

# SINH DỮ LIỆU KIỂM THỬ CHO MÔ HÌNH HỆ THỐNG NHÚNG SỬ DỤNG KỸ THUẬT KIỂM THỬ THEO CẶP

Đỗ Thị Bích Ngọc

Học Viện Công Nghệ Bưu Chính Viễn Thông

**Tóm tắt:** Hệ thống nhúng đang ngày càng đóng vai trò quan trọng trong đời sống hiện đại. Các hệ thống này đòi hỏi tính an toàn rất cao. Vì vậy, đảm bảo chất lượng cho các hệ thống nhúng này đã và đang thu hút sự quan tâm của cả giới nghiên cứu và công nghiệp. Trong các hệ thống nhúng, quy trình kiểm thử thường yêu cầu độ phủ cao, với nhiều độ đo theo các chuẩn quốc tế, như CC, DC, MC/DC của ISO 26262. Bài báo này đề xuất 1 phương pháp sinh dữ liệu kiểm thử tự động áp dụng kỹ thuật kiểm thử theo cặp nhằm thu được bộ test data với độ phủ cao. Thực nghiệm cho thấy phương pháp đề xuất cho kết quả tốt hơn so với kiểm thử ngẫu nhiên.

**Từ khoá:** Độ phủ CC, độ phủ DC, độ phủ MC/DC, mô hình hệ thống nhúng, kiểm thử, kiểm thử theo cặp, tín hiệu liên tục

## I. GIỚI THIỆU

Hiện nay, hệ thống nhúng đang phát triển mạnh mẽ và ngày càng đóng vai trò quan trọng trong cuộc sống của con người. Hệ thống nhúng bao gồm cả phần cứng và phần mềm hoạt động liên kết với nhau. Nhiều loại hệ thống nhúng có những yêu cầu rất cao về chất lượng, tính ổn định và độ tin cậy. Lý do là lỗi của hệ thống nhúng có thể gây ra tai nạn khủng khiếp, đặc biệt là các hệ thống điều khiển máy bay, tên lửa, hệ thống điều khiển động cơ ô tô... Lỗi trên hệ thống nhúng có thể không sửa được (ví dụ: vệ tinh nhân tạo), nếu sửa được thì chi phí cũng rất cao, phải thu hồi sản phẩm hoặc thiết kế lại toàn bộ. Do vậy, đảm bảo chất lượng cho các hệ thống nhúng là quan trọng và cần thiết.

Cùng với sự phát triển mạnh mẽ của hệ thống nhúng, các công cụ hỗ trợ thiết kế mô hình các hệ thống nhúng được áp dụng ngày càng nhiều. Việc thiết kế mô hình hệ thống nhúng trên các công cụ trước khi thiết kế mẫu thật là cần thiết để dễ dàng phát hiện, sửa lỗi cũng như chỉnh sửa thiết kế nhằm đảm bảo chất lượng. Vì vậy, có nhiều nghiên cứu, ứng dụng hỗ trợ việc kiểm thử mô hình các hệ thống nhúng thiết kế [1,2,3,5,6,7,8,9,10,14]. Bài báo này hướng tới bài toán kiểm thử cho mô hình hệ thống nhúng.

Hoạt động kiểm thử một hệ thống phần mềm nói chung và hệ thống nhúng nói riêng là vấn đề thường chiếm tới 30-50% thời gian cũng như kinh phí làm ra một hệ thống. Trong đó việc sinh dữ liệu kiểm thử (test data) quyết định đến chất lượng của kết quả kiểm thử. Một bộ test data được đánh giá là tốt nếu có khả năng phát hiện lỗi cao, đạt độ phủ theo chuẩn cho trước. Đã có nhiều tiêu chuẩn yêu cầu về an toàn các hệ thống nhúng, như ISO 26262, IEC 61508, EN-50128, IEC 60880, and DO-178C... Trong các hệ thống này, yêu cầu kiểm thử cao hơn, đòi hỏi phải có bộ test data để phủ hết tất cả các đường chạy xảy ra.

Không như các hệ thống phần mềm thông thường, tín hiệu đầu vào của hệ thống nhúng là các chuỗi dữ liệu theo thời gian. Do vậy, việc sinh test data cho hệ thống nhúng gặp nhiều khó khăn và phức tạp hơn so với hệ thống phần mềm thông thường.

