text{If } a * b_k(K-1) \leq \ p_k(K) \leq \dfrac{b_k(K+1)}{a}, \text{do } K = K+1, \text{and } \widehat{\boldsymbol{h}}_k = \widehat{\boldsymbol{h}}_{k-1} + \boldsymbol{P}_k^H y_k; \\ \text{Elseif } p_k(K) \leq \dfrac{b_k(K)}{a}, \text{do } K = K-1, \text{and } \widehat{\boldsymbol{h}}_k = \widehat{\boldsymbol{h}}_{k-1} + \boldsymbol{P}_k^H y_k - \boldsymbol{P}_{k-K}^H y_{k-K} - \boldsymbol{P}_{k-K-1}^H y_{k-K-1}; \\ \text{Otherwise } \widehat{\boldsymbol{h}}_k = \widehat{\boldsymbol{h}}_{k-1} + \boldsymbol{P}_k^H y_k - \boldsymbol{P}_{k-K}^H \boldsymbol{y}_{k-K}. \end{cases}$$

In the above algorithm, $a$ is a scalar and we have found through simulation that $a = 1.1$ is a good choice. The benchmark $b_k(K)$ is updated through a recursive equation $b_k(K) = \beta * b_k(K) + (1-\beta) * p_k(K)$, with the forgetting factor $\beta = 0.4$.

## 4.2 Method 2: Estimation of Phase Variation

The second method intends to estimate the phase variation for each UAV channel at the base receiver, and then feedback both the beamforming vector and phase variations to UAVs. Let $\widehat{\boldsymbol{h}}_k$ be the channel estimates obtained with $K = \frac{M}{N}$. The channel phase variations can be estimated as $\boldsymbol{\phi}_k = \angle(\widehat{\boldsymbol{h}}_k \odot \text{conj}(\widehat{\boldsymbol{h}}_{k-1}))$, where $\odot$ denotes element-wise multiplication. Different sub-blocks see different phase shifts, hence there is no a constant phase difference between any two channel estimates for any UAV. When the phase shift is relatively small, the estimation performance is acceptable as will be seen from the simulation results in Section 5. Once $\boldsymbol{\phi}_k$ is estimated, each UAV can generate a phase shifting sequence, multiplied to its signals to be transmitted, to compensate for its phase variation due to its movement and residual CFO.

## 4.3 Method 3: Tracking through Channel Prediction

A theoretically more rigorous approach is to apply a channel prediction algorithm to predict what the channel, or DTB vector, should be in the next packet period, using the current and past received signals at the base. We use a linear prediction [17] for the problem here. Given the estimates $\widehat{\boldsymbol{h}}_k, \widehat{\boldsymbol{h}}_{k-1}, \dots, \widehat{\boldsymbol{h}}_{k-L+1}$, we can derive a linear estimator for either the vector or for each element in the vector individually. It would be more accurate to predict for the vector if there are high correlation between the elements, but the computational complexity is also much higher. When the sources that cause channel variation are mainly the residual CFOs, there are generally little correlation between them. Hence we use element-wise prediction here. Let $\hat{h}_k$ denote any element in $\widehat{\boldsymbol{h}}_k$. The linear predictor is given by

$$\hat{h}_{k+1} = \alpha_1 \hat{h}_k + \alpha_2 \hat{h}_{k-1} + \cdots + \alpha_L \hat{h}_{k-1},$$

where $\alpha_l$ are the coefficients to be determined. There are various ways to determine $\alpha_l$. Here we propose a minimum mean square error criterion, which can be efficiently solved by the Levinson Recursive algorithm [17]. In the simulation, we use $L = 4$ and a correlation matrix of size 4x4 to compute the coefficients $\alpha_l$.

## V. SIMULATION RESULTS

We present simulation results for the three proposed beamforming vector generation and updating schemes, together with one using fixed value of $K = \frac{M}{N}$. The system setup is as described in Section II. In the following presentation, the SNR is defined as the mean SNR of the received signal from each UAV, and the DTB gain is normalized to the maximum ideal value when the channel is perfectly known. We will compare DTB gains under the conditions with and without residual CFO, in line-of-sight multipath only channels or more general multipath channels simulated using the well-known Jakes model. For the residual CFO, we use ppm (parts per million) to represent its value, e.g., 1ppm for a carrier frequency 900MHz means a residual CFO of 900Hz. The residual CFO value for each UAV is generated following a uniform distribution between 0 and a specified maximum value.

