

Beamforming distribuído baseado em inversão remota de fase: Uma prova de conceito em GNU-Radio/USRP

Angilberto Muniz Sobrinho, Edmar Candeia Gurjão e José Sérgio da Rocha Neto

Resumo—*Beamforming* distribuído, no contexto de redes de sensores, é uma técnica de transmissão cooperativa que pode ser usada com o objetivo de aumentar o nível de sinal no receptor e, como consequência, melhorar a relação sinal-ruído (SNR). Neste trabalho propõe-se um método de *beamforming* distribuído em malha aberta para alinhamento da fase na recepção. O método baseia-se na utilização de um protocolo local Mestre-Escravo para estimar e inverter remotamente, em banda base, a fase de chegada da portadora de um sinal piloto enviado pelo elemento interrogador da rede. Um arcabouço para prova de conceito foi construído sobre a plataforma GNU Radio, e foi utilizado para validar o método proposto. Os resultados experimentais apresentaram maior estabilidade e menor latência quando comparados aos resultados obtidos com um método popular conhecido *Time-slotted Round-trip synchronization*, também implementado sobre o mesmo arcabouço.

Palavras-Chave—Beamforming, Redes de Sensores, Sincronismo, Software Defined Radio, GNU Radio, Transponders.

Abstract—In wireless sensor networks distributed beamforming is a cooperative technique to improve the received signal quality. In this work is proposed an open-loop distributed beamforming method for carrier phase alignment at receiving destination. The approach is inspired in Retrodirective Antenna Array and is based on the phase-conjugate of a pilot (beacon) signal sent from the Master Base Station to the remote sensors nodes. A GNU Radio based experiment was implemented to validate the concept, and the obtained results shows better stability and lower latency of the proposed method compared to Time-slotted Round Trip synchronization method.

Keywords—Beamforming, Sensor Networks, Synchronization, Software Defined Radio, GNU Radio, Transponders.

I. INTRODUÇÃO

Na técnica de transmissão cooperada conhecida como “*beamforming* distribuído”, múltiplos rádios independentes e com mesma frequência transmitem simultaneamente a mesma informação utilizando portadoras cujas fases são alinhadas de forma a criar uma interferência construtiva no ponto de recepção [1]. Com isso há o aumento na relação sinal-ruído (SNR), o que pode permitir maior alcance de transmissão, maior taxa de transmissão ou menor consumo de energia, dependendo da aplicação pretendida.

Duas abordagens básicas (sistemas de malha aberta e de malha fechada) tem sido utilizadas para investigar técnicas de

beamforming distribuído, conforme mencionado em [2] e [3] que também discutem as tendências e os principais desafios existentes nessa tecnologia.

Os sistemas de malha aberta [4] se baseiam na utilização de múltiplos *phase locked loops* (PLLs) e sinais de treinamento enviados pelo receptor distante para obter a sincronização entre os elementos da rede. Em [4] o método denominado *Round-Trip Synchronization* foi proposto e posteriormente aperfeiçoado e apresentado em [5] como *Time-Slotted Round-Trip Synchronization* (RT).

Os sistemas de malha fechada se baseiam em métodos interativos nos quais a intensidade relativa do sinal recebido pelo receptor distante é informada (realimentada) para os elementos sensores que efetuam ajustes nas fases de seus osciladores locais. Em [6] é descrito um método representativo da abordagem em malha fechada e em [7] esse método é otimizado e apresentado como *One-bit Feedback Synchronization* que utiliza apenas um bit como indicador para o ajuste interativo das fases das portadoras dos sensores. Em [8] e [9] são apresentadas realizações experimentais dos métodos *One bit feedback* e *Time-slotted Round-trip*, respectivamente.

Neste trabalho é proposto um método em malha aberta, baseado na inversão da fase de um sinal de treinamento enviado por um receptor distante (interrogador). A transmissão, a partir dos sensores, de um sinal com fase inicial igual a fase de chegada do sinal de treinamento, invertida, anula o efeito do percurso e garante uma recepção coerente no ponto de recepção (interrogador). Como os transceptores dos sensores possuem osciladores locais distintos para transmissão e recepção, a inversão direta da fase do sinal recebido é inviável devido à diferença de fase entre recepção e transmissão. Então um elemento sensor é escolhido como referência e, utilizando um protocolo Mestre-Escravo, efetua a inversão de fase desejada para os outros elementos.

