한국정보전자통신기술학회논문지(jkiiect)'19-12 Vol.12 No.6
논문 19-12-06-597 http://dx.doi.org/10.17661/jkiiect.2019.12.6.597

Adaptive Application of CPP Algorithm to Test Suite Generation for Protocol Conformance Testing

Chul Kim*

프로토콜 적합성 시험항목 생성시 CPP 알고리즘의 적응적 적용 방안

김 철*

Abstract In this paper, we propose an improved method on an adaptive application of the CPP(Chinese Postman Problem) algorithm to the protocol test suite generation for conformance testing. Also, we present an example application of this CPP algorithm to B-ISDN Q.2931 call/connection control procedure for the purpose of showing how it can be adapted to generate a test suite for conformance testing of a communication protocol. The proposed method has an advantage of an optimization technique which finds a minimum cost of test suite from a standardized specification, so this optimization technique of the CPP algorithm can be practically applied to a real environment for testing a conformity of a protocol implementation.

요 약 본 논문은 프로토콜 구현물이 프로토콜의 사양에 대한 적합성을 시험하기 위한 시험항목 생성시 CPP(중국 집배원 문제) 알고리즘의 개선된 적응적 적용 방법을 제안한다. 또한 본 논문에서 제안한 개선된 CPP 알고리즘 방법이 통신 프로토콜의 적합성 시험을 위한 시험항목 생성에 어떻게 적응될 수 있는지 B-ISDN Q.2931 호/연결 제어 절차에 실례로 적용하여 본다. 본 방법의 실험적인 적용 결과 기존의 시험열 생성 방법들에 비해 최적화된 최소 비용의 시험열을 생성함을 확인하였다. 본 논문의 제안된 방법은 표준화된 사양으로부터 최소 비용의 시험항목을 구하는 최적화 기법으로써의 장점을 가지고 있는데 본 논문의 개선된 CPP 알고리즘의 이 최적화 기법은 프로토콜 구현물의 적합성을 시험하기 위한 실재 환경에 실제적으로 적용될 수 있다. 향후 연구에서는 프로토콜 적합성시험을 위한 시험항목 생성시 상위 시험기와 하위 시험기간의 시험 조정 절차에서 발생할 수 있는 동기화 문제를 해결하고 적용하는 방안이 강구되어야 할 것으로 사료된다.

Key Words : Chinese Postman Problem(CPP), conformance testing, Finite State Machine(FSM), implementation under test, minimum-cost tour, test sequence

1. Introduction

A protocol can be specified as a deterministic Finite State Machine (FSM), where the state of the protocol is defined as a stable condition in which the protocol rests until a stimulus, called an input, is applied. An input causes the protocol to generate an output (which may be null) and to undergo a transition from the current state to a new state, where it stays until the next input [1, 2, 3]. The purpose of conformance testing is to test whether there is a discrepancy between the specification and implementation of an FSM [4]. The technique for generating a test sequence is

*Department of Computer Science, Yongin University
Received November 27, 2019 Revised December 08, 2019 Accepted December 10, 2019

based upon an optimization technique, such as the well-known Chinese Postman Problem (CPP), on directed graphs [5, 6]. In this paper, we present an adaptive application of the CPP algorithm which finds a minimum cost input sequence for exercising a given set of transitions of an FSM.

From the experimental result of its application to the Q.2931, shown in Section 4, the proposed method has a good capability of optimization in a test sequence, while the existing methods such as transition tour, characterization set, distinguishing sequence, unique input/output does not provide an optimum length of the test sequence. Thus, our method does shorten the total length of a test sequence because the state identification sequences and test subsequences should be minimized when they might be derived, and also because the total length of a test sequence should be minimum-costed based on the optimization technique.

2. Preliminaries

We first introduce the relevant basic properties of a directed graph, and then discuss some important applications of the directed graph such as the maximum flow, and the maximum bipartite matching which are relating to the protocol testing [5, 6, 7, 8].

Definition 1 (Subgraph, G′ and Components of Graph) It is said that $G'=(V', E')$ is a subgraph of $G=(V, E)$ if $V' \subseteq V$ and $E' \subseteq E$ such that G' can be obtained by the removal of a (non-zero) number of edges and/or vertices of G. The subgraphs induced in turn by the subsets $G_1', G_2', \ldots, G_n'$ whose a pair of vertices should be connected, are called components of the graph G.

Definition 2 (Edge-Induced Subgraph or Edge-Induced Spanning Subgraph, G[E_tss]) It is said that an edge-induced subgraph $G[E_{tss}]$ where tss stands for test subsequence is the subgraph of G whose vertex set is the set of heads and tails of edges in E_{tss} where $V' \subseteq V$ and whose edge set is E_{tss} where $E_{tss} \subseteq E$. Or it is also said that $G[E_{tss}]$ is an edge-induced spanning subgraph of G if its vertex set is V such that $G[E_{tss}]$ can be obtained by removing edges of G only and by remaining any pair of vertices of V unchanged.