We first consider channels with only line-of-sight multipath between UAVs and the base. In this case, Doppler frequency causes minor variations of the channel. Figs. 5 and 6 plot the normalized beamforming gain in the absence and presence of residual CFO, respectively, where $N$=$M$. From the two figures we can have the following three observations. (1) The adaptive method (Method 1) performs very well when CFO is very small because it can combine many subblocks to improve the SNR in channel estimation. It adapts to channel variation and achieves performance close to the one with fixed K when CFO is large; (2) The method calculating the phase shift (Method 2) performs much better than Method 1 when CFO is large, as expected; and (3) The prediction method (Method 3) performs well in both cases thanks to its capability of both channel prediction and improving SNR. In Fig. 6, Method 3 outperforms Method 2 when more samples become available because its prediction accuracy improves over time.

We then consider channels generated from the Jakes model, which combines multiple multipath signals with randomly generated Doppler frequencies for each UAV. The number of multipath used in the model is 7. Fig. 7 shows two sample channels generated using this model, with $N$=$M$. For these channels, Figs. 8 and 9 present the results for the cases without CFO and with CFO=4ppm, respectively. From these figures, we can get observations similar to those from Figs. 5 and 6.

Finally, in Fig. 10, we show the results with $N$=$M/2$, and the other setup is similar to those used in Fig. 8. In this case, the single complete training block is split into two sub-blocks, and one is put in the middle and the other is put in the end of the packet. The basic channel estimates, that are subsequently used as inputs to Method 2 and 3, are obtained by combing received signals over $K$=$2$ sub-blocks. The figure indicates that Method 2 performs relatively well due to its phase tracking capability. The performance of Method 3 degrades notably as its prediction accuracy is sensitive to the deteriorated inputs for prediction.

Based on these simulation results, together with the underlying principle of these three methods, we can have the following summary for the physical meaning of these methods, and their respective advantages and disadvantages. (1) Method 1 intends to achieve a good balance between collecting more training signals and avoiding introducing too large channel variations. It hence works great when channel changes slowly and has a low complexity. But it becomes inferior for fast varying channels. (2) Method 2 tries to estimate the phase difference between varying channels to update the BF vector. It is simple and efficient when channels have one dominating multipath and CFO is large. However, it is inefficient in combining training signals. (3) Method 3 is a statistically optimal solution in a determined problem (when $N$ equals to or larger than $M$), and it can achieve both good prediction for phase changes and combination of training signals for improved SNR. However, its performance can degrade significantly when $N<M$.