Việc kiểm thử mô hình hệ thống nhúng có 2 vấn đề chính;

Vấn đề 1: Đầu vào của hệ thống nhúng thường là tín hiệu liên tục theo thời gian. Các phương pháp kiểm thử hoặc xác minh đa phần áp dụng cho hệ thống với thời gian rời rạc. Các nghiên cứu này sinh dữ liệu kiểm thử rời rạc cho các mô hình nhúng với mục tiêu tìm lỗi runtimes [3,11], vi phạm các tính chất dựa trên đặc tả hình thức (formal specification) [9], và độ phủ cao [8]. Tuy nhiên, dữ liệu test rời rạc (discrete test data) hiếm khi tồn tại trong mô hình hệ thống nhúng mà là các tín hiệu liên tục theo thời gian.

Vấn đề 2: Các mô hình thường phức tạp với số lượng tín hiệu vào, tín hiệu ra, sự kiện lớn. Nghĩa là, để kiểm thử thủ công sao cho phủ hết các trường hợp xảy ra là không khả thi. Thêm vào đó, việc áp dụng các phương pháp hình thức hay phân tích tĩnh để sinh ra dữ liệu test với độ phủ cao là không dễ.

Có một số nghiên cứu liên quan tới bài toán sinh dữ liệu kiểm thử cho mô hình hệ thống nhúng [1,5,6,14].

Matinnejad [5, 6]) sinh dữ liệu kiểm thử cho mô hình Simulink với thời gian liên tục. Tác giả hướng tới thiết kế cho từng công vào 1 tín hiệu dựa trên thuật toán tìm kiếm nhằm thu được tín hiệu đầu ra đa dạng nhất. Phương pháp

này không thể áp dụng để sinh dữ liệu kiểm thử với tiêu chí các độ phủ CC, DC, MC/DC vì nó hướng tới mối quan hệ giữa thay đổi của output và inport. Trong tiêu chí các độ phủ CC, DC, MC/DC, chúng ta quan tâm tới tất cả các điều kiện (condition) bên trong model.

Trong [1], Godbole et.al. sinh dữ liệu kiểm thử cho mô hình Simulink với Độ phủ nhánh. Đầu tiên, tác giả sinh code C cho mô hình Simulink, sau đó áp dụng công cụ kiểm thử cho code C để thu được bộ dữ liệu kiểm thử và tính toán các độ phủ CC, DC, MC/DC. Vấn đề duy nhất của phương pháp này là nó sử dụng 1 công cụ kiểm thử tĩnh cho code C. Do vậy, phương pháp này không khả thi với mô hình lớn.

Trong [14], Tomita đề xuất 1 phương pháp kiểm thử trong đó mỗi tín hiệu vào được gán cho 1 khuôn mẫu sẵn (tín hiệu sine, step...). Sau đó, một dữ liệu kiểm thử sẽ là 1 trường hợp cụ thể của khuôn mẫu này. Phương pháp này cho phép sinh dữ liệu kiểm thử 1 cách dễ dàng và kiểm soát việc sinh dữ liệu thông qua số lượng ít tham số của khuôn mẫu (tần số, biên độ...) thay vì phải sinh dữ liệu cho toàn bộ tín hiệu. Tuy nhiên, khi mô hình lớn, số lượng inports lớn, việc chọn ngẫu nhiên các loại tín hiệu khiến cho các độ phủ CC, DC, MC/DC có thể không được như kì vọng.

Bài báo này đề xuất 1 phương pháp sinh dữ liệu kiểm thử tự động dựa trên các loại tín hiệu mẫu kết hợp với kĩ thuật kiểm thử theo cấp nhằm thu được bộ dữ liệu kiểm thử với độ phủ cao. Kết quả thử nghiệm cho thấy phương pháp đề xuất cho kết quả tốt hơn phương pháp kiểm thử trong [14].

## II. TỔNG QUAN VỀ KIỂM THỬ MÔ HÌNH HỆ THỐNG NHÚNG

### II.1 Các khái niệm cơ bản về kiểm thử mô hình hệ thống nhúng

Có nhiều công cụ dùng để thiết kế mô hình các hệ thống nhúng, trong đó MATLAB/Simulink là một công cụ được sử dụng nhiều trong cả nghiên cứu và thực tế. Bài báo sẽ thực hiện kiểm thử cho các mô hình biểu diễn bằng MATLAB/Simulink [13].

