Phase-Level Synchronization for Physical-Layer Network Coding

Yang Huang*, Qingyang Song*, Shiqiang Wang†, and Abbas Jamalipour‡
*School of Information Science and Engineering, Northeastern University, Shenyang 110819, P. R. China
†Department of Electrical and Electronic Engineering, Imperial College London, SW7 2AZ, United Kingdom
‡School of Electrical and Information Engineering, University of Sydney, NSW, 2006, Australia
Email: hyocean198927@gmail.com, songqingyang@ise.neu.edu.cn, shiqiang.wang11@imperial.ac.uk, a.jamalipour@ieee.org

***Abstract*—Physical-layer network coding (PNC) brings throughput improvement for wireless networks. However, its synchronization requirement is widely recognized as an obstacle to its implementation. In this paper, we focus on phase-level synchronization and propose a time-slotted carrier synchronization scheme for PNC. We then analyze the phase error tolerance of PNC under different bit error rate (BER) requirements, and the synchronization overhead for obtaining synchronous signals below the phase error margin. We also consider the impact of different hardware (in particular, the phase-locked loop) parameters on the overhead in our analysis. Afterwards, we evaluate the performance of the proposed synchronization scheme with simulations. The results show that the proposed scheme is feasible with some typical hardware parameters. The throughput gain of PNC when using the proposed scheme is only slightly lower than the theoretical gain.**

***Index Terms*—Bit error rate (BER) analysis; phase error; physical-layer network coding (PNC); synchronization; wireless networks.**

I. Introduction

Relay-aided communications are widely adopted in cellular networks and ad hoc networks, when end nodes cannot directly communicate with each other [1], [2]. Compared with the store-and-forward relaying method, network coding reduces the necessary number of communication phases, and thereby increases network throughput [3]. Physical-layer network coding (PNC) employs not only the broadcast nature of wireless channel but also the natural network coding ability derived from the superposition of electromagnetic waves [4], which makes PNC benefit more throughput improvement than conventional network coding (CNC). This paper considers a typical two-way relay network, and the PNC process includes multiple access (MA) phase and broadcast (BC) phase, as illustrated in Fig. 1. In this paper, we focus on the denoise-and-forward (DNF) scheme of PNC, because it offers lower packet error rate compared with the amplify-and-forward scheme [5].

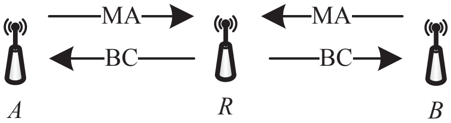

Fig. 1. Two-way relay network with physical-layer network coding.

Synchronization is a significant and difficult issue for PNC. Existing works on PNC often assume the presence of phase synchronization [6]–[8]. Recently, there is also much focus on asynchronous PNC schemes [9], [10]. However, these schemes require the tracking of phase variations during data packet transmission[1], which can be difficult especially for the superposed signal. The complexity of obtaining symbol mapping under various phase differences can also be high, in particular with high level modulations. Therefore, in this paper, we focus on synchronous PNC and propose a synchronization scheme that needs relatively few and feasible modifications on existing wireless transceivers. Meanwhile, synchronous PNC also has some benefits. Because the superposed signals align with each other, the use of more efficient signal constellations [11] and capacity-approaching channel codes, such as lattice codes [12], becomes possible. It was also proven in [13] that synchronous PNC can nearly reach the capacity region of the Gaussian two-way relay channel.

Phase-level synchronization schemes have been extensively studied under the context of distributed beamforming. A round-trip carrier synchronization scheme for beamforming was studied in [14], and its bit error rate (BER) with phase error was analyzed in [15]. Recently, synchronization schemes for beamforming were implemented both in acoustic communication systems [16] and in RF systems operating at the 2.4 GHz frequency band [17]. However, PNC differs from distributed beamforming in the sense that, in beamforming, multiple end nodes transmit identical data to the destination [16], while in PNC, two end nodes exchange different messages via the relay. This difference implies that the mutual communication among source nodes which is often used to assist the synchronization process in beamforming is not applicable to PNC. Meanwhile, only two nodes are generally involved in the PNC process [18] and we only need to keep two nodes (rather than multiple nodes as in beamforming) synchronized. Their synchronization precision requirements can also be different. In this paper, we propose a time-slotted carrier synchronization scheme for PNC and analyze its feasibility under different hardware parameters.