O trabalho está organizado da seguinte forma: na seção II é apresentado o método proposto, na seção III descrevemos a implementação do arcabouço GNU Radio, na seção IV são apresentados resultados preliminares e as conclusões são apresentadas na seção V.

II. MÉTODO PROPOSTO

O sistema considerado está ilustrado na Fig. 1 e é composto de um interrogador (D) com potência suficiente para alcançar os N sensores S_i ($i = 0, \dots, N - 1$). Os transceptores possuem osciladores locais (LO) independentes com fases iniciais para

transmissão e recepção iguais a ϕ_{tx_D} , ϕ_{tx_i} e ϕ_{rx_D} , ϕ_{rx_i} , respectivamente.

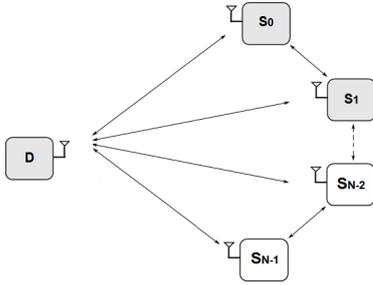


Fig. 1. Modelo de rede com 1 Interrogador e N sensores.

Como o interesse principal neste trabalho é o sincronismo de fase, assume-se que os elementos da rede se encontram perfeitamente sincronizados em frequência pelo uso compartilhado de um sinal de referência de 10 MHz. Em um cenário mais realista, diversos métodos [10] para sincronismo global de frequência estão disponíveis e mesmo sistemas de distribuição de *clock* sem fio podem ser considerados [11]-[12].

Considerando um canal sem seletividade em frequência e quasi-estático (sem alterações durante o tempo de um quadro de RF), modelado como um atraso de tempo, e sinais de banda estreita, tem-se que o efeito do canal sobre os sinais será equivalente a um deslocamento de fase que pode ser representado como θ_i para o deslocamento de fase entre o interrogador e o sensor S_i e θ_{ij} para o deslocamento de fase entre os sensores S_i e S_j . Considera-se que $\theta_{ij} = \theta_{ji}$.

A. Método de Referência (Round-Trip Synchronization)

O método *Time-slotted round-trip carrier synchronization* (RT), utiliza topologia em anel e se baseia na equivalência dos atrasos de propagação de um sinal piloto originado no interrogador (D) que é replicado de sensor a sensor, retornando ao ponto de origem (D) após “circular” o anel em duas direções opostas. Considerando a parte hachurada da Fig. 1 como exemplo para uma rede com 1 interrogador e 2 sensores, a idéia é que o sinal piloto circulando no sentido ($D \rightarrow S_0 \rightarrow S_1 \rightarrow D$) deve apresentar deslocamento total de fase igual ao deslocamento de fase apresentado no sentido ($D \rightarrow S_1 \rightarrow S_0 \rightarrow D$).

A circulação virtual do sinal pelo anel é implementada rastreando o sinal recebido pelo sensor com um *PLL* e, posteriormente, operando esse *PLL* em modo *holdover* para retransmitir uma extensão periódica do sinal recebido no intervalo de tempo anterior. São utilizados dois *PLLs* (PLL_1 e PLL_2), um para cada sentido de propagação. Para mais detalhes sobre uso de *PLL* em *beamforming* distribuído, ver [13].

Em uma rede composta de N sensores, um ciclo completo do protocolo é realizado em $2N$ intervalos de tempo (*time-slots*). Para o exemplo de rede considerado tem-se quatro intervalos de tempo, TS_0 a TS_3 :

TS_0 : O interrogador transmite um sinal piloto que é recebido por cada sensor e utilizado para sincronizar o PLL_1 .

TS_1 : O sensor S_0 transmite o sinal de seu PLL_1 para S_1 que o utiliza para sincronizar seu PLL_2 .