#### a. Mô hình hệ thống nhúng Simulink

Một mô hình hệ thống nhúng MATLAB/Simulink, được tạo bởi nhiều loại blocks, bao gồm: inport/outputport (vào/ra), mathematical operator (phép toán), logical/relational operator (phép logic/quan hệ), (multiport) switch, delay... Các blocks liên kết với nhau bằng lines, truyền dữ liệu Boolean, integer hoặc floating/fixed point giữa chúng. Đặc biệt, một mô hình hệ thống nhúng cho phép nhận được một số tín hiệu (các giá trị liên tục theo thời gian) bằng cách sử dụng các block Inport và tạo ra một số tín hiệu đầu ra được đại diện bởi các block Outputport.

#### b. Các độ phủ CC, DC, MC/DC

Trong khoa học máy tính, mức độ phủ (Test coverage) là một thước đo được sử dụng để mô tả mức độ một hệ thống được thực thi trên một bộ test data cụ thể. Một bộ test data có mức độ phủ cao sẽ làm cho nhiều block trong

hệ thống được thực thi được trong quá trình kiểm thử. Vì vậy, sẽ có khả năng phát hiện được nhiều lỗi hơn.

Yêu cầu thông thường cho việc kiểm thử mô hình là một bộ test data có độ phủ cao. Trên thực tế, chuẩn ISO 26262 (“Road vehicles – Functional safety”) yêu cầu kiểm thử và đánh giá các độ phủ CC, DC, MC/DC. MATLAB/Simulink có một công cụ phục vụ việc đánh giá Độ phủ này: Simulink Design Verifier (SLDV) và Simulink Verification and Validation (V&V).

Decision coverage (DC) hay còn được biết đến là Branch coverage (Bao phủ nhánh). Độ phủ DC đánh giá số lượng các các điểm quyết định, rẽ nhánh (như là if, switch ...) trong hệ thống được thực thi cho cả trường hợp giá trị true và trường hợp false khi thực hiện kiểm thử.

Condition coverage (CC) tương tự như bao phủ quyết định nhưng nó có độ nhạy tốt hơn với các điều kiện con. Độ phủ DC đánh giá số lượng các biểu thức Boolean con trong từng điểm quyết định của hệ thống cho cả trường hợp giá trị true và trường hợp false khi thực hiện kiểm thử. Các biểu thức boolean con được phân tách bằng các phép logic-AND hoặc logic-OR nếu cùng xảy ra

Modified condition and decision coverage (MC/DC) là mức đánh giá chi tiết cho từng biểu thức điều kiện. Độ phủ MC/DC đánh giá số lượng các biểu thức Boolean con trong từng điểm quyết định của hệ thống cho cả trường hợp true và trường hợp false và giá trị này sẽ phải quyết định tới cả giá trị điều kiện cha của nó.

#### c. Kiểm thử mô hình hệ thống nhúng dựa trên tín hiệu mẫu

Một test data cho mô hình hệ thống nhúng được xem như là 1 vector của các tín hiệu inport. Các tín hiệu có thể ở dạng bất kì. Tuy nhiên, các tín hiệu inport được cung cấp bởi các thiết bị điều khiển, hoặc được sinh ra bởi một đối tượng vật lý và thường tuân theo các định luật vật lý. Do vậy, ta không nhất thiết phải sử dụng các tín hiệu bất kì. Ngoài ra, việc sử dụng các loại tín hiệu vào mẫu (sine, step,...) sẽ giúp điều khiển tín hiệu đơn giản hơn với chỉ một vài tham số (ví dụ, tần số, cường độ...). [14] sử dụng các tín hiệu mẫu để sinh test data cho mô hình Simulink.

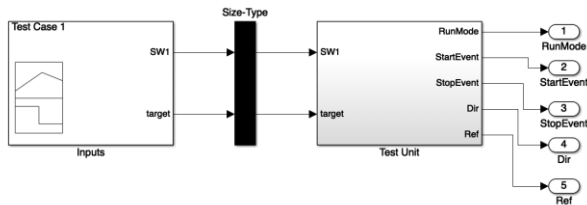
#### Định nghĩa 1[14]: Test data/Bộ Test data.

Một test data là một nhóm các tín hiệu đầu vào. Một bộ test data là một tập các test data.

Với một test data, độ phủ của mô hình nhúng là độ phủ của các đối tượng trong mô hình. Độ phủ của mô hình cho một bộ test data là tổng hợp các độ phủ của từng test data. Trong các độ phủ DC, CC và MC/DC, đối tượng để đánh giá là block có các hành vi logic. Trong Simulink, thì chúng là block logic, phép relation, block chuyển đổi (multiport switch), các subsystem với các công điều khiển hoạt động...