The remainder of this paper is organized as follows. Section

[1] Note that the phase difference between the two signals that are superposed may continuously change due to frequency drifts.

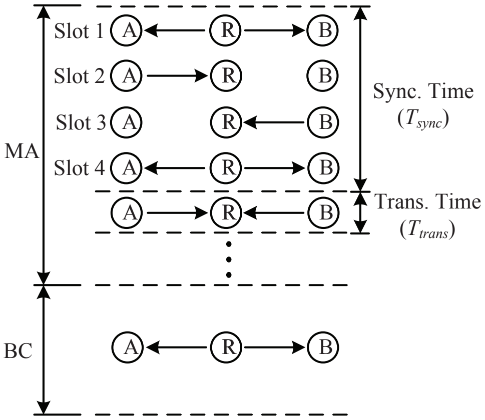

Fig. 2. Timing diagram of the proposed synchronization scheme, where the synchronization (sync.) time and the transmitting (trans.) time alternate over the MA phase.

II discusses the proposed synchronization scheme. In Section III, we analyze the relationship between the BER and the phase error level. Section IV studies the impact of the overhead and phase-locked loop (PLL) parameters on the throughput. Conclusions are drawn in Section V.

II. Proposed Synchronization Scheme

This section proposes a carrier synchronization scheme for PNC. In the proposed scheme, round-trip estimation is exploited for phase compensation, which is widely used in the synchronization for beamforming, such as [14] and [19]. However, unlike in [14] and [19], the scheme for PNC does not allow the two end nodes to share information, and phase ambiguity (as discussed in Section II-B) can be resolved by checking the preamble of each frame. The synchronization procedure includes frequency synchronization, phase compensation and phase inversion correction.

A. Synchronization Process

Fig. 2 depicts the timing diagram of the proposed time-slotted synchronization scheme for a two-way relay network. In the MA phase, synchronization and transmission are performed alternately. The lengths of the synchronization time T_{sync} and the transmitting time T_{trans} depend on the BER requirement and several hardware parameters, which will be further discussed in Sections III and IV.

The synchronization process is divided into four timeslots. In timeslot 1, the relay R broadcasts a beacon $b_0(t) = \cos(2\pi f_c t + \phi_0)$, where f_c denotes the reference frequency, and ϕ_0 is defined as the initial phase at $t = 0$. The received beacon $b_{R,A}(t)$ at end node A (the case for node B is similar, therefore we only focus on node A in the subsequent discussions) is given by $b_{R,A}(t) = \cos(2\pi f_c t + \phi_0 - \phi_{A-R})$, where ϕ_{A-R} denotes the phase offset between the receiving node A and the sending node R. Remark that we neglect the amplitude attenuation here. The received signal $b_{R,A}(t)$ is input into a PLL at node A. With the continuous input of the beacon, the PLL keeps adjusting frequency and finally becomes locked, and the necessary time for the PLL to get locked is known as settling time. Then, with a sample and hold circuit, the desired output of the loop filter in the PLL is held and used for the subsequent recovery of the reference carrier. We assume the use of a second-order PLL, which is often used in the precision analysis of synchronization schemes, such as in [19]. The steady-state phase error of such a PLL is approximately zero due to a large DC loop gain [20]. In practice, digital PLLs can also be used, which generally offer higher performance than second-order analog PLLs.

In timeslots 2 and 3, the end nodes A and B respectively bounce the recovered beacons back to the relay R. By assuming channel reciprocity, the beacon that is bounced by node A and received at the relay R is given by $b_{A,R}(t) = \cos(2\pi f_c t + \phi_0 - 2\phi_{A-R} + \phi(t))$, where $\phi(t)$ denotes the observed phase rotation caused by the frequency noise inside the free-running oscillator. We neglect the effect of $\phi(t)$ because the synchronization time is generally very short. According to the generated $b_0(t)$ and the received $b_{A,R}(t)$, the relay R can estimate the phase offset ϕ_{A-R}. Thus, the estimated ϕ_{A-R} is given by

$$\hat{\phi}_{A-R} = ((2\phi_{A-R}) \bmod (2\pi))/2\,. \qquad (1)$$

Then, in timeslot 4, the relay R sends the estimated phase offsets back to the end nodes. The end nodes compensate the phases of their carrier signals to ensure phase synchrony during data transmission.