TS_2 : O sensor S_1 transmite o sinal de seu PLL_1 para S_0 que o utiliza para sincronizar seu PLL_2 .

TS_3 : Ambos os sensores transmitem, simultaneamente, os sinais de seus PLL_2 .

O sinais resultantes da propagação pelos percursos ($D \rightarrow S_0 \rightarrow S_1 \rightarrow D$) e ($D \rightarrow S_1 \rightarrow S_0 \rightarrow D$) podem ser representados por

$$rx_{d0}(t) = \cos(\omega t + \phi_{tx_D} - \theta_1 + \phi_{\delta 1} - \theta_{01} + \phi_{\delta 0} - \theta_0)$$

e

$$rx_{d1}(t) = \cos(\omega t + \phi_{tx_D} - \theta_0 + \phi_{\delta 0} - \theta_{01} + \phi_{\delta 1} - \theta_1),$$

respectivamente, onde $\phi_{\delta 0} = -\phi_{rx_0} + \phi_{tx_0}$ e $\phi_{\delta 1} = -\phi_{rx_1} + \phi_{tx_1}$.

O sinal efetivamente recebido pelo interrogador durante o intervalo TS_3 é $rx_d(t) = rx_{d0}(t) + rx_{d1}(t)$ resultando em

$$rx_d(t) = 2\cos(\omega t + \phi_{tx_d} - \theta_0 + \phi_{\delta 0} - \theta_{01} + \phi_{\delta 1} - \theta_1), \quad (1)$$

que é uma combinação coerente de $rx_{d0}(t)$ e $rx_{d1}(t)$. Este esquema foi implementado de acordo com diagrama de blocos representado na Fig. 2.

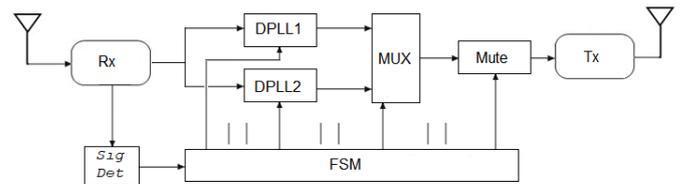


Fig. 2. Diagrama de blocos dos receptores para o método RT.

B. Método Proposto (Inversão Remota de Fase - IRF)

Considerando a rede ilustrada na Fig. 1, e ignorando os termos ϕ_{tx_D} e ϕ_{rx_D} por serem constantes e não contribuírem para diferenças relativas de fase, o sinal piloto transmitido pelo interrogador, $tx_d(t) = \cos(\omega t)$, é recebido pelos sensores como $rx_i(t) = \cos(\omega t - \theta_i - \phi_{rx_i})$. A transmissão de volta, simultânea, desse sinal pelo conjunto de sensores S_i será percebida pelo interrogador como

$$rx_d(t) = \sum_{i=0}^{N-1} \cos(\omega t - 2\theta_i - \phi_{rx_i} + \phi_{tx_i}). \quad (2)$$

Definindo a fase de (2) como $\gamma = -2\theta_i - \phi_{rx_i} + \phi_{tx_i}$, a potência do sinal recebido, dada por [7]

$$P_{rx_d}(t) = \left| \sum_{i=0}^{N-1} \cos(\omega t + \gamma) \right|^2 \quad (3)$$

é maximizada quando $\gamma = \text{constante}$. Como não existe controle direto sobre ϕ_{tx_i} , ϕ_{rx_i} nem sobre θ_i , a idéia é introduzir uma rotação adicional de fase Ψ_i ao sinal ainda em banda base de modo que o sinal a ser transmitido por S_i seja igual a $tx_i(t) = \cos(\omega t + \theta_i - \phi_{tx_i})$. Isto é equivalente a inverter θ_i , cancelar ϕ_{rx_i} e pré-cancelar ϕ_{tx_i} , mas como

esses termos são desconhecidos não é possível determinar Ψ_i diretamente. Entretanto, o efeito desejado pode ser alcançado escolhendo um sensor qualquer, S_0 por exemplo, como elemento de referência comum, denominado Mestre, com a função de inverter o sinal de todos os outros sensores S_i ($i = 1, \dots, N - 1$), denominados Escravos, como descrito a seguir.