Để có thể kiểm thử mô hình hệ thống nhúng, ta cần đưa các test data vào mô hình và gán cho mỗi inport 1 tín hiệu tương thích sau đó chạy mô phỏng với các inport này rồi đánh giá độ phủ tương ứng. Để làm điều đó, một mô hình harness được tạo ra từ mô hình gốc, trong đó mô hình gốc được thể hiện như 1 hệ thống con (Test unit). Các test data

có thể đưa vào qua khối lệnh Signal Builder (Inputs) trong Simulink và phân phối cho các inport tương ứng của Test unit thông qua khối Size-Type. Hình 1 minh hoạ 1 hệ thống harness như vậy.



Hình 1: Mô hình hệ thống nhúng harness trong Simulink cho phép thêm các test data.

**Định nghĩa 2[14]:** Độ phủ

Một test data là phủ DC đầy đủ nếu, đối với mỗi đối tượng để đánh giá b, tất cả giá trị ra có thể của b đều được xảy ra tại một thời điểm nào đó bởi một test data trong bộ test data. Một test data là phủ CC đầy đủ nếu, với mỗi điều kiện c của b, mỗi giá trị ra có thể của c xảy ra tại một thời điểm nào đó bởi một test data trong bộ test data. Một test data là phủ MC/DC đầy đủ nếu, đối với mỗi b mà quyết định phụ thuộc vào nhiều điều kiện  $c_1, \dots, c_n$ , test data bao gồm các test data trong đó từng  $c_m$  ảnh hưởng độc lập đến decision của nó.

Tư tưởng của việc sinh test data dựa trên tín hiệu mẫu như sau : đối với mỗi ca kiểm thử truyền vào, tiến hành tính toán các độ phủ DC, CC, MC/DC cho mô hình. Nếu độ phủ của ca kiểm thử sau lớn hơn hoặc bằng ca kiểm thử trước, hiển thị kết quả tính toán độ phủ lên màn hình. Sau khi tính toán hết độ phủ của các ca kiểm thử, kết quả trả về là các độ phủ DC, CC, MCDC cao nhất và bộ test. Cụ thể:

- Bước 1: Sinh ngẫu nhiên một test data ứng viên (là một nhóm các tín hiệu inport).
- Bước 2: Chạy mô phỏng cho mô hình với test data vừa sinh và đánh giá độ phủ của nó. Việc chạy mô phỏng và đánh giá các độ phủ CC, DC, MC/DC.
- Bước 3: Nếu test data này làm tăng độ phủ DC hoặc CC hoặc MC/DC, ta thêm nó vào bộ test data.
- Bước 4: Các bước trên được thực hiện lặp đi lặp lại cho tới khi đạt độ phủ tối đa hoặc đạt tới ngưỡng số lượng lần lặp cho trước.

**II.2 Kỹ thuật kiểm thử theo cặp**

Pairwise testing (hay All-pairs testing) [12] là một phương pháp kiểm thử hộp đen. bằng cách sử dụng phương pháp tổ hợp để kiểm tra tất cả sự kết hợp rời rạc có thể của các tham số liên quan. Phương pháp này dựa trên sự quan sát cho thấy phần lớn lỗi đều bắt nguồn từ sự tương tác giữa hai tham số. Do đó, pairwise testing tạo ra các ca kiểm thử phủ hết giá trị của hai tham số. Thực nghiệm cho thấy, Pairwise testing có khả năng sinh test data phủ MC/DC tốt hơn so với kiểm thử ngẫu nhiên [12].

- Phương pháp pairwise testing bao gồm:
  - Lựa chọn tham số đầu vào và các giá trị tương ứng
  - Lấy tổ hợp (pairwise) của các giá trị giữa 2 tham số

- Xây dựng bộ test sao cho bao phủ được tất cả các cặp xác định ở trên.

Lấy một ví dụ đơn giản như sau:

Một Mô hình hệ thống nhúng có 4 inports, trong đó inport1 nhận giá trị {1,2,3}; inport2 nhận giá trị {4,5}, inport3 nhận giá trị {1,2,5,6}, inport4 nhận giá trị {1,5}

Bảng 1: Các ca kiểm thử theo cặp

Inport1	Inport2	Inport3	Inport4
2	4	6	1
2	5	2	5
3	5	1	1
2	4	5	5
1	4	2	1
3	5	6	5
3	5	5	1
3	4	2	5
1	5	6	5
1	4	1	5
1	4	5	1
2	5	1	1
2	4	6	1

Khi áp dụng kỹ thuật kiểm thử theo cặp, ta có tất cả 13 ca kiểm thử phủ hết tất cả các cặp giá trị xảy ra của 2 inports bất kì.