During the transmitting time that follows, the PLL operates in open-loop mode. Due to the synchronization errors caused by noise, the phase drift increases with time. Consequently, synchronization needs to be performed periodically, as shown in Fig. 2. The synchronization period also needs to be within the channel coherence time.

B. Phase-Inversion Correction

Eq. (1) indicates that the effective estimation range of phase offset is $[0, \pi)$. Hence, the round-trip estimation method suffers from π-ambiguity (for instance, offsets of $\pi/4$ and $5\pi/4$ will be both estimated as $\pi/4$). An additional phase shift of π normally leads to an inversion of the corresponding data bits. When using the XOR-based network coding scheme, this issue can be resolved by making use of the preamble of each data frame, which is a preset sequence known to both nodes. When receiving the superposed data frame, the relay first decodes the superposed preamble. If the decoded preamble is the inverse of the desired result, an inverse operation will be performed on the network-coded data before forwarding. This method can solve the phase-inversion problem because of the properties of the XOR operation: $X_1 \oplus \overline{X_2} = \overline{X_1 \oplus X_2}$ and $\overline{X_1} \oplus \overline{X_2} = X_1 \oplus X_2$.

III. BER Analysis with Phase Error

This section analyzes the relationship between the required average BER and the tolerable phase error. We focus on the quadrature phase-shift keying (QPSK) modulation in this and the subsequent sections, while the results can also easily be

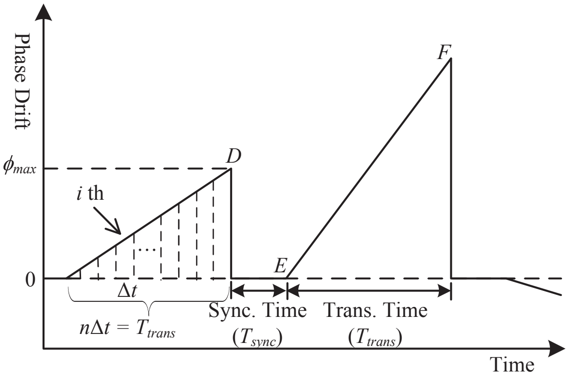

Fig. 3. Phase drift over time (phase error process).

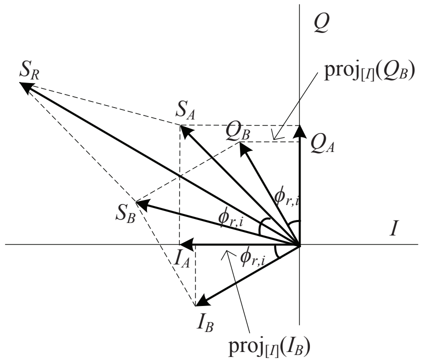

Fig. 4. Constellation of the received signal at the relay with a relative phase drift of $\phi_{r,i}$.

extended to higher level modulation schemes. We first study the phase error occurring in the transmitting period. Then, we derive the average BER under different phase error levels.

A. Phase Error Model

Upon receiving the broadcasted beacon in timeslot 1, end nodes estimate the frequency and phase of the received signals through their PLLs. During this process, additive white noise causes a random frequency offset Δf in the estimated frequency. Within the transmitting time T_{trans}, Δf causes phase drifts over time. We mainly focus on the dominant frequency offsets caused by white noise and neglect impacts from flicker noise that generally has lower power [19]. As shown in Fig. 3, the phase error process is a cyclostationary zero-mean white Gaussian process, in which Δf relates to the additive white noise following a zero-mean Gaussian distribution [20]. The phase error at the beginning of the transmitting time is regarded as zero, since the steady-state error of second-order PLLs is very small. Due to the frequency offset, the phase drift increases linearly until the next synchronization and reaches a peak ϕ_{max}. Therefore, ϕ_{max} follows the same distribution, and the variance of ϕ_{max} can be regarded as a metric of phase error.

B. Tolerence to Phase Error

To estimate the impact of parameters related to the synchronization scheme, we analyze the BER under different phase error levels.