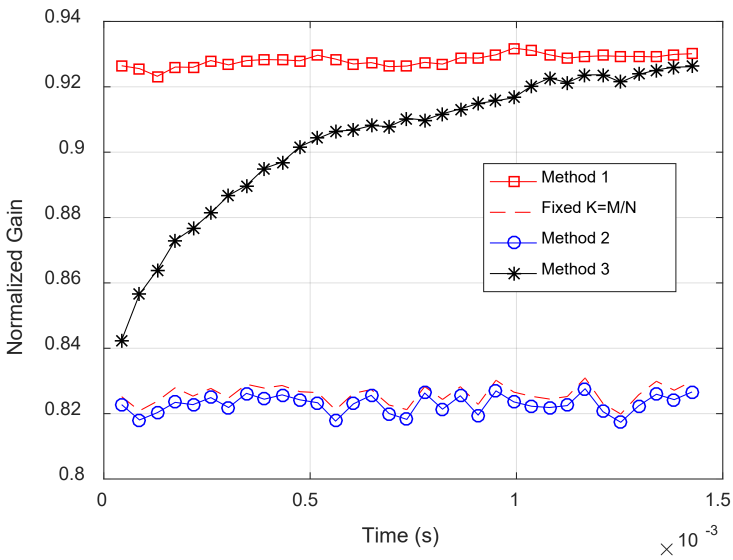


**Fig.5** *DTB gain with CFO=0, SNR=-5dB. LOS channels.*

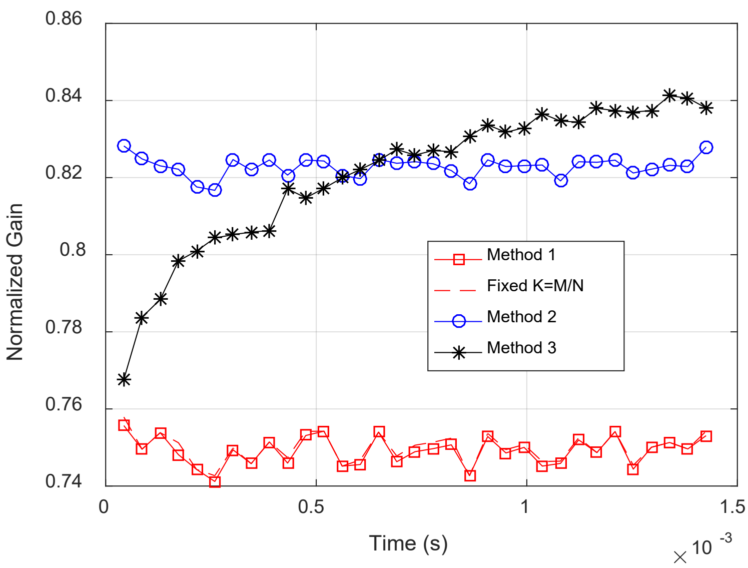


**Fig.6** *DTB gain with CFO up to 4ppm, SNR=-5dB*

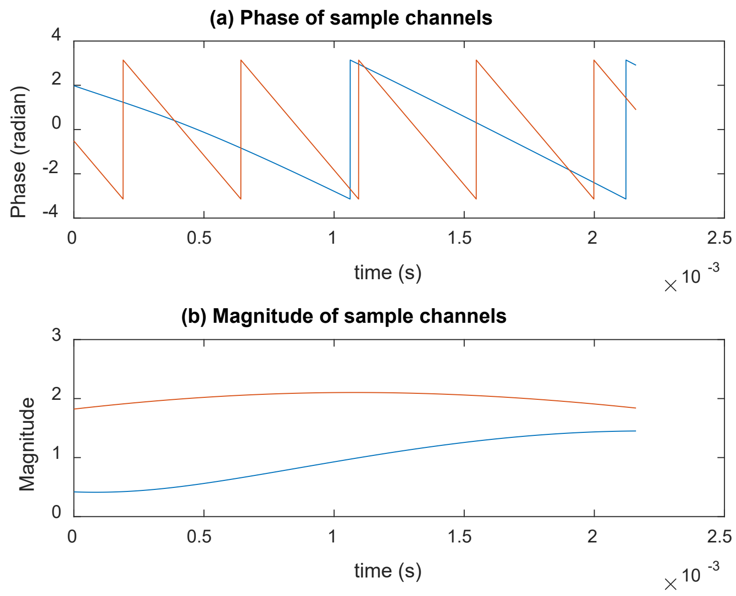


**Fig.7** *The phase (subfigure a) and magnitude (subfigure b) of sample channels between two UAVs and the base, generated using the Jakes model.*