Chi tiết về kỹ thuật pairwise testing có thể tham khảo tại [13], vì vậy, trong phạm vi bài báo này sẽ không trình bày lại kỹ thuật sinh pairwise test nữa.

**III. ÁP DỤNG KIỂM THỬ THEO CẶP CHO SINH DỮ LIỆU KIỂM THỬ MÔ HÌNH HỆ THỐNG NHÚNG**

**III.1. Mã hoá và giải mã miền dữ liệu đầu vào tương ứng với kỹ thuật kiểm thử theo cặp**

Mô hình hệ thống nhúng đang ở dạng sơ đồ với các inport, và một file mô tả các tín hiệu sẽ sử dụng để truyền vào các inport.

Tuy nhiên, do yêu cầu của kỹ thuật kiểm thử theo cặp cần đầu vào là các tham số, mỗi tham số nhận các giá trị rời rạc, không phải là các tín hiệu.

Vì vậy, ta cần mã hoá miền dữ liệu đầu vào thành các bộ tham số và giá trị dựa theo định nghĩa sau.

**Định nghĩa 1:** Cho mô hình M, miền dữ liệu  $D = \{(p_1, \dots, p_n) \mid p_i \in \{1, \dots, k_i\}\}$ , với tham số  $p_i$  tương ứng với đầu vào thứ i trong mô hình M,  $\{1, \dots, k_i\}$  là chỉ số cho các loại tín hiệu đầu vào thứ i có thể nhận.

*Ví dụ 1:* Đầu vào của mô hình là Inport1 và Inport2, suy ra số đầu vào của mô hình là 2.

*Giả sử,* Trong mỗi Inport, có 6 loại tín hiệu được sử dụng: Sine, Step, Square, Linear, Constant, Triangle. Ta đánh chỉ số cho Sine là 1, Step là 2, Square là 3, Linear là 4, Constant là 5, Triangle là 6. Do vậy, số tín hiệu sẽ sử dụng của mỗi đầu vào là 6, ta mã hoá lần lượt là 1,..., 6. Do đó, miền D là:  $\{(inport1, inport2) \mid inport1, inport2 \in \{1,2,3,4,5,6\}\}$

Sau khi áp dụng kỹ thuật kiểm thử theo cặp, chúng ta sẽ thu được các bộ giá trị  $(v_1, \dots, v_n)$  đảm bảo phủ hết các cặp giá trị xảy ra của 2 tham số  $p_i, p_j$  bất kì. Chúng ta cần chuyển bộ giá trị này thành test data của mô hình M tức là tương ứng với inport i của mô hình M, sẽ nhận tín hiệu thứ  $v_i$ .

III.2. Thuật toán đề xuất

Để áp dụng kỹ thuật kiểm thử theo cặp vào sinh dữ liệu kiểm thử cho mô hình hệ thống nhúng, ta cần cải tiến thuật toán của của Tomita [14] để có thể áp dụng kỹ thuật kỹ thuật kiểm thử theo cặp, cụ thể ta phải bổ xung các yêu cầu sau:

- ta cần mã hoá thông tin các inport của mô hình thành dạng tham số (là inport) giá trị (là các loại tín hiệu mà inport đó có thể nhận) để thành đầu vào của kỹ thuật kiểm thử theo cặp

- thực hiện sinh các ca kiểm thử theo cặp để phủ hết các cặp giá trị của 2 tham số bất kì (là bất kì tổ hợp của 2 loại tín hiệu của 2 inports).

- biến đổi các ca kiểm thử theo cặp thành các tín hiệu đầu vào cho mô hình hệ thống nhúng.

Như vậy, thay vì lựa chọn loại tín hiệu cho các inport 1 cách ngẫu nhiên, các loại tín hiệu sẽ được gán cho các inports theo kỹ thuật kiểm thử theo cặp.

Thuật toán 1 được đề xuất như sau.