Fig. 4 shows the superposed signals from end nodes received at the relay R in the transmission stage. S_R denotes the superposed signal. S_A and S_B represent QPSK signals from node A and node B respectively. With power control, we get $E[|S_A|^2] = E[|S_B|^2]$, and the corresponding orthogonal components I_A, I_B, Q_A and Q_B share the same amplitude given by $a = \sqrt{E_b}$. When synchronization is performed, and minimum distance estimation [9] or hard decision decoding [21] is employed, S_R is divided into channels I and Q, and the decision thresholds are $-a$ and a for the superposed signal. When a relative instantaneous phase drift of $\phi_{r,i}$ occurs between S_A and S_B at the ith instant as shown in Fig. 3, I_B and Q_B rotate an angle of $\phi_{r,i}$ from I_A and Q_A respectively, where i denotes the index of the instantaneous phase drift. Since a receiver usually estimates channel state information through preambles [22], we assume that the receiver can only track the phase rotation from knowledge of the preamble at the beginning of each data frame. Hence, the receiver is unaware of these undesired rotations in the subsequent symbols, and the decoding thresholds remain unchanged. In cases where the receiver can track the phase rotations and/or soft decoding is used, the BER performance will be no worse than that in our analysis. Therefore, our analysis provides a conservative value of the BER.

When the end nodes transmit symbols with equal probability, the BER calculated on the I-axis and the Q-axis are euqal. Therefore, we only focus on decoding in channel I in the subsequent discussions. As shown in Fig. 4, the superposed signal in channel I is given by $proj_{[I]}(S_R) = I_A + proj_{[I]}(I_B) + proj_{[I]}(Q_B)$, where $proj_{[x]}(y)$ denotes the orthogonal projection of the vector y into the line spanned by the vector x. When phase asynchrony exists, the noise-free superposed signal projected on the I-axis takes up to eight possible values. Hence, the superposed signal under the kth case is given by: $proj_{[I]}(S_{R,k}) = c_1(1 + c_2 \cos\phi_{r,i} + c_3 \sin\phi_{r,i})a$, where the coefficients c_1, c_2 and c_3 under different value of k are given in Table I. Hence, at the relay R, the exact BER $P_{e,i}$ with a relative instantaneous phase drift can be easily obtained by:

$$\begin{aligned}
P_{e,i}(\phi_{r,i}) = \frac{1}{4}\Bigg\{ & 2Q\left((\cos\phi_{r,i} + \sin\phi_{r,i})\sqrt{\frac{2E_b}{N_0}}\right) \\
& - Q\left((2 + \cos\phi_{r,i} + \sin\phi_{r,i})\sqrt{\frac{2E_b}{N_0}}\right) \\
& + Q\left((2 - \cos\phi_{r,i} - \sin\phi_{r,i})\sqrt{\frac{2E_b}{N_0}}\right) \\
& + 2Q\left((\cos\phi_{r,i} - \sin\phi_{r,i})\sqrt{\frac{2E_b}{N_0}}\right) \\
& - Q\left((2 + \cos\phi_{r,i} - \sin\phi_{r,i})\sqrt{\frac{2E_b}{N_0}}\right)
\end{aligned}$$

TABLE I
COEFFICIENTS OF $proj_{[I]}[S_{R,k}]$ UNDER DIFFERENT VALUES OF k

k	1	2	3	4	5	6	7	8
c_1	-1	$+1$	$+1$	-1	-1	$+1$	$+1$	-1
c_2	$+1$	$+1$	-1	-1	$+1$	$+1$	-1	-1
c_3	$+1$	-1	-1	$+1$	-1	$+1$	$+1$	-1

$$+Q\left((2-\cos\phi_{r,i}+\sin\phi_{r,i})\sqrt{\frac{2E_b}{N_0}}\right)\Bigg\}\,, \quad (2)$$

where E_b/N_0 denotes the average per-bit signal to noise ratio (SNR) at the relay R, and N_0 represents the power spectrum density of additive white Gaussian noise (AWGN).

The relative phase drift $\phi_{r,i}$ at a certain time t can be evaluated by $\phi_{r,i}(t) = \phi_{maxr}t/T_{trans}$, where ϕ_{maxr} denotes the maximum relative phase drift, with standard deviation σ_{maxr}. Due to the stochastic frequency offset and cyclic transmitting time, $\phi_{r,i}$ at the ith instant of each cycle follows the same distribution. By using the infinitesimal approach as illustrated in Fig. 3, the variance of $\phi_{r,i}$ is obtained as

$$\sigma_{r,i}^2 = \frac{t^2}{T_{trans}^2}\sigma_{maxr}^2 \underset{T_{trans}=n\cdot\Delta t}{\overset{t=i\cdot\Delta t}{=\!=\!=\!=\!=\!=}} \left(\frac{i}{n}\right)^2 \sigma_{maxr}^2\,. \quad (3)$$