Inicialmente o método proposto (IRF), pode ser considerado como uma variação do método RT, porém o método IRF alcança a coerência de fase na recepção pelo cancelamento do deslocamento de fase do sinal piloto causado pela propagação entre o interrogador e cada sensor S_i .

1) *Inversão Remota de Fase (IRF)*: Sequencialmente, utilizando um protocolo Mestre-Escravo, cada sensor S_i ($i = 1..N - 1$) replica o sinal $rx_i(t)$ para S_0 que é recebido como

$$rx_{0i}(t) = \cos(\omega t - \theta_i - \phi_{rx_i} + \phi_{tx_i} - \theta_{i0} - \phi_{rx_0}).$$

A fase desse sinal é invertida por S_0 gerando

$$rx_{0i_inv}(t) = \cos(\omega t + \theta_i + \phi_{rx_i} - \phi_{tx_i} + \theta_{i0} + \phi_{rx_0} + \phi_{tx_0}),$$

que é transmitido de volta por S_0 para S_i que o recebe como

$$rx_{i_inv}(t) = \cos(\omega t + \theta_i - \phi_{tx_i} + \phi_{rx_0} + \phi_{tx_0}).$$

Fazendo $tx_i(t) = rx_{i_inv}(t)$ ($i = 1..N - 1$), temos

$$tx_i(t) = \cos(\omega t + \theta_i - \phi_{tx_i} + \phi_{rx_0} + \phi_{tx_0}) \quad (4)$$

para ($i = 1..N - 1$).

E fazendo $tx_0(t)$ igual a $rx_0(t)$ com fase invertida temos

$$tx_0(t) = \cos(\omega t + \theta_0 + \phi_{rx_0}). \quad (5)$$

Quando todos os sensores, S_i ($i = 0, \dots, N - 1$), transmitem $tx_i(t)$ simultaneamente, o sinal recebido pelo interrogador devido a transmissão de S_i ($i = 1..N - 1$) é expresso por

$$rx_{d_i}(t) = \sum_{i=1}^{N-1} \cos(\omega t + \phi_{rx_0} + \phi_{tx_0}),$$

e o sinal recebido devido a transmissão de S_0 é dado por

$$rx_{d_0}(t) = \cos(\omega t + \phi_{rx_0} + \phi_{tx_0}),$$

portanto

$$rx_d(t) = \sum_{i=0}^{N-1} \cos(\omega t + \gamma), \quad (6)$$

onde $\gamma = \phi_{rx_0} + \phi_{tx_0} = \text{constante}$, maximizando (3).

2) *Minimização de Latência*: O sinal a ser transmitido, $tx_i(t) = \cos(\omega t + \theta_i - \phi_{tx_i} + \phi_{rx_0} + \phi_{tx_0})$, é equivalente a uma inversão local da fase do sinal piloto recebido $rx_i(t) = \cos(\omega t - \theta_i - \phi_{rx_i})$ com uma rotação adicional de fase $\Psi_i = -\phi_{rx_i} - \phi_{tx_i} + \phi_{rx_0} + \phi_{tx_0}$, que se mantém constante durante toda a campanha e que pode ser determinada pela inversão remota da fase de um sinal $tx_{cal_i}(t) = \cos(\omega t)$ gerado localmente durante um estágio prévio de calibração.

O protocolo é dividido em dois estágios: *Calibração*, com $2N - 2$ *time-slots* por campanha e *Operação*, com 2 *time-slots* por ciclo, definidos a seguir.

