

Ước lượng kênh truyền trong hệ thống đa robot sử dụng SDR

Đỗ Hải Sơn*, Nguyễn Hữu Hưng†, Phạm Duy Hưng†, Trần Thị Thúy Quỳnh†

* Viện Công nghệ Thông tin - Đại học Quốc gia Hà Nội

† Trường Đại học Công nghệ - Đại học Quốc gia Hà Nội

Email: dohaison1998@vnu.edu.vn, nguyễnhuuhung032@gmail.com, {hungpd, quynhttt}@vnu.edu.vn

Tóm tắt—Nghiên cứu này tập trung xây dựng một hệ thống thực nghiệm ước lượng kênh truyền phục vụ truyền thông trong hệ thống đa robot di động sử dụng các thiết bị vô tuyến định nghĩa bằng phần mềm (SDR - Software Define Radio). Hệ thống gồm 2 robot di động được lập trình cho 2 tình huống robot đứng yên và robot di động theo quỹ đạo cho trước. Hệ thống truyền thông được thực hiện thông qua điều chế đa sóng mang trực giao OFDM (Orthogonal Frequency Division Multiplexing) nhằm làm giảm ảnh hưởng của môi trường đa đường trong nhà. Hiệu năng của hệ thống được đánh giá thông qua tỷ lệ lỗi bit BER (Bit Error Rate). Các kết nối liên quan đến chuyển động của robot và phần truyền thông được thực hiện tương ứng thông qua Raspberry Pi 3 và BladeRF x115. Kỹ thuật bình phương tối thiểu LS (Least Squares) được thực hiện để ước lượng kênh với tỷ lệ lỗi bit khoảng 10^{-2} .

Từ khóa—Ước lượng kênh, Robot di động, Vô tuyến định nghĩa mềm, GNU Radio.

I. GIỚI THIỆU

Mạng truyền thông trong các hệ thống đa robot đang là chủ đề được quan tâm hiện nay [1] do gắn liền với cách mạng công nghiệp 4.0. Trong các mạng này, mỗi robot thường yêu cầu có các bộ thu phát vô tuyến để giao tiếp với các robot khác trong mạng hoặc máy chủ điều khiển từ bên ngoài. Tuy nhiên, trong các điều kiện môi trường phức tạp [2], các robot cỡ nhỏ sẽ gặp phải khó khăn khi phải ước lượng kênh truyền vô tuyến để thiết lập tuyến truyền chính xác. Một số phương pháp ước lượng kênh truyền cho các robot cỡ nhỏ đã được đề cập trong [3], [4]. Tuy nhiên, các nghiên cứu thực nghiệm về vấn đề này còn khá mới hoặc các sản phẩm thương mại thường sử dụng các bo mạch có sẵn, khó can thiệp.

Trong những năm gần đây, các thiết bị vô tuyến định nghĩa bằng phần mềm SDR (Software Define Radio) được sử dụng nhiều cho mục đích kiểm nghiệm các thuật toán trong viễn thông [5], do tính linh hoạt và đa dạng trong các thư viện xử lý tín hiệu có sẵn trên phần mềm GNU Radio [6]. Từ kết quả trong nghiên cứu trước đây của nhóm [7], các thiết bị SDR có khả năng triển khai trên các robot cỡ nhỏ với nguồn điện độc lập được cấp từ pin ngay trên robot. Trong nghiên cứu này, một hệ thống thử nghiệm gồm hai robot cỡ nhỏ được điều khiển bằng Raspberry Pi 3 [8], mỗi robot lắp một thiết bị SDR (BladeRF x115 [9]) để thực hiện việc thu phát và ước lượng kênh truyền thời gian thực. Hệ thu phát sử dụng điều chế ghép kênh phân chia theo tần số trực giao OFDM (Orthogonal Frequency-division Multiplexing) với mã nguồn mở có tên SDR4All tương tự như chuẩn WiFi 802.11 được

phát triển bởi [5], [10], [11]. Mã nguồn mở này sử dụng bộ cân bằng ép về không ZF (Zero Forcing) để ước lượng kênh truyền vô tuyến. Trong bài báo này, nhóm nghiên cứu không thay đổi thuật toán bình phương tối thiểu LS (Least Squares) của SDR4All mà tập trung vào xem xét hiệu năng của giải thuật này trên các robot trong các điều kiện khác nhau. Hiệu năng của hệ thống được đánh giá thông qua tỷ lệ lỗi bit BER (Bit Error Rate).

Các đóng góp chính của bài báo gồm: xây dựng các khái niệm trên GNU Radio dùng để mô phỏng kênh truyền vô tuyến và tính hệ số BER; đánh giá chất lượng của ước lượng kênh truyền trong hệ thống đa robot thông qua mô phỏng và kiểm nghiệm thực tế.