We then achieve the average BER for the ith time instant:

$$\overline{P}_{e,i} = \int_{-\pi}^{\pi} P_{e,i}(\phi_{r,i})p_{r,i}(\phi_{r,i})\,d\phi_{r,i} \ \ (i = 1, 2 \ldots n)\,, \quad (4)$$

where $P_{e,i}$ is the BER which is evaluated from (2), and $p_{r,i}(\phi_{r,i})$ represents the probability density function (PDF) of $\phi_{r,i}$, with $\phi_{r,i} \sim \mathcal{N}(0, \sigma_{r,i}^2)$.

According to Borel's law of large numbers, the average BER can be approximated by:

$$\begin{aligned}\overline{P}_e &= \frac{\overline{P}_{e,1}R_b\Delta t + \overline{P}_{e,2}R_b\Delta t + \cdots + \overline{P}_{e,n}R_b\Delta t}{nR_b\Delta t} \\ &= \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\overline{P}_{e,i}\,, \end{aligned} \quad (5)$$

where R_b denotes the bit rate on unit bandwidth, which is measured in bps/Hz. Combining (4) with (5), the average BER is further derived as

$$\overline{P}_e = \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\int_{-\pi}^{\pi} P_{e,i}(\phi_{r,i})p_{r,i}(\phi_{r,i})\,d\phi_{r,i}\,. \quad (6)$$

Since the values of ϕ_{max} in different synchronization cycles are independent and identically distributed, the standard deviation of ϕ_{max} is given by $\sigma_{max} = \sigma_{maxr}/\sqrt{2}$. Then, according to (2), (6) can be also expressed as a function of σ_{max} and E_b/N_0, i.e.

$$\overline{P}_e = F(\sigma_{max}, E_b/N_0)\,. \quad (7)$$

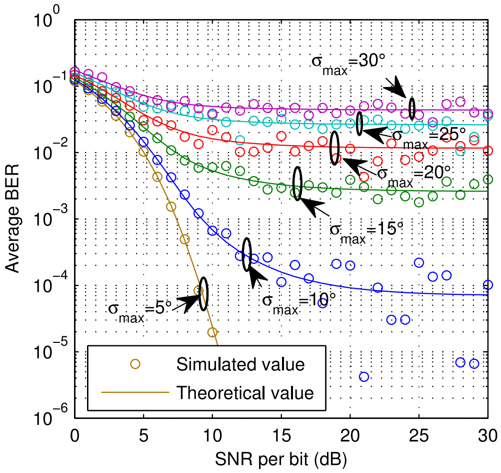

Fig. 5. Comparisons between the theoretical average BERs and their simulated values.

C. Simulation Results

Assume $n = 25$, the theoretical average BER values are compared with the Monte Carlo simulation results, as shown in Fig. 5. In the simulation, 2.56×10^4 bits are transmitted each time. We performed 800 simulations to obtain the overall values. It can be observed that the theoretical results nearly agree with the simulated results, although some imperfect match exists in low BER regions due to the limited number of transmitted bits in the simulations. Hence, the BER performance with impact of phase error can be predicted from (7), and this relationship between the average BER and the phase error can be used to determine the tolerable phase error with a given BER requirement in the subsequent analysis. It can also be observed that the BER increases with the rising phase error scale σ_{max}. The average BER curves do not keep a downward trend with the increasing SNR, but level off and converge to stable values. The reason is that, in the high SNR regime, the bit errors are mainly caused by the phase error.

IV. THROUGHPUT ANALYSIS

This section first studies the synchronization overhead introduced by the proposed scheme, and then analyzes the throughput performance with different PLL parameters.