Estágio de calibração ($k = 1..N$):

TS_{2k-1} : S_k transmite $tx_{cal_k}(t) = \cos(\omega t)$ para S_0 que recebe $rx_{cal_k}(t) = \cos(\omega t + \phi_{tx_k} - \theta_{k0} - \phi_{rx_0})$. Esse sinal tem sua fase invertida por S_0 gerando

$$rx_{cal_k_inv}(t) = \cos(\omega t - \phi_{tx_k} + \theta_{k0} + \phi_{rx_0}).$$

TS_{2k} : S_0 transmite $rx_{cal_k_inv}(t)$ para S_k , que recebe

$$rx_{cal_0k_inv}(t) = \cos(\omega t - \phi_{rx_k} - \phi_{tx_k} + \phi_{rx_0} + \phi_{tx_0}).$$

A diferença de fase entre $rx_{cal_0k_inv}(t)$ e $tx_{cal_k}(t)$ é calculada por S_k gerando

$$\Psi_k = -\phi_{rx_k} - \phi_{tx_k} + \phi_{rx_0} + \phi_{tx_0},$$

que é armazenada para uso posterior durante o estágio de operação.

Estágio de operação:

TS_0 : O interrogador transmite um sinal piloto que é recebido por cada sensor S_i como $rx_i(t) = \cos(\omega t - \theta_i - \phi_{rx_i})$. Esse sinal tem sua fase invertida por S_i , gerando

$$rx_{i_inv}(t) = \cos(\omega t + \theta_i + \phi_{rx_i}).$$

TS_1 : Os sensores S_i fazem $tx_i(t) = rx_{i_inv}(t)e^{j(\Psi_i)}$ (com $\Psi_0 = 0$) e transmitem simultaneamente

$$tx_0(t) = \cos(\omega t + \theta_0 + \phi_{rx_0})$$

$$tx_i(t) = \cos(\omega t + \theta_i - \phi_{tx_i} + \phi_{rx_0} + \phi_{tx_0})$$

para ($i = 1..N - 1$), gerando (6) no ponto de recepção (interrogador).

Para uma mesma campanha, o primeiro ciclo do método IRF ocupa o mesmo número de *time-slots* que o método RT, mas a partir do segundo ciclo o número de *time-slots* por ciclo se mantém constante = 2, independente do número de sensores, conforme ilustrado na Fig. 3, para um sistema hipotético composto de 3 sensores e 1 interrogador.

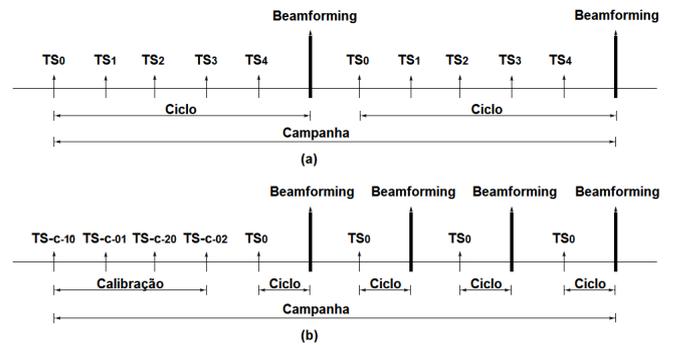


Fig. 3. Relação temporal: (a) Round-trip, (b) Inversão Remota de Fase.

III. IMPLEMENTAÇÃO

Para validação experimental do método proposto, um arca-bouço em GNU Radio foi construído e utilizado na implementação dos receptores, descrita a seguir.

A. Implementação dos Receptores

A implementação dos receptores para o método proposto (IRF), pode ser representada pelo diagrama (*flowgraph*) mostrado na Fig. 4, cuja dinâmica é capturada pelo fluxograma mostrado na Fig. 5. Durante o estágio de calibração, a fase do sinal de retorno do mestre é capturada pelo bloco tracejado na Fig. 4. Posteriormente, durante o estágio de operação, o sinal piloto enviado pelo interrogador tem sua fase invertida pelo bloco denominado *Phase_inverter* e sofre uma rotação adicional de fase pelo bloco *Phase_rotator*.

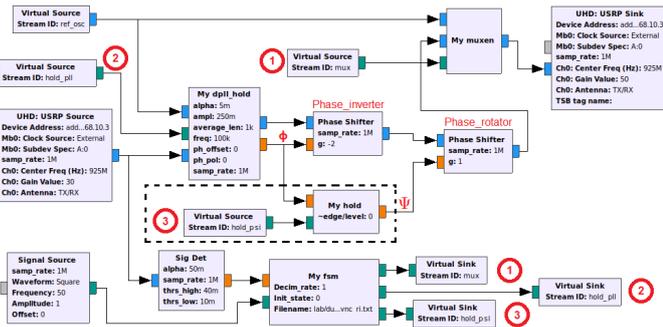


Fig. 4. Diagrama esquemático da implementação dos receptores em GNU-Radio para o método proposto. A parte tracejada é mantida inativa no modo Mestre.