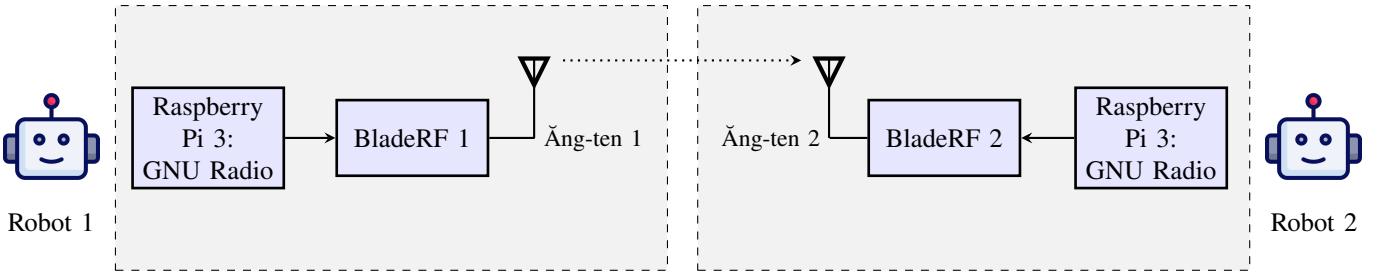
Bài báo được tổ chức như sau: phần II trình bày về mô hình hệ thống, sơ lược về lý thuyết sử dụng cho ước lượng kênh truyền trong bộ công cụ SDR4All. Kết quả mô phỏng và thực nghiệm được biểu diễn trong phần III. Một vài thảo luận về hiệu năng của hệ thống sau khi thực nghiệm được trình bày trong phần IV. Kết luận của bài báo được đưa ra trong phần V.

II. ƯỚC LƯỢNG KÊNH TRUYỀN TRONG SDR4ALL

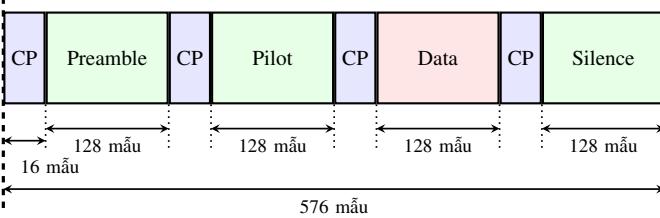
Phần này trình bày về mô hình thu phát sử dụng mã nguồn SDR4ALL, bao gồm cấu trúc của các khung OFDM và giải thuật của bộ cân bằng ZF.

A. Mô hình hệ thống

Xét mô hình hệ thống gồm hai robot đơn ăng-ten thu phát như trên hình 1 sử dụng điều chế OFDM theo mã nguồn SDR4All. Trong đó, mỗi robot bao gồm các thành phần điều khiển, chuyển động, một máy tính cỡ nhỏ là Raspberry Pi 3 và một BladeRF x115. Raspberry Pi chịu trách nhiệm điều khiển robot và chạy GNU Radio để gửi và nhận dữ liệu từ BladeRF x115. Tất cả các linh kiện trên robot di động đều được cấp nguồn bằng một pin gắn ngoài ngay trên robot. Lý thuyết về việc truyền nhận OFDM đã được biết đến rộng rãi [12] nên trong bài báo này sẽ không nhắc lại mà tập trung vào giới thiệu tổ chức khung OFDM của mã nguồn SDR4All như trên hình 2 [13]. Khung dữ liệu OFDM được tạo ra bao gồm 4 thành phần chính: Preamble, Pilot, Data, và Silence, được truyền liên tục theo thời gian. Preamble, Pilot, và Silence là các thành phần không mang thông tin, có tác dụng lần lượt là đồng bộ hệ thống phát, ước lượng kênh truyền, và ước lượng



Hình 1: Mô hình hệ thống thực nghiệm ước lượng kênh truyền cho robot sử dụng SDR.



Hình 2: Minh họa cấu trúc một khung OFDM của mã nguồn SDR4All.