**Thuật toán 1: Sinh Testsuite áp dụng kỹ thuật pairwise testing**

```

Input:
- mô hình cho kiểm thử MT
- mô tả cấu hình các loại tín hiệu C
Output:
- test data T
Các bước thực hiện:
Begin
    D = encode(MT, C);
    PT = pairwise (D);
    ST = decode (PT);
    T= ∅;
    Coverage = (0,0,0);
    for t ∈ ST
    begin
        simulation (MT,t);
        tCoverage = checkMCDCC();
        if (Coverage <tCoverage)
        begin
            T = T ∪ {temp};
            Coverage = tCoverage;
        end if;
    end for;
    return T;
end;
```

Trong Thuật toán 1:

- Hàm encode(MT, C) sẽ lấy thông tin các inport của mô hình MT, cùng với mô tả cấu hình các loại tín hiệu C, từ đó mã hoá thành miền dữ liệu D theo Định nghĩa 1.
- Hàm pairwise(D) sẽ áp dụng kỹ thuật pairwise testing để sinh bộ test data PT tương ứng với D.
- Hàm decode (PT) sẽ biến đổi từng phần tử của PT thành test data của MT, tức là bộ tín hiệu đầu vào

cụ thể tương ứng với chỉ số của nó.

- Hàm simulation(MT,t) thực hiện chạy mô phỏng cho mô hình MT với thiết lập inport là t.
- Hàm checkMCDCC() trả về giá trị phủ theo DC, CC, và MC/DC khi chạy thêm mô phỏng với t.

Thuật toán này chỉ lựa chọn các test data t vào tập test data T nếu mô phỏng MT với t làm tăng độ phủ DC, hoặc CC, hoặc MC/DC.

IV. THỰC NGHIỆM VÀ ĐÁNH GIÁ

Thuật toán đề xuất được cài đặt và thử nghiệm bằng mScript trên môi trường MATLAB để sinh dữ liệu test cho các mô hình biểu diễn bằng MATLAB/Simulink, với đầu vào là 1 mô hình Simulink, 1 file đặc tả các tín hiệu, đầu ra sẽ là bộ test data và độ phủ tương ứng.

4 mô hình được lựa chọn từ các mô hình được áp dụng trong thực tế của các hệ thống nhúng trong điều khiển ô tô. Bảng sau mô tả các mô hình bao gồm số lượng blocks trong mô hình, số lượng inport tương ứng.

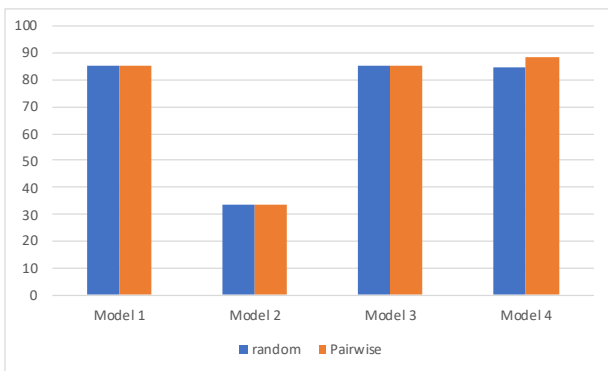
Bảng 2: Danh sách model thử nghiệm

Model No.	Số lượng block	Số lượng inport
Model 1	325	2
Model 2	501	2
Model 3	453	19
Model 4	2432	51

Ba thử nghiệm được thực hiện để đánh giá các độ phủ DC, CC, MC/DC và so sánh giữa kiểm thử ngẫu nhiên và phương pháp đề xuất (áp dụng kiểm thử theo cặp). Để đảm bảo việc so sánh là công bằng, số lượng lần lặp và thời gian thực hiện được cấu hình giống nhau giữa kiểm thử ngẫu nhiên và phương pháp đề xuất. Kết quả thử nghiệm được chi ra ở 3 bảng sau (Bảng 3, Bảng 4, Bảng 5) và biểu đồ minh hoạ tương ứng (Hình 2, Hình 3, Hình 4). Kết quả ở dạng % số DC (hay CC, MC/DC) mà các ca kiểm thử đã phủ được trên tổng số DC (hay CC, MC/DC) có trong mô hình.

Bảng 3: So sánh độ phủ DC giữa random và pairwise

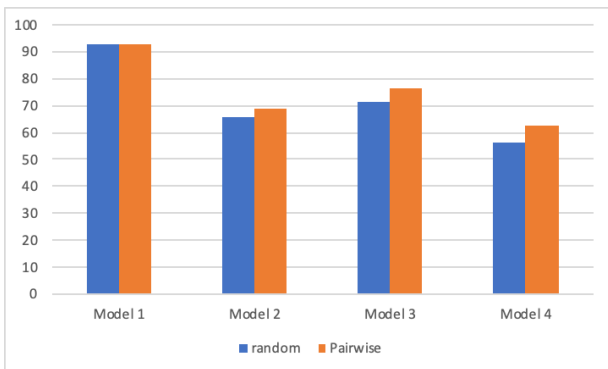
Model No.	random (%)	Pairwise (%)
Model 1	86	86
Model 2	33	33
Model 3	85	85
Model 4	84	<b>89</b>