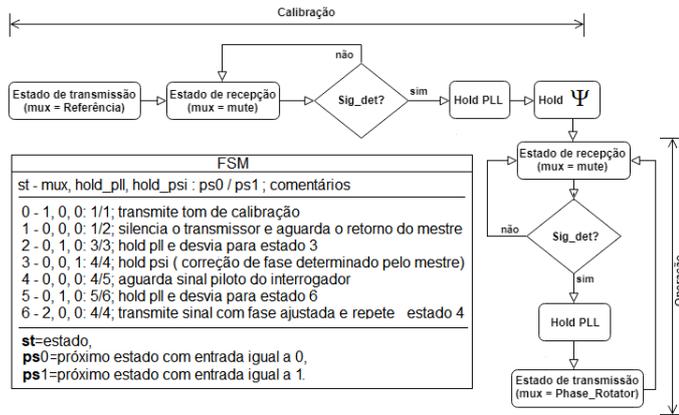


Fig. 5. Fluxograma representando a dinâmica do algoritmo IRF. O Mestre ignora o estágio de Calibração (estados de 0 a 3), iniciando diretamente no estágio de Operação (estados de 4 a 6).

B. Arcabouço

GNU Radio [14] é uma ferramenta de software gratuita e de código aberto, composta de um conjunto de funções de processamento de sinais, utilizada na implementação de Rádios Definidos por Software.

O arcabouço utilizado neste trabalho é uma coleção de blocos OOT (*Out-of-Tree*), terminologia GNU Radio para funções de processamento de sinais definidas e criadas pelo usuário, suficientemente flexível para uso em diferentes projetos. Alguns desses blocos estão mostrados Fig. 6.

Nesse trabalho em particular, o principal elemento é o bloco *My fsm* que implementa um controlador sequencial baseado em Máquina de Estados Finitos do tipo Moore. Sua função

principal é controlar a sequência de operações, ativando e desativando os outros blocos de forma controlada. A sequência de operações é especificada por uma tabela de transição de estados codificada em um arquivo texto que é lido e interpretado pelo bloco no início da execução do sistema.

Segue uma descrição dos principais blocos:

- **Sig Det:** Detector simples de sinal baseado no cálculo da média móvel do módulo do sinal de entrada, guiado por um limiar parametrizado ('*threshold*').
- **My hold:** Controle 'Sample and Hold', que memoriza e mantém em sua saída o valor presente na entrada quando da transição negativa do sinal 'hold'.
- **My muxen:** Seletor, direcionando para a saída o sinal da entrada 1 quando o sinal 'sel' for igual a 1 ou o sinal da entrada 2 quando 'sel' for igual a 2 ou 'zero' (mute) quando 'sel' for igual a zero.
- **Phase Shifter:** Implementa, de forma contínua, um deslocador de fase calculando $out = in * e^{j(g*phase)}$
- **My dpll_hold:** PLL digital que rastreia a frequência e a fase do sinal de entrada e o apresenta na saída quando o sinal 'hold' for igual a 1 e interrompe o rastreamento da entrada mantendo, na saída, um sinal com frequência e fase iguais ao do sinal de entrada quando da transição negativa do sinal 'hold'. Como conveniência, também fornece a diferença de fase entre o sinal de entrada e um sinal local de referência.
- **My fsm:** Controlador, cujas saídas são ativadas e desativadas, sequencialmente, de forma programável.

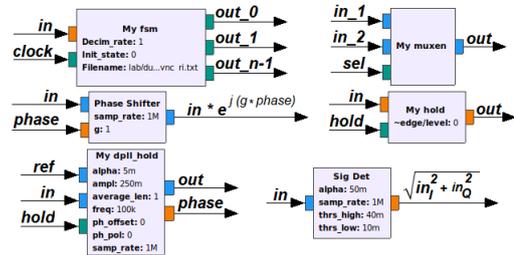


Fig. 6. Blocos básicos integrantes do arcabouço.