A. Synchronization Overhead

Ideally, the throughput of PNC is $S_{PNC} = R_b$ and the throughput of CNC is $S_{CNC} = 2R_b/3$. However, when performing PNC, phase-level synchronization introduces additional overhead, and the actual throughput is

$$S'_{PNC} = R_b(1 - overhead)\,. \quad (8)$$

Hence, the actual throughput gain of PNC over CNC is given by $G = 3(1 - overhead)/2$. Since PNC contains MA and BC phases, the time consumption for data exchange of one transmission period is $2T_{trans}$. Therefore, the synchronization overhead can be defined as

$$overhead = T_{sync}/(T_{sync} + 2T_{trans})\,. \quad (9)$$

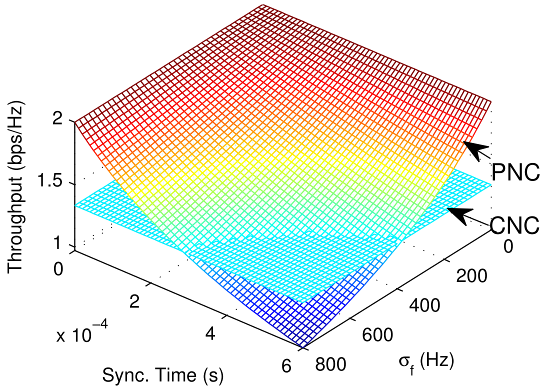

Fig. 6. Impact of the synchronization time and the frequency offset scale on the throughput.

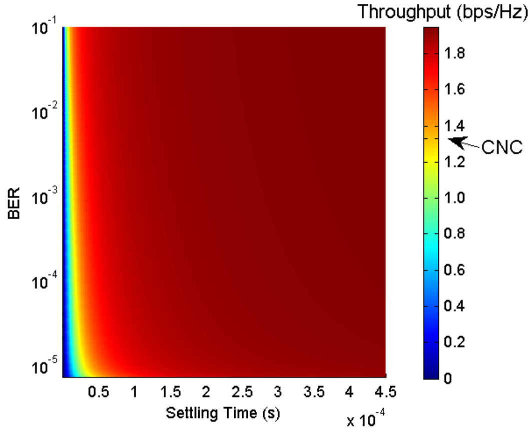

Fig. 7. Throughput performance with different average BER requirements and settling time.

According to the phase error model as shown in Fig. 3, the relationship between the frequency offset Δf and the transmitting time T_{trans} is $\phi_{max} = \Delta f T_{trans}$, and therefore the variance of ϕ_{max} is

$$\sigma_{max}^2 = \sigma_f^2 T_{trans}^2 \,, \tag{10}$$

where σ_f^2 represents the variance of the frequency offset. Similar to σ_{max}, σ_f indicates the frequency offset scale, and (10) can be rewritten as $T_{trans} = \sigma_{max}/\sigma_f$. Thus, according to (8)–(10), the throughput of PNC is given by

$$S'_{PNC} = \frac{2R_b\sigma_{max}}{\sigma_f T_{sync} + 2\sigma_{max}} \,. \tag{11}$$

B. Impact of The PLL Settling Time

In the proposed scheme, a typical second-order PLL is applied, and the transfer function is given by

$$H(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \,, \tag{12}$$

where ω_n and ξ denote the natural frequency and the damping ratio respectively. The settling time T_s is the time for a PLL to track the phase until getting locked, and $T_s \approx 4/(\xi\omega_n)$ in a second-order system [23]. Since the loop filter in the PLL is designed with a certain bandwidth, the settling time and the hold time (i.e. transmitting time) are related with each other. It follows that the synchronization time interacts with the transmitting time, which requires an explicit analysis.

With the analytical approach proposed in [19], we calculate the statistics at specific time instants D, E and F in Fig. 3, to associate the transmitting time T_{trans} with the settling time T_s. At the start of the transmitting time (i.e. time E), $\phi_E = \rho\phi_{max} + \psi_n$, where $\rho\phi_{max}$ denotes the initial error which is fraction of ϕ_{max}, and ψ_n denotes the the phase rotation term which is caused by noise. The variance of ϕ_E is $\sigma_E^2 = \rho^2\sigma_{max}^2 + \omega_n N_p$, where N_p denotes the power spectral density (PSD) of the frequency noise. During the transmitting time, the frequency offset is $\Delta f = \omega_n\phi_E = \omega_n\rho\phi_{max} + \omega_n\psi_n$, and $\sigma_f^2 = \rho^2\omega_n^2\sigma_{max}^2 + \omega_n^3 N_p$. At time F, $\phi_F = T_{trans}\Delta f + \phi_E$. Since ϕ_E is small, the variance of ϕ_F is $\sigma_F^2 \approx \sigma_f^2 T_{trans}^2$. Due to the cyclic procedure, $\sigma_F^2 = \sigma_D^2 = \sigma_{max}^2$. After combining the above statistics, according to [19], we achieve

$$T_{trans} = \left(\frac{\xi^3 T_s^3 \sigma_{max}^2}{64N_p + 16T_s\rho^2\xi\sigma_{max}^2} \right)^{1/2} \,. \tag{13}$$