tỷ số SNR. Biểu diễn của Preamble tại thời điểm n như sau:

$$\mathbf{g} = [g_1[n], g_2[n], \dots, g_{N_{OCC}}[n]]^\top, \quad (1)$$

với $\mathbf{g} \in \mathbb{C}^{N_{OCC} \times 1}$ và N_{OCC} là số sóng mang con được sử dụng [14]. Các giá trị $g[n]$ được định nghĩa trong [15]. Thành phần Pilot được chèn vào khung OFDM theo kiểu Block-type [16] tức được chèn vào toàn bộ các tần số ở các khe thời gian xác định. Các thành phần Pilot, Data, và Silence lần lượt có kích thước $\mathbf{p} \in \mathbb{C}^{N_{OCC} \times 1}$, $\mathbf{D} \in \mathbb{C}^{N_{OCC} \times N_d}$, $\mathbf{z} \in \mathbb{C}^{N_{OCC} \times 1}$, với N_d là độ dài véc-tơ dữ liệu truyền đi, và được biểu diễn như sau:

$$\mathbf{p} = [p_1[n], p_2[n], \dots, p_{N_{OCC}}[n]]^\top. \quad (2)$$

$$\mathbf{D} = \begin{bmatrix} d_1[n] & \dots & d_1[n + N_d - 1] \\ d_2[n] & \dots & d_2[n + N_d - 1] \\ \vdots & \ddots & \vdots \\ d_{N_{OCC}}[n] & \dots & d_{N_{OCC}}[n + N_d - 1] \end{bmatrix}. \quad (3)$$

$$\mathbf{z} = [z_1[n], z_2[n], \dots, z_{N_{OCC}}[n]]^\top. \quad (4)$$

Một khung OFDM $\mathbf{S} \in \mathbb{C}^{N_{OCC} \times (3+N_d)}$ chỉ chứa 4 thành phần kể trên được biểu diễn như dưới đây [11]:

$$\mathbf{S} = [\mathbf{g} | \mathbf{p} | \mathbf{D} | \mathbf{z}], \quad (5)$$

Để đơn giản hóa ký hiệu, các sóng mang con trống (N_l) ở hai phía trái phải của phổ chưa được đề cập. Trên thực tế, $2N_l \times (3 + N_d)$ sóng mang con sẽ được chèn vào để tạo thành khung OFDM có kích thước $(N_{OCC} + 2N_l) \times (3 + N_d)$. Sau các khôi IFFT và chuẩn hóa năng lượng, cả bốn thành phần kể trên sẽ được chèn thêm tiền tố vòng CP (Cyclic prefix) bằng cách sao chép N_{CP} mẫu cuối cùng mỗi luồng song song rồi

chèn lên đầu để tạo thành khoảng bảo vệ. Kích thước cuối cùng của khung OFDM được robot truyền đi như trên hình 2 là $(N_{OCC} + 2N_l + N_{CP}) \times (3 + N_d)$.

B. Bộ cân bằng kênh Zero Forcing

Ở phía robot thu, sau khi qua các bước như đồng bộ, chuyển đổi nối tiếp thành song song, loại bỏ CP, và FFT sẽ tách dữ liệu nhận thành dạng các khung OFDM như bên phát. Chi tiết các bước này có thể xem tại [11], [13]. Ma trận của khung OFDM nhận được bên thu là:

$$\hat{\mathbf{S}} = [\hat{\mathbf{g}} | \hat{\mathbf{p}} | \hat{\mathbf{D}} | \hat{\mathbf{z}}]. \quad (6)$$

Dựa vào thành phần $\hat{\mathbf{p}}$, bộ cân bằng ZF ước lượng ảnh hưởng của kênh truyền đến các ký hiệu pilot biết trước qua đó đảo ngược quá trình này để khôi phục các tín hiệu dữ liệu \mathbf{D} . Ma trận kênh cân bằng ZF $\hat{\mathbf{H}}_{est} \in \mathbb{C}^{N_{OCC} \times N_{OCC}}$ được ước lượng bởi:

$$\hat{\mathbf{H}}_{est} = \text{diag} \left(\left[\frac{1}{\hat{h}_{N_l}}, \frac{1}{\hat{h}_{N_l+1}}, \dots, \frac{1}{\hat{h}_{N_l+N_{OCC}-1}} \right] \right), \quad (7)$$

với

$$\hat{h}[k] = \frac{\hat{p}[k]}{p[k]}, \quad N_l \leq k \leq N_l + N_{OCC} - 1, \quad (8)$$

Dữ liệu (các sóng mang con chứa thông tin) được khôi phục như sau:

$$\hat{\mathbf{D}}_{est} = \hat{\mathbf{H}}_{est} \hat{\mathbf{D}}. \quad (9)$$

Ngoài ước lượng tín hiệu bên phát, tỷ số SNR cũng được tính toán sử dụng thành phần Preamble và Silence như trong (10).

$$\hat{\text{SNR}} = 10 \log \left(\frac{\sum_{k=N_l}^{k=N_l+N_{OCC}-1} |\hat{g}_k|^2}{\sum_{k=N_l}^{k=N_l+N_{OCC}-1} |\hat{z}_k|^2} \right). \quad (10)$$

Các bước còn lại như chuyển đổi luồng song song thành nối tiếp và giải ánh xạ chòm sao để khôi phục các bit gửi đi xem tại [13].