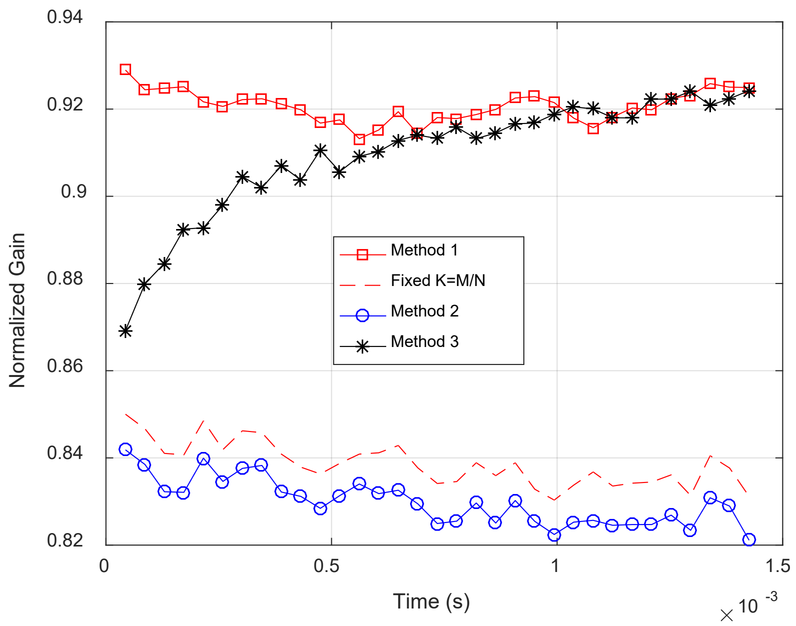


**Fig.8** *DTB gain with CFO=0, SNR=-5dB. Channels generated from the Jakes model*

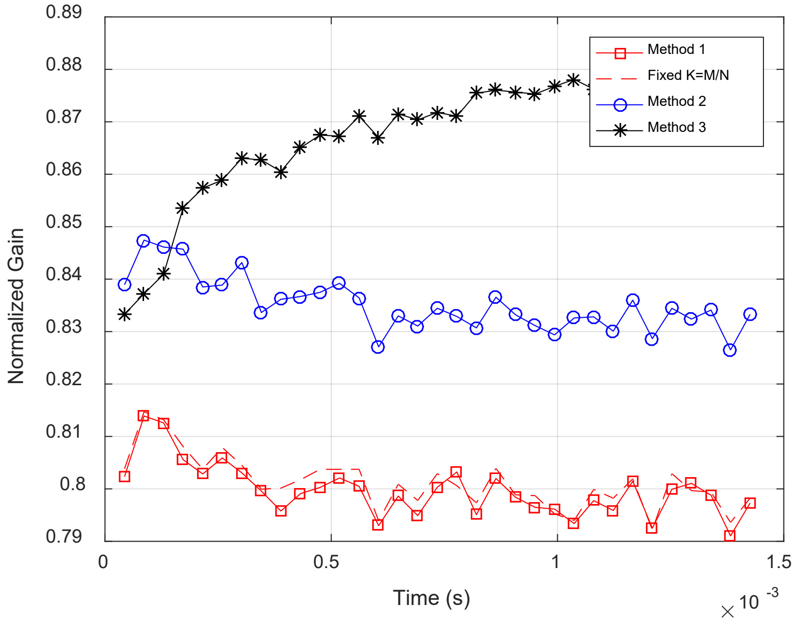


**Fig.9** *DTB gain with CFO=4ppm and SNR=-5dB. Channels generated from the Jakes model.*

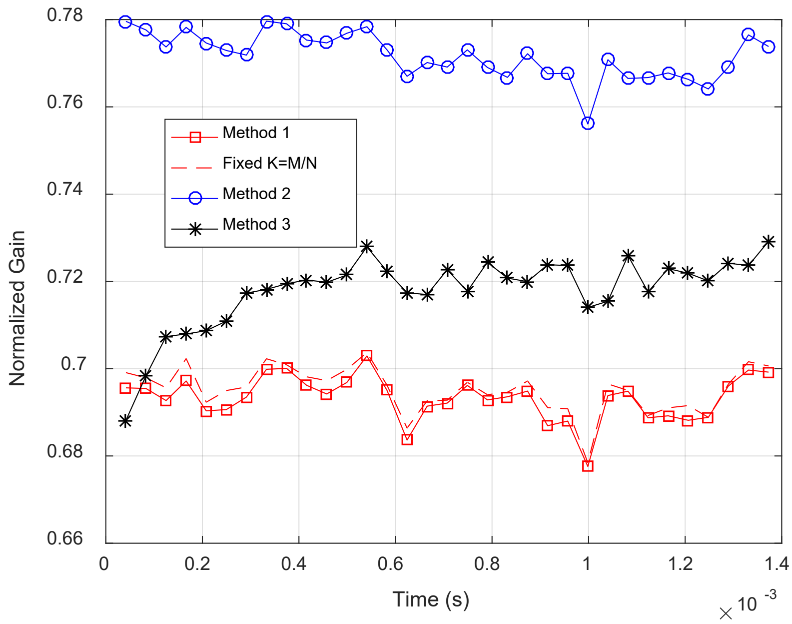


**Fig.10** *DTB gain with CFO=4ppm, SNR= -5dB, and N=M/2. Channels generated from the Jakes model.*

## VI. Conclusion

Distributed transmit beamforming (DTB) is an efficient solution for extending the communication distance between a warm of UAVs and the base. We reviewed major function modules for realizing DTB, including timing and carrier synchronization, phase drift tracking and compensation, and beamforming vector generation and updating. We also discussed potential solutions to implementing these modules. In particular, we introduce the concatenated training scheme and a recursive channel estimation and updating algorithm for generating and updating beamforming vectors. We also proposed three approaches for tracking channel variation and updating the vector: Adaptive-K, phase estimation, and channel prediction. The adaptive-K method is simple and can achieve great performance when channel variation is slow. The prediction method achieves the best performance but also has the highest complexity. The phase estimation method is simple for implementation and performs well when channels change rapidly. Our future work includes developing a new method that is robust to channel variations by combining the Adaptive-K and the phase estimation methods.