As aforementioned, T_{sync} is composed of the settling time T_s and the time for control data transmission. In timeslot 1, the relay R keeps sending the beacon until the PLLs in the end nodes are locked. Ignoring the propagation time, the duration of timeslot 1 equals T_s. The subsequent synchronization timeslots are used for control data transmission, and the duration is denoted by T_{ctrl}. Hence, T_{sync} can be written as $T_{sync} = T_s + T_{ctrl}$. With (8), (9) and (13), we can obtain the relationship between the throughput and the settling time of the PLL in the proposed synchronization scheme.

C. Simulation Results

In this subsection, we evaluate the throughput performance of the proposed scheme with consideration of the synchronization overhead. In the simulations, the PSD of the frequency noise N_p is set to $7 \times 10^{-11}\,\text{Hz}^{-1}$ ($-101\,$dBc/Hz) [24], the damping ratio ξ is set to 1 [25], the initial phase error fraction ρ is set to $1\,\%$ [19], T_{ctrl} is set to $200\,\mu$s, and R_b is set to 2 bps/Hz.

Fig. 6 shows the impact of the frequency offset and the synchronization time on the throughput. Assume the required average BER is 1.0×10^{-3}. When the per-bit SNR is 15 dB, according to (7), the sustainable phase error scale is $\sigma_{max} \approx 13°$. It can be observed that with the increasing necessary synchronization time, a lower-noise oscillator (with smaller σ_f) has to be used to maintain the throughput at the same value.

Then, the throughput performance under different settling time of the PLL is investigated when the per-bit SNR is 10 dB. Fig. 7 shows the throughput performance under different average BER requirements and settling time. It can be observed that with a certain acceptable average BER, the throughput increases with the settling time and finally converges to a maximum value. This observation is because a larger settling time allows the PLL to be synchronized more precisely, making a larger transmission time possible, which also follows from (13). However, in practice, the PLL design has to balance the time needed for synchronization against the synchronization precision. Some alternative noise sources may also exist in practice, which reduces the actual transmission time. When PNC is performed and the required average BER at the relay R is 1.367×10^{-4}, the throughput gain over CNC converges to approximately 1.467, which is slightly lower than the theoretical throughput gain 1.5 [4]. Additionally, from the numerical results, it is clear that, with the proposed synchronization scheme, PNC outperforms CNC in most cases with reasonable BERs.

V. Conclusions

In this paper, we have proposed a phase-level synchronization scheme for PNC. The proposed scheme is based on round-trip estimation and performs synchronization periodically. Then we have investigated the relationship between the average BER requirement and the tolerable phase error. Based on this relationship, we have analyzed the performance under different conditions such as the frequency offset and the PLL settling time. The simulation results indicate that with the proposed synchronization scheme, the throughput gain of PNC over CNC can approach its theoretical value. We have focused on QPSK modulation in this paper. The case of higher-level modulated PNC as well as countermeasures for insufficient symbol alignment will be considered in our future work.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (61172051, 61071124), the Fok Ying Tung Education Foundation (121065), the Program for New Century Excellent Talents in University (08-0095, NCET-11-0075), the Specialized Research Fund for the Doctoral Program of Higher Education (20110042110023, 20110042120035), and the Fundamental Research Funds for the Central Universities (N100404008, N110204001, N110804003).

References

[1] Z. Ning, L. Guo, Y. Peng, and X. Wang, "Joint scheduling and routing algorithm with load balancing in wireless mesh networks," *Computers and Electrical Engineering*, vol. 38, no. 3, pp. 533–550, May 2012.

[2] L. Guo, L. Zhang, Y. Peng, J. Wu, X. Zhang, W. Hou, and J. Zhao, "Multi-path routing in spatial wireless ad hoc networks," *Computers and Electrical Engineering*, vol. 38, no. 3, pp. 473–491, May 2012.

[3] R. Ahlswede, C. Ning, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[4] S. Zhang, S. C. Liew, and P. P. Lam, "Hot topic: Physical-layer network coding," in *Proc. ACM MobiCom*, Sep. 2006, pp. 358–365.

[5] K. Lee and L. Hanzo, "Resource-efficient wireless relaying protocols," *IEEE Wireless Commun. Mag.*, vol. 17, no. 2, pp. 66–72, Apr. 2010.