III. KẾT QUẢ MÔ PHỎNG VÀ THỰC NGHIỆM

Phần này sẽ trình bày về các bước thiết lập mô phỏng và thực nghiệm. Phân tích kết quả được đưa ra cùng với đánh giá về hiệu năng của hệ thống trong các kịch bản thử nghiệm khác nhau.

A. Kết quả mô phỏng

1) Thiết lập mô phỏng: Từ lý thuyết đã trình bày ở mục II, trước hết, một mô phỏng trên Matlab được triển khai dựa trên [12]. Trong đó, kênh truyền (thành phần đa đường) sẽ được mô hình hóa dưới dạng một bộ lọc đáp ứng xung hữu hạn FIR (Finite Impulse Response). Các hệ số của bộ lọc FIR được lựa chọn ngẫu nhiên, trong nghiên cứu này, chúng tôi chọn một bộ lọc FIR có độ dài bằng 2 như sau:

$$\mathbf{h} \equiv \text{FIR}([0, 8 + 0, 9i; 0, 6 + 0, 7i]). \quad (11)$$

Đối với mô phỏng trên GNU Radio, bài báo này sử dụng phiên bản GNU Radio 3.7.11¹. Mã nguồn SDR4All được sử dụng để xây dựng một bộ thu phát sử dụng OFDM hoàn chỉnh. Ảnh hưởng của kênh truyền được thực hiện thông qua khối “FIR Channel” do nhóm nghiên cứu xây dựng.

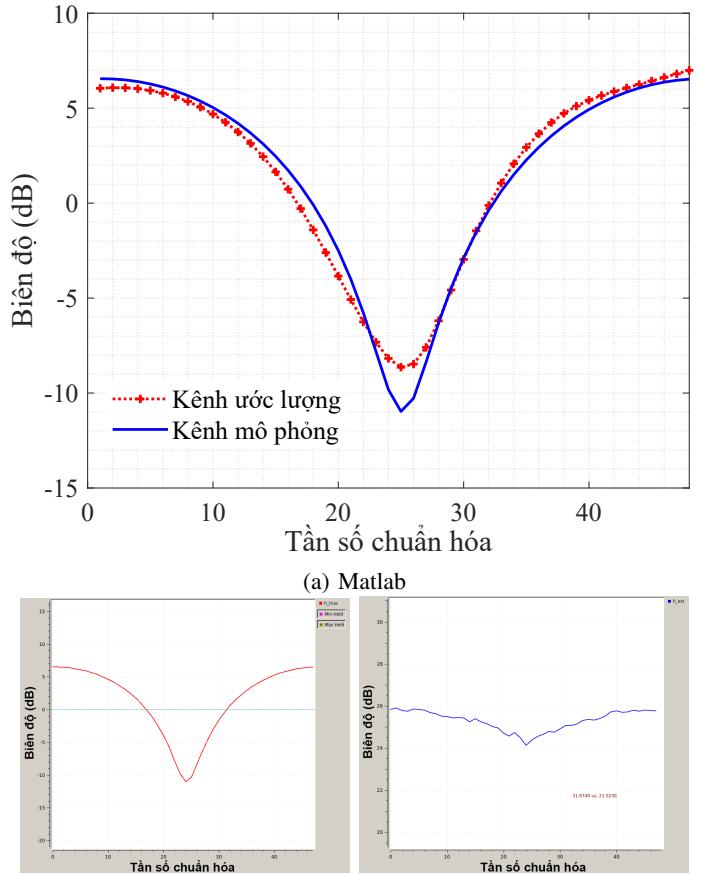
2) Kết quả: Hình 3 biểu diễn đáp ứng tần số của kênh truyền đã cho và kênh truyền ước lượng được với 3a sử dụng Matlab và 3b sử dụng GNURadio. Kênh truyền được mô phỏng theo công thức (11) có đáp ứng biên độ tại các tần số chuẩn hóa từ 20 đến 30 (tương ứng là sóng mang con thứ 20 đến 30) suy hao đáng kể và được bộ cân bằng ZF ước lượng khá chính xác (hình 3a). Với GNU Radio và SDR4All như trên hình 3b, ở SNR lớn, khoảng 30 dB, về hình dạng của đáp ứng kênh truyền vẫn được đảm bảo nhưng biên độ có sự sai khác khá lớn. Điều này xảy ra do bản thân mã nguồn SDR4All đã thêm khôi chuẩn hóa năng lượng (“Power Scaling”) trước khi thực hiện ước lượng kênh truyền.

Hình 4 biểu diễn chuỗi dữ liệu thu được trong mô phỏng dùng GNU Radio. Kết quả khá chính xác và BER được tính toán và hiển thị sau mỗi 100.000 bit.