Hình 2: Biểu đồ so sánh độ phủ DC

Bảng 4: So sánh độ phủ CC giữa random và pairwise

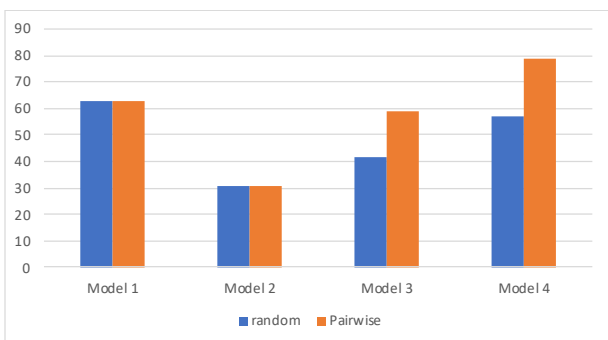
Model No.	random (%)	Pairwise (%)
Model 1	93	93
Model 2	66	<b>69</b>
Model 3	71	<b>77</b>
Model 4	56	<b>63</b>



Hình 3: Biểu đồ so sánh độ phủ CC

Bảng 5: So sánh độ phủ MC/DC giữa random và pairwise

Model No.	random (%)	Pairwise (%)
Model 1	63	63
Model 2	31	31
Model 3	41	<b>59</b>
Model 4	57	<b>79</b>



Hình 4: Biểu đồ so sánh độ phủ MC/DC

Từ kết quả thử nghiệm trên, ta có nhận xét: Với các mô hình nhỏ (số lượng block, số lượng inport nhỏ), thì việc sử dụng phương pháp sinh test data ngẫu nhiên (Random) sẽ đạt hiệu quả tương đương, còn với các mô hình lớn, thì phương pháp áp dụng pairwise cho kết quả tốt hơn, đặc biệt là độ phủ MC/DC, CC.

## V. KẾT LUẬN

Bài báo đề xuất một phương pháp để sinh test data tự động cho các mô hình hệ thống nhúng với tín hiệu đầu vào liên tục. Thay vì sinh test data là các tín hiệu ngẫu nhiên, kỹ thuật kiểm thử theo cặp được sử dụng nhằm giúp việc sinh dữ liệu kiểm thử hiệu quả hơn. Để thực hiện việc này, Thuật toán 1 đã được đề xuất nhằm biến đổi yêu cầu kiểm thử và mô hình thành đầu vào của thuật toán pairwise testing, sau đó kết quả của thuật toán pairwise được biến đổi thành các test data của mô hình, tức là các tín hiệu mẫu. Nhờ vậy, các test data cho mô hình nhúng có thể được sinh và thực hiện tự động. Kết quả thử nghiệm bước đầu cho thấy phương pháp đề xuất có khả năng sinh ra bộ test data cho các mô hình nhúng với Độ phủ cao hơn so với phương pháp đề xuất trong [14].

Hướng phát triển của đề tài là áp dụng các kỹ thuật phân tích tĩnh (static analysis) để đưa ra thông tin về mô hình, từ đó hỗ trợ việc sinh test data có Độ phủ cao hơn. Một hướng phát triển khác của đề tài là thử nghiệm các phương pháp kiểm thử và sinh dữ liệu test khác, từ đó đưa ra lựa chọn các test data có độ phủ cao hơn nữa.

## REFERENCES

- [1] Godbole, S., Sridhar, A., Kharpuse, B., Mohapatra, D.P. and Majhi, B., Generation of branch coverage test data for simulink/stateflow models using crest tool. International Journal of Advanced Computer Research, 3(4), p.222, 2013
- [2] Godefroid, P., Klarlund, N. and Sen, K., 2005, June. DART: directed automated random testing. In ACM Sigplan Notices (Vol. 40, No. 6, pp. 213-223). ACM.
- [3] Holling, D., Pretschner, A. and Gemmar, M., 2014, September. 8cage: lightweight fault-based test generation for simulink. In Proceedings of the 29th ACM/IEEE international conference on Automated software engineering (pp. 859-862). ACM.
- [4]. Kuhn, D. Richard, Dolores R. Wallace, and Albert M. Gallo. "Software fault interactions and implications for software testing." IEEE transactions on software engineering 30.6 (2004): 418-421.
- [5] Matinnejad, R., Nejati, S., Briand, L.C. and Bruckmann, T., 2016, May. Automated test data generation for time-continuous simulink models. In Proceedings of the 38th international conference on software engineering (pp. 595-606). ACM.
- [6] Matinnejad, R., Nejati, S., Briand, L. C., & Bruckmann, T. (2016, May). SimCoTest: A test data generation tool for Simulink/Stateflow controllers. In Proceedings of the 38th International Conference on Software Engineering Companion (pp. 585-588). ACM.
- [7] Pacheco, C., Lahiri, S.K., Ernst, M.D. and Ball, T., 2007, May. Feedback-directed random test generation. In Proceedings of the 29th international conference on