[6] D. Wübben and Y. Lang, "Generalized sum-product algorithm for joint channel decoding and physical-layer network coding in two-way relay systems," in *Proc. IEEE GLOBECOM*, Dec. 2010, pp. 1–5.

[7] S. Zhang and S. C. Liew, "Channel coding and decoding in a relay system operated with physical-layer network coding," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 5, pp. 788–796, Jun. 2009.

[8] D. To and J. Choi, "Convolutional codes in two-way relay networks with physical-layer network coding," *IEEE Trans. Wireless Commun.*, vol. 9, no. 9, pp. 2724–2729, Sep. 2010.

[9] T. Koike-Akino, P. Popovski, and V. Tarokh, "Optimized constellations for two-way wireless relaying with physical network coding," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 5, pp. 773–787, Jun. 2009.

[10] L. Lu, S. C. Liew, and S. Zhang, "Optimal decoding algorithm for asynchronous physical-layer network coding," in *Proc. IEEE ICC*, Jun. 2011, pp. 1–6.

[11] M. Noori and M. Ardakani, "On symbol mapping for binary physical-layer network coding with PSK modulation," *IEEE Trans. Wireless Commun.*, vol. 11, no. 1, pp. 21–26, Jan. 2012.

[12] M. P. Wilson, K. Narayanan, H. D. Pfister, and A. Sprintson, "Joint physical layer coding and network coding for bidirectional relaying," *IEEE Trans. Inf. Theory*, vol. 56, no. 11, pp. 5641–5654, Nov. 2010.

[13] W. Nam, S.-Y. Chung, and Y. H. Lee, "Capacity of the gaussian two-way relay channel to within 1/2 bit," *IEEE Trans. Inf. Theory*, vol. 56, no. 11, pp. 5488–5494, Nov. 2010.

[14] D. R. Brown and H. V. Poor, "Time-slotted round-trip carrier synchronization for distributed beamforming," *IEEE Trans. Signal Process.*, vol. 56, no. 11, pp. 5630–5643, Nov. 2008.

[15] S. Song, J. S. Thompson, P.-J. Chung, and P. M. Grant, "BER analysis for distributed beamforming with phase errors," *IEEE Trans. Veh. Technol.*, vol. 59, no. 8, pp. 4169–4174, Oct. 2010.

[16] D. R. Brown, B. Zhang, B. Svirchuk, and M. Ni, "An experimental study of acoustic distributed beamforming using round-trip carrier synchronization," in *Proc. IEEE ARRAY*, Oct. 2010, pp. 316–323.

[17] S. Sigg, R. M. E. Masri, and M. Beigl, "Feedback-based closed-loop carrier synchronization: A sharp asymptotic bound, an asymptotically optimal approach, simulations, and experiments," *IEEE Trans. Mobile Comput.*, vol. 10, no. 11, pp. 1605–1617, Nov. 2011.

[18] S. Wang, Q. Song, X. Wang, and A. Jamalipour, "Rate and power adaptation for analog network coding," *IEEE Trans. Veh. Technol.*, vol. 60, no. 5, pp. 2302–2313, Jun. 2011.

[19] R. Mudumbai, G. Barriac, and U. Madhow, "On the feasibility of distributed beamforming in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 6, no. 5, pp. 1754–1763, May 2007.

[20] F. M. Gardner, *Phaselock Techniques*, 3rd ed. Hoboken: John Wiley & Sons, 2005.

[21] E. C. Y. Peh, Y.-C. Liang, and Y. L. Guan, "Power control for physical-layer network coding in fading environments," in *Proc. IEEE PIMRC*, Sep. 2008, pp. 1–5.

[22] W. U. Bajwa, J. Haupt, A. M. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proc. IEEE*, vol. 98, no. 6, pp. 1058–1076, Jun. 2010.

[23] F. Golnaraghi and B. C. Kuo, *Automatic Control System*. New York: John Wiley & Sons, 2003.

[24] B. Razavi, "A study of phase noise in CMOS oscillators," *IEEE J. Solid-State Circuits*, vol. 31, no. 3, pp. 331–343, Mar. 1996.

[25] J. W. M. Bergmans, "Effect of loop delay on phase margin of first-order and second-order control loops," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 52, no. 10, pp. 621–625, Oct. 2005.