Hiệu năng của hệ thống được xem xét thông qua BER theo các mức SNR như trên hình 5. Để nhận thấy, với mô phỏng Matlab dựa trên lý thuyết, đường BER có thể xuất phát từ các mức SNR rất thấp (-10 dB) và giảm nhỏ hơn 10^{-3} tại SNR ngưỡng 16 dB. Mô phỏng GNU Radio sử dụng SDR4All gần với thực tế hơn, phải đến khi SNR lớn hơn 2 dB, bên thu mới nhận dạng và tách được tín hiệu của bên phát khỏi nền nhiễu. Xét về độ chính xác, rõ ràng SDR4All thấp hơn đáng kể so với lý thuyết và chỉ đạt tiệm cận $\text{BER} \approx 10^{-2}$ tại các ngưỡng SNR lớn hơn 20 dB. Cần lưu ý rằng, các giá trị BER thu được từ SDR4All khá cao do chưa bao gồm các khôi mã sửa lỗi.

B. Kết quả thực nghiệm

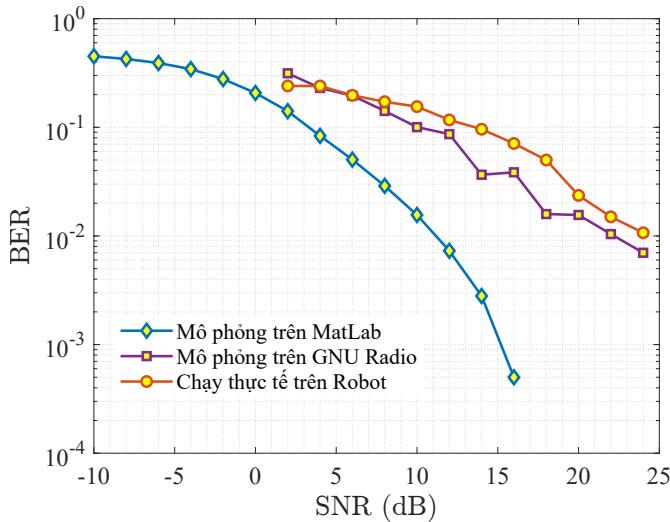
1) Thiết lập thí nghiệm: Hai robot cỡ nhỏ sử dụng BladeRF để truyền tín hiệu được thiết lập như trên hình 6. Các thông số chạy thực nghiệm được cho trên bảng I. Địa điểm thực nghiệm là tầng 7 nhà E3, Trường Đại học Công nghệ - DHQGHN, bố trí hệ thống ở ba vị trí khác nhau để đưa ra kết quả trung bình của việc ước lượng kênh truyền. Các giá trị BER vẫn được tính trung bình và thu thập sau mỗi 100.000 bit được giải điều chế.



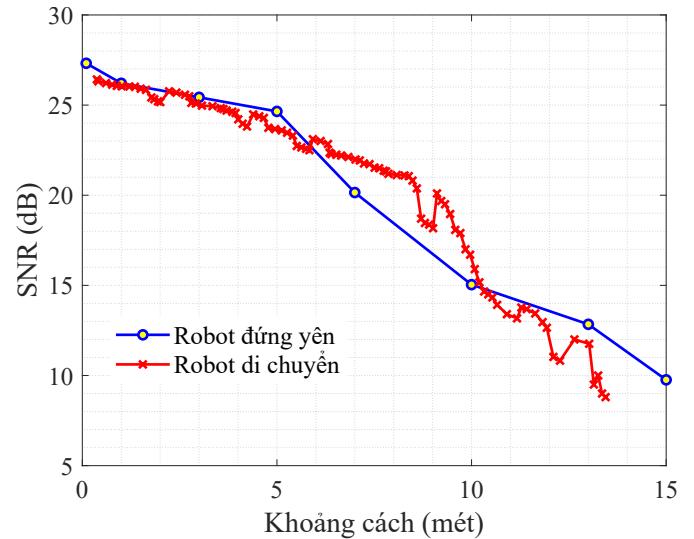
(b) GNU Radio: (trái) kênh mô phỏng và (phải) kênh ước lượng
 Hình 3: Kết quả mô phỏng ước lượng kênh truyền trên (a) Matlab và (b) GNU Radio.