IV. RESULTADOS

Os experimentos foram realizados utilizando a plataforma de hardware USRP-N2x0 com *front-end* de RF RFX900 [15], operando na frequência de $f_{RF} = 925$ MHz. A taxa de amostragem foi programada para 1 Msamples/s e o sinal de treinamento (piloto) foi gerado com $f_1 = 100$ kHz.

Um modelo simplificado de rede com 1 Interrogador e 2 Sensores, representado pela parte hachureada da Fig. 1 foi considerado, onde as distâncias entre o Interrogador D e os sensores S_0 e S_1 são 3,4 m e 3,2 m, respectivamente. A distância entre os sensores é de 60 cm.

Os dois métodos (*RT* e *IRF*) apresentam resultados semelhantes, quando se compara as amplitudes de (1) e (6), como se pode ver na Fig. 7 onde ambos apresentam ganhos de *beamforming* em torno de 93% do ganho ideal, em uma rede composta de 2 sensores. Entretanto o método *RT* apresenta maior oscilação no ganho ao longo do tempo, o que pode

ser parcialmente explicado, entre outros fatores, pelo maior número de estimativas de fase inerente ao protocolo, com maior erro acumulado.

Interrompendo a repetição de ciclos e mantendo constante o estado de *beamforming*, a manutenção da curva do IRF próximo ao ideal, visto na Fig. 8, evidencia que o método IRF suporta um intervalo de tempo maior entre sequências de treinamento (sinal piloto), o que pode representar menor consumo de energia para uma taxa de ciclos igual ao método RT.

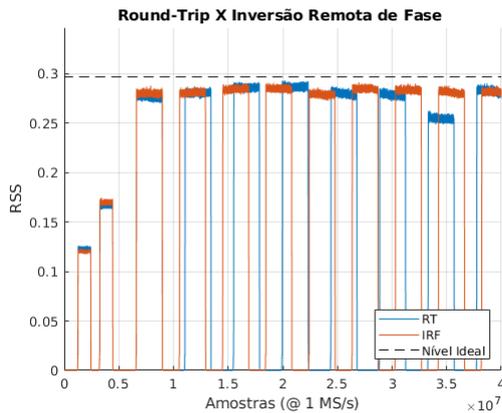


Fig. 7. Comparação entre os níveis de sinal dos dois métodos. No início da campanha de testes, os sensores S_0 e S_1 são ligados e desligados individualmente e, então, o beamforming é ativado ciclicamente.

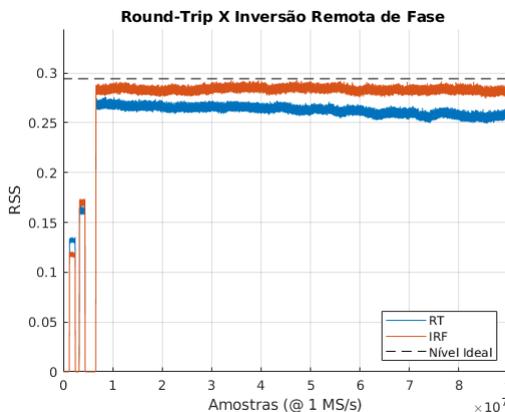


Fig. 8. Efeito do erro acumulado de fase no protocolo RT.

Uma comparação preliminar entre os métodos pode ser vista na Tabela I onde destacamos a menor latência apresentada pelo método IRF bem como complexidade mais baixa para implementação.

TABELA I

COMPARAÇÃO ENTRE O MÉTODO TIME-SLOTTED ROUND-TRIP (RT) E O MÉTODO PROPOSTO, INVERSÃO REMOTA DE FASE (IRF)

Método	Time Slots	Latência	Complexidade
RT	$2 N^*$	Cresce com N^*	Referência
IRF	2^{**}	Baixa, constante**	Mais baixa

*Número de elementos sensores. **A partir do segundo ciclo.