Software Engineering (pp. 75-84). IEEE Computer Society.

[8] Peranandam, P., Raviram, S., Satpathy, M., Yeolekar, A., Gadkari, A. and Ramesh, S., 2012, March. An integrated test generation tool for enhanced coverage of Simulink/Stateflow models. In Proceedings of the Conference on Design, Automation and Test in Europe (pp. 308-311). EDA Consortium.

[9] Richardson, D.J., Aha, S.L. and O'malley, T.O., 1992, June. Specification-based test oracles for reactive systems. In Proceedings of the 14th international conference on Software engineering (pp. 105-118). ACM.

[10] Satpathy, M., Yeolekar, A. and Ramesh, S., 2008, October. Randomized directed testing (REDIRECT) for Simulink/Stateflow models. In Proceedings of the 8th ACM international conference on Embedded software (pp. 217-226). ACM.

[11] Sanchez, J., 2016. A review of pair-wise testing. arXiv preprint arXiv:1606.00288.

[12] Simulink Design Verifier: <https://www.mathworks.com/products/sldesignverifier.html> [Online; accessed 5-May- 2020].

[13]. Sims, S. and DuVarney, D.C., 2007, October. Experience report: the reactis validation tool. In ACM SIGPLAN Notices (Vol. 42, No. 9, pp. 137-140). ACM.

[14]. T. Tomita, Daisuke.I, Toru. M, Shigeki.T, T. Aoki, Template-Based Monte-Carlo Test Generation for Simulink Models, Workshop on Design, Modeling and Evaluation of Cyber Physical Systems (CyPhy'17 ), LNCS 11267.

[15] Vu T.D., Hung P.N., Nguyen V.H. (2017) A Method for Automated Test data Generation from UML Models with String Constraints. In: Król D., Nguyen N., Shirai K. (eds) Advanced Topics in Intelligent Information and Database Systems. ACIIDS 2017. Studies in Computational Intelligence, vol 710. Springer, Cham



**Đỗ Thị Bích Ngọc** sinh ngày 08/03/1981 tại Hà Nội. Năm 2004, bà tốt nghiệp Kỹ sư tài năng Công nghệ thông tin tại Trường Đại học Bách khoa Hà Nội. Năm 2007 bà nhận bằng Thạc sĩ khoa học tại Trường Đại học Sư Hà Nội. Năm 2010, bà nhận học vị Tiến sĩ chuyên ngành Khoa học thông tin tại Viện Khoa học và Công nghệ Tiên tiến Nhật Bản (JAIST). Từ

năm 2010-2012, bà làm sau tiến sĩ tại viện AIST – Nhật Bản. Từ 2013- nay, bà là Giảng viên tại Học viện Công nghệ Bưu chính Viễn thông.

*Lĩnh vực nghiên cứu:* Phân tích mã nguồn, Kiểm thử phần mềm, Kiểm chứng phần mềm, Công nghệ phần mềm.

## TEST DATA GENERATION FOR EMBEDDED MODEL USING PAIRWISE TESTING TECHNIQUE

**Abstract:** Embedded systems are playing more and more important role in society. These systems require high safety. Thus, quality assurance for these kinds of systems has been attracted many attention and investment of both academic research and industry communities. In embedded systems, testing often requires high coverage with different measures respect to international standards like DC, CC, MC/DC coverages respected to ISO 26262. In this paper, in order to have good test cases with high DC, CC, MC/DC coverages, we propose a method to automatically generate test-cases by applying pairwise testing technique. Experiments shown that our method has order of magnitude better than that of the random testing method.

**Keywords:** CC coverage, DC coverage, MC/DC coverage, Embedded model, Testing, Pairwise testing, Continuous signal.