Hình 4: Chuỗi nhận được tại bên thu và tỷ số BER sau mỗi 100.000 bit thu được

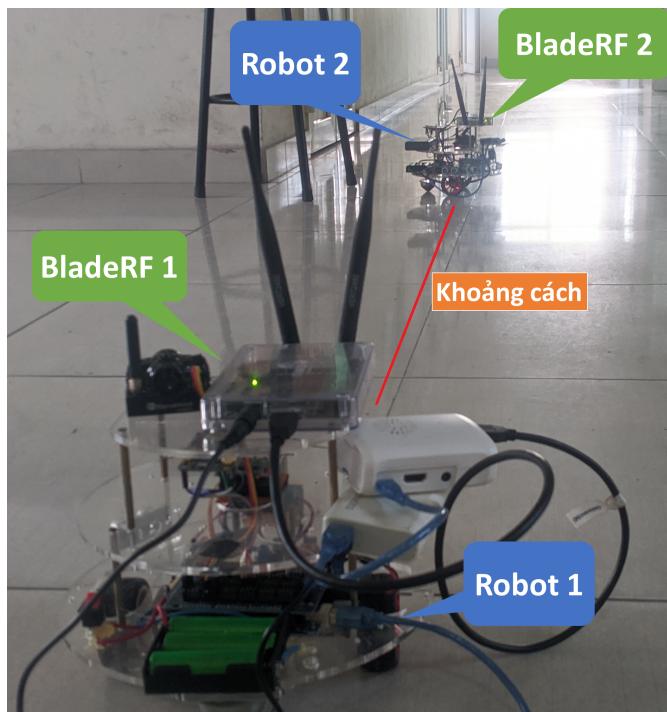
¹https://github.com/DoHaiSon/SDR_NC/blob/master/Documents/Readme.pdf



Hình 5: BER thu được thông qua mô phỏng Matlab, GNU Radio, và chạy thực tế trên BladeRF.



Hình 7: Sự thay đổi của SNR theo khoảng cách giữa hai robot.



Hình 6: Hình ảnh hệ thống thực nghiệm gồm hai robot và hai BladeRF.

Bảng I: Các tham số của hệ thống thực nghiệm ước lượng kênh truyền cho robot sử dụng SDR

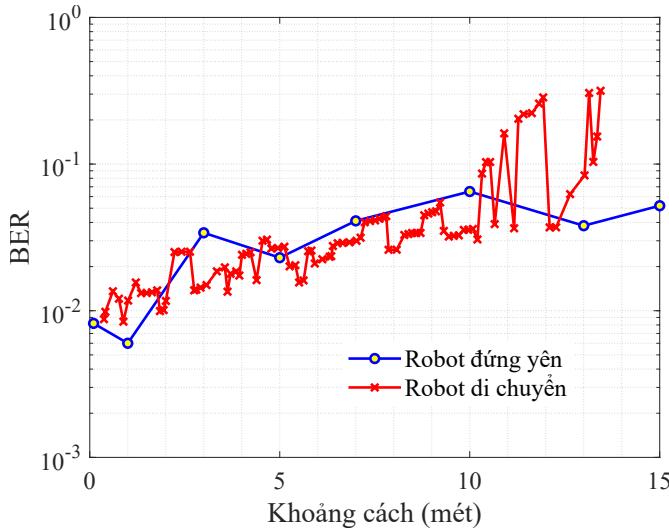
Thông số	Giá trị
Tần số sóng mang	$f_c = 2,515 \text{ GHz}$
Loại điều chế	BPSK (Binary phase shift keying)
Chuỗi dữ liệu truyền đi	[1, 2, ..., 20]
Độ dài FFT	$N_{fft} = 128$
Số sóng mang con được sử dụng	$N_{Occ} = 48$
Kiểu chèn Pilot	Block-type
Độ dài tiền tố vòng	$N_{CP} = 16$

2) Kết quả: Độ chính xác của việc ước lượng kênh truyền khi sử dụng các thiết bị BladeRF thực so với mô phỏng trên Matlab và GNU Radio được biểu diễn trên hình 5 trong trường hợp hai robot đứng yên không có vật cản ở giữa. Cả ba trường hợp mô phỏng và thực nghiệm đều cho BER giảm khi SNR tăng lên. Tuy nhiên, BER trong trường hợp thực nghiệm thực tế lớn hơn so với mô phỏng Matlab và GNU Radio do sự phức tạp của kênh truyền thực. Tương tự như GNU Radio, các BladeRF cũng yêu cầu SNR đạt đến một ngưỡng SNR lớn nhất định để phát hiện và tách được tín hiệu nguồn khỏi nhiễu, theo thực nghiệm, ở khoảng 2-4 dB.