## References


[1] Uher J., Wysocki T. and Wysocki B., "Review of Distributed Beamforming," *Journal of Telecommunications and Information Technology,* no. 1, pp. 78-88, 2011.

[2] Mudumbai R., Brown D., Madhow U. and Poor, H., "Distributed transmit beamforming: challenges and recent progress," *IEEE Communications Magazine,* vol. 47, no. 2, pp. 102-110, 2009.

[3] Ahmed M. F. A. and Vorobyov S. A., "Collaborative Beamforming for Wireless Sensor Networks with Gaussian Distributed Sensor Nodes," *IEEE Transactions on Wireless Communications*, vol. 8, no. 2, pp. 638-643, 2009.

[4] Mudumbai R., Barriac G. and Madhow U., "On the Feasibility of Distributed Beamforming in Wireless Networks," *IEEE Transactions on Wireless Communications*, vol. 6, no. 5, pp. 1754-1763, 2007.

[5] Jayaprakasam S., Rahim S. K. A. and Leow C. Y., "Distributed and Collaborative Beamforming in Wireless Sensor Networks: Classifications, Trends, and Research Directions," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2092-2116, 2017.

[6] Kocaman I., Distributed Beamforming in a Swarm UAV Network, Defense Technical Information Center, 2008.

[7] Koutsonikolas D., Jafri S. A. R. and Hu Y. C., "Energy-efficient MAC and routing design in distributed beamforming sensor networks," in *Proceedings of the 2007 ACM CoNEXT conference*, New York, 2007, pp. 1-2.

[8] Meyr H., Moeneclaey M., Fechtel S., Digital Communication Receivers: Synchronization, Channel Estimation, and Signal Processing, John Wiley & Sons Inc, 1998.

[9] Brown D. R. and Poor H. V., "Time-Slotted Round-Trip Carrier Synchronization for Distributed Beamforming," *IEEE Transation on Signal Processing*, vol. 56, no. 11, pp. 5630-5643, 2008.

[10] Suleiman U. A. M. N., Esa M., Yusof K., Yusoff M. and Hamid M., "A review on frequency synchronization in collaborative beamforming: A practical approach," *Journal of Advanced Research in Applied Mechanics*, vol. 31, no. 1, pp. 1-15, 2017.

[11] Brown D. R., Mudumbai R., and Dasgupta S., "Fundamental limits on phase and frequency tracking and estimation in drifting oscillators," in *IEEE International Conference on Acoustics, Speech and Signal Processing* (*ICASSP*), Kyoto, 2012, pp. 5225-5228.

[12] Peiffer B., Goguri S., Dasgupta S., and Mudumbai R., "An approach to Kalman filtering for oscillator tracking," in *49th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, 2015, pp. 261-265.

[13] Ochiai H., Mitran P., Poor H. V., and Tarokh V., "Collaborative Beamforming for Distributed Wireless Ad Hoc Sensor Networks," *IEEE Transactions on Signal Processing*, vol. 53, no. 11, pp. 4110-4124, 2005.

[14] Mudumbai R., Hespanha J., Madhow U., and Barriac G., "Distributed Transmit Beamforming Using Feedback Control," *IEEE Transactions on Information Theory*, vol. 56, no. 1, pp. 411-426, 2010.

[15] Sriploy P., and Uthansakul M., “Nonfeedback Distributed Beamforming Using Spatial-Temporal Extraction,” *International Journal of Antennas and Propagation*, vol. 16, no.2, pp. 1-16, 2016.
[16] Zhang J. A., Yang T. and Zhuo C., “Under-determined Training and Estimation for Distributed Transmit Beamforming Systems,” *IEEE Transactions on Wireless Communications*, vol. 12, no. 4, pp. 1936-1946, 2013.
[17] Vaidyanathan P. P., The Theory of Linear Prediction, California Institute of Technology: Morgan & Claypool, 2008.