V. CONCLUSÕES

Neste trabalho foi apresentado um método de *beamforming* distribuído e um arcabouço baseado em GNU Radio para validação experimental. O arcabouço foi utilizado na implementação de uma prova de conceito de dois métodos de *beamforming* distribuído: *Time-Slotted Round-Trip Synchronization*, utilizado como referência e um método proposto denominado “Inversão Remota de Fase”. Os resultados demonstram que o método proposto apresenta um melhor desempenho em termos de estabilidade de ganho bem como apresenta menor latência quando comparado com o método de referência. O próximo passo é investigar o sincronismo distribuído de frequência permitindo a implementação de um sistema completo de *beamforming* distribuído.

REFERÊNCIAS

- [1] H. Ochiai, P. Mitran, H. Poor, and V. Tarokh, “Collaborative beamforming for distributed wireless ad hoc sensor networks,” IEEE Transactions on Signal Processing], vol. 53, no. 11, pp. 4110–4124, Nov. 2005.
- [2] R. Mudumbai, D.R. Brown III, U. Madhow, and H.V. Poor, “Distributed transmit beamforming: Challenges and recent progress,” IEEE Communications Magazine, vol. 47, no. 2, pp. 102–110, February 2009.
- [3] S. Jayaprakasam, S. K. A. Rahim and C. Y. Leow, “Distributed and Collaborative Beamforming in Wireless Sensor Networks: Classifications, Trends, and Research Directions,” in IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2092–2116, 2017
- [4] D. R. Brown III, G. B. Prince, and J. A. McNeill, “A method for carrier frequency and phase synchronization of two autonomous cooperative transmitters,” in IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Jun. 2005, pp. 260 – 264.
- [5] D.R. Brown III and H.V. Poor, “Time-slotted round-trip carrier synchronization for distributed beamforming,” IEEE Trans. on Signal Processing, vol. 56, no. 11, pp. 5630–5643, November 2008.
- [6] R. Mudumbai, J. Hespanha, U. Madhow, and G. Barriac, “Scalable feedback control for distributed beamforming in sensor networks,” Information Theory, 2005. ISIT 2005. Proceedings. International Symposium on, pp. 137–141, 4–9 Sept. 2005.
- [7] R. Mudumbai, B. Wild, U. Madhow, and K. Ramchandran, “Distributed Beamforming using 1 Bit Feedback: from Concept to Realization,” Proc. of 44th Allerton Conference on Communication Control and Computing, Sept. 2006.
- [8] M. M. U. Rahman, H. E. Baidoo-Williams, S. Dasgupta, and R. Mudumbai, “Fully wireless implementation of distributed beamforming on a software-defined radio platform,” in the 11th ACM/IEEE Conference on Information Processing in Sensor Networks (IPSN 2012), 2012.
- [9] D. R. Brown, B. Zhang, B. Svirchuk and M. Ni, “An experimental study of acoustic distributed beamforming using round-trip carrier synchronization,” 2010 IEEE International Symposium on Phased Array Systems and Technology, Waltham, MA, 2010, pp. 316-323.
- [10] Suleiman, Umar Dauda, “A Review On Frequency Synchronization in Collaborative Beamforming: A Practical Approach”, Penerbit Akademia Baru, 2017.
- [11] S. Mghabghab, H. Ouassal and J. A. Nanzer, “Wireless Frequency Synchronization for Coherent Distributed Antenna Arrays,” 2019 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting, Atlanta, GA, USA, 2019, pp. 1575-1576.
- [12] R. L. Schmid, T. M. Comberiate, J. E. Hodkin and J. A. Nanzer, “A Distributed RF Transmitter Using One-Way Wireless Clock Transfer,” in IEEE Microwave and Wireless Components Letters, vol. 27, no. 2, pp. 195-197, Feb. 2017.
- [13] Ni, Min, “An Optimized Software-Defined-Radio Implementation of Time-Slotted Carrier Synchronization for Distributed Beamforming”, MSc thesis, WORCESTER POLY-TECHNIC INSTITUTE, 2010.
- [14] GNU radio, <http://gnuradio.org>, 2020.
- [15] USRP products, <http://www.ettus.com/products>, 2020.