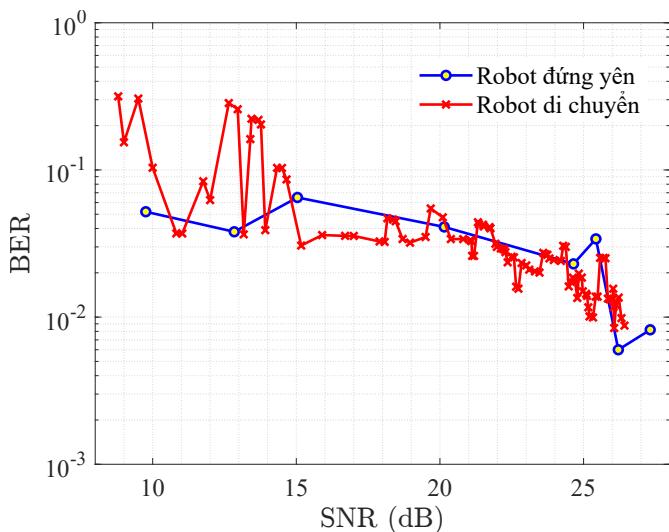
Tiếp đến, sự phụ thuộc của SNR vào khoảng cách giữa hai robot sẽ được xem xét trong hai trường hợp: (i) robot đứng yên tại 8 khoảng cách khác nhau (0, 1, 3, 5, 7, 10, 13, và 15 mét) và (ii) robot di chuyển với vận tốc 0,3 m/s. Việc thu thập dữ liệu được thực hiện liên tục. Kết quả thu được như trên hình 7 cho thấy sự suy giảm nhanh chóng của giá trị SNR khi khoảng cách giữa hai robot tăng dần. Độ dốc của đường SNR ở cả hai trạng thái đều khá giống nhau, dù vẫn có những sự không ổn định nhất định với dữ liệu được thu thập khi robot di chuyển. Đặc biệt, khi khoảng cách lớn hơn 13 mét, robot thu không nhận được giá trị SNR hay robot thu phát hiện ra tín hiệu bên phát ẩn trong nền nhiễu.

Kết quả thu được trên hình 8 cho thấy BER tăng dần khi khoảng cách giữa hai robot tăng. Ở trạng thái robot tĩnh, với khoảng cách 1 mét và 15 mét, BER đạt $6 * 10^{-3}$ và $5 * 10^{-2}$ tương ứng. Đây là chênh lệch lớn, tỷ lệ lỗi bit tăng lên hơn 8 lần khi khoảng cách tăng từ 1 lên 15 mét. Khi robot di chuyển, ở các khoảng cách nhỏ hơn 10 mét, BER vẫn bám sát với sai số khi robot tĩnh. Tuy nhiên với các khoảng cách lớn hơn, độ không ổn định tăng, dẫn đến BER có thể tăng đến 0,3 hay thậm chí không thể nhận dạng tín hiệu phát ở khoảng cách 13 mét trở lên.

Hình 9 biểu diễn sự phụ thuộc của BER vào SNR. Có thể thấy rằng, với mã nguồn SDR4All, tỷ lệ SNR nên ở ngưỡng



Hình 8: Sự thay đổi của BER theo khoảng cách giữa hai robot.



Hình 9: Sự thay đổi của BER theo tỷ lệ SNR.

10 dB trở lên để hệ thống có thể hoạt động. Kết quả thực nghiệm này có thể gợi ý cho việc lựa chọn dải truyền thông an toàn trong hệ thống đa robot và các ngưỡng tham chiếu cho các nghiên cứu tiếp theo về điều chế thích nghi để cải thiện hiệu suất của mã nguồn SDR4All.

IV. THẢO LUẬN

Trong nghiên cứu này, điểm hạn chế cần được thảo luận đó là tỷ lệ lỗi bit khi mô phỏng trên GNU Radio cũng như chạy thực nghiệm còn cách khá xa so với mô phỏng trên Matlab. Sai số này bản thân xuất phát từ mã nguồn SDR4All. Cụ thể, như trong hình 3b, do việc cân bằng tín hiệu diễn ra sau khi tín hiệu đã được chuẩn hóa năng lượng, dẫn đến sai số trong việc ước lượng lượng kênh truyền. Để tỷ số BER giảm, trong nghiên cứu [10] của chúng tôi, hai hướng giải quyết được đưa ra để đảm bảo hệ thống hoạt động tốt hơn trong các nhiệm vụ yêu cầu độ chính xác cao hơn như truyền hình ảnh: (i),

thêm các khối mã kênh; (ii), chia nhỏ bản tin thành các gói tin (packet) nhỏ hơn, thêm cơ chế kiểm tra lỗi cho từng gói tin, bổ sung cơ chế phản hồi cho bên thu để thông báo cho bên phát truyền lại các gói tin bị lỗi.

V. KẾT LUẬN

Nhóm nghiên cứu đã xây dựng mô hình thực nghiệm hệ thống truyền thông đa robot sử dụng SDR. Một số kịch bản mô phỏng và thực nghiệm được đưa ra để kiểm chứng khả năng ước lượng kênh truyền của hai robot khi đứng yên và di chuyển liên tục. Kết quả được đưa ra trong bài báo chứng minh tính khả thi của việc ứng dụng các thiết bị SDR vào các nghiên cứu về mạng truyền thông đa robot trong tương lai cũng như bộc lộ một số điểm hạn chế của bộ nhận dạng trong mã nguồn SDR4All. Trong tương lai, nhóm nghiên cứu sẽ xem xét việc tăng thêm số robot hoạt động trong mạng cũng như phát triển thêm các bộ cân bằng kênh khác cho mã nguồn SDR4All phục vụ cho các mục đích nghiên cứu và giảng dạy.

Các lưu đồ trên GNU Radio đã được sử dụng trong bài báo, các khối do nhóm nghiên cứu xây dựng có thể tham khảo tại: https://github.com/DoHaiSon/SDR_NC/tree/BER

LỜI CẢM ƠN

Nghiên cứu này được tiến hành trong khuôn khổ đề tài QG.21.26 “Giải pháp chống, chịu nhiễu duy trì mạng đa robot trong môi trường động” của ĐHQGHN.

TÀI LIỆU THAM KHẢO

- [1] K.-C. Chen, S.-C. Lin, J.-H. Hsiao, C.-H. Liu, A. F. Molisch, and G. P. Fettweis, “Wireless networked multirobot systems in smart factories,” *Proceedings of the IEEE*, vol. 109, no. 4, pp. 468–494, 2021.
- [2] A. Girma, N. Bahadori, M. Sarkar, T. G. Tadewos, M. R. Behnia, M. N. Mahmoud, A. Karimoddini, and A. Homayifar, “IoT-enabled autonomous system collaboration for disaster-area management,” *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 5, pp. 1249–1262, 2020.
- [3] W. Burgard, O. Brock, and C. Stachniss, *An Experimental Study of Exploiting Multipath Fading for Robot Communications*. MIT Press, 2008, pp. 289–296.
- [4] X. Gao, X. Mu, W. Yi, and Y. Liu, “Intelligent trajectory design for ris-noma aided multi-robot communications,” *IEEE Transactions on Wireless Communications*, early access, 2023.
- [5] T. T. T. Quynh, N. K. Hoang, N. Van Ly, N. L. Trung, N. Q. Tuan, E. Bastug, S. Azarian, V. N. Q. Bao, T. X. Nam, M. Debbah, and P. Duhamel, “Network coding with multimedia transmission and cognitive networking: An implementation based on software-defined radio,” *REV Journal on Electronics and Communications*, vol. 10, no. 3–4, 2020.
- [6] GNU Radio. [Online]. Available: www.gnuradio.org
- [7] D. H. Son and T. T. T. Quynh, “Đồng bộ nhiều sdr trong thực thi thuật toán ước lượng hướng sóng đèn music,” in *Hội nghị Quốc gia lần thứ XXIV về Điện tử, Truyền thông và Công nghệ Thông tin*, 2021.
- [8] Raspberry Pi 3 Model B. [Online]. Available: <https://www.raspberrypi.com/products/raspberry-pi-3-model-b/>
- [9] BladeRF x115 - Nuand. [Online]. Available: www.nuand.com
- [10] SDR4All. [Online]. Available: https://github.com/avitech-vnu/SDR_NC/
- [11] E. Baştug, “Study of vandermonde frequency division multiplexing on software defined radio platform,” Ph.D. dissertation, CentraleSupélec, 2012.
- [12] Y. S. Cho, J. Kim, W. Y. Yang, and C. G. Kang, *MIMO-OFDM Wireless Communications with MATLAB*. John Wiley & Sons, 2010.

- [13] T. T. T. Quynh, N. K. Hoang, N. V. Ly, D. H. Son, and N. L. Trung, “Thiết lập nền tảng sdr cho hệ thống ofdm,” in *Truyền thông chuyển tiếp hai chiều: Lý thuyết và Thực nghiệm*, N. L. Trung, V. N. Q. Bảo, and T. T. T. Quỳnh, Eds. Nhà xuất bản Đại học Quốc gia Hà Nội, Jul. 2022, ch. 9, pp. 145–228 (preprint).
- [14] Y. Wu, S. Attallah, and J. W. M. Bergmans, “On the optimality of the null subcarrier placement for blind carrier offset estimation in ofdm systems,” *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 2109–2115, 2009.
- [15] T. Schmidl and D. Cox, “Robust frequency and timing synchronization for ofdm,” *IEEE Transactions on Communications*, vol. 45, no. 12, pp. 1613–1621, 1997.
- [16] A. Ladaycia, A. Mokraoui, K. Abed-Meraim, and A. Belouchrani, “Performance bounds analysis for semi-blind channel estimation in mimo-ofdm communications systems,” *IEEE Transactions on Wireless Communications*, vol. 16, no. 9, pp. 5925–5938, Sept. 2017.