Definition 3 (Flow Function of Network, F) A flow F of a network G_F with source s and sink t is a non-negative real valued function defined on the edges of G such that *for any*

(1) Capacity constraint:

$$0 \le F(v_j, v_k) \le H(v_j, v_k), \quad (1)$$

for each edge $(v_j, v_k) \subseteq E$.

(2) Skew symmetry:

$$F(v_j, v_k) = -F(v_k, v_j),$$

(2)

for $\forall v_j, v_k \in V - \{s, t\}$.

(3) Flow conservation equation:

$$\sum_{vi \in V} F(v_j, v_k) = 0,$$

(3)

for $\forall v_i \in V - \{s, t\}$

$$\text{and } F(v_j, v_k) = 0, \quad (4)$$

edge (v_j, v_k) *where* $v_k=s$ *or* $v_j=t$.

$F(v_j, v_k)$ can be either positive or negative and it is called the net flow from v_j to v_k. The value of a flow F is given by

$$|F| = \sum_{vi \in V} F(s, v_k). \quad (5)$$

Definition 4 (Bipartite Graph, G_B, Bipartite Matching, M_B, and Maximum Bipartite Matching, M_{MB}) For a directed graph $G(V, E)$, if a disjoint vertex set V_s and V_t can be found such that $V = V_s \cup V_t$ and all edges in E are between V_s and V_t. Then the directed graph $G_B(V_B, E_B)$ is denoted as bipartite graph where $V_B \equiv V \equiv V_s \cup V_t$ and $E_B \equiv E$. A bipartite matching M_B is a matching on a directed bipartite graph. A maximum bipartite matching M_{MB} is a maximum matching on a directed bipartite graph.

We can transform the problem of finding a bipartite maximum matching in a directed bipartite graph $G_B(V_B, E_B)$ into the problem of finding the maximum flow in a bipartite matching flow network $G_{BF}(V_{BF}, E_{BF})$ with $H_{BF}(v_j, v_k)$, augmented from the bipartite graph G_B, where

$$H_{BF}(v_j, v_k) = 1 \text{ for } \forall(v_j, v_k) \in E_B, \quad (6)$$

$$V_{BF} \equiv V_B \cup \{s, t\} \equiv \{V_s \cup V_t\} \cup \{s, t\}, \quad (7)$$

$$E_{BF} \equiv E_B \cup \{(s, v_j): v_j \in V_s\} \cup \{(v_k, t): v_k \in V_t\}$$
$$\equiv \{(v_j, v_k): (v_j, v_k) \in E,\ v_j \in V_s \text{ and } v_k \in V_t\}$$
$$\cup \{(s, v_j): v_j \in V_s\} \cup \{(v_k, t): v_k \in V_t\}. \quad (8)$$

3. An Improvement of the CPP(Chinese Postman Problem) Algorithm

The minimum-cost symmetric augmentation of G for a *Chinese* postman path(or tour) can be achieved by minimizing $Cost(R)$ of the replication R as follows:

$$Cost(R) = \sum_{(vj,\ vk;\ ip/oq) \in E} Cost(v_j, v_k; i_p/o_q) \cdot R(v_j, v_k; i_p/o_q), \quad (9)$$

where $R(v_j, v_k; i_p/o_q) \geq 0$ and R is an integer, and $Cost(v_j, v_k; i_p/o_q)$ is the cost of edge $(v_j, v_k) \in E$. For $\forall(v_j, v_k) \in E$, Equation (9) is subject to the following Case 1 and Case 2 which are relevant to the symmetry condition.

(1) *Case 1:* $\sum(R \text{ for a vertex } v_i \text{ that is } Head(E))$
$$- \sum(R \text{ for a vertex } v_i \text{ that is } Tail(E)) = d_{in}(v_i) - d_{out}(v_i),\ \forall v_i \in V - \{v_j, v_k\}, \quad (10)$$

$$\sum(R \text{ for a vertex } v_j \text{ that is } Head(E)) - \sum(R \text{ for a vertex } v_j \text{ that is } Tail(E)) = d_{in}(v_j) - d_{out}(v_j) + 1, \quad (11)$$

and

$$\sum(R \text{ for a vertex } v_k \text{ that is } Head(E)) - \sum(R \text{ for a vertex } v_k \text{ that is } Tail(E)) = d_{in}(v_k) - d_{out}(v_k) - 1, \quad (12)$$

where i is an arbitrary number, i.e., $i = 1, 2, 3, \ldots, n$, and v_j is the start vertex and v_k is the final vertex of the *Chinese* postman path.

(2) *Case 2:* $\sum(R \text{ for a vertex } v_i \text{ that is } Head(E))$
$$- \sum(R \text{ for a vertex } v_i \text{ that is } Tail(E)) = d_{in}(v_i) - d_{out}(v_i),\ \forall v_i \in V, \quad (13)$$

where i is an arbitrary number, i.e., $i = 1, 2, 3, \ldots, n$.

We here denote the *index of in-out degree* of a vertex $v_i \in G$, $K(v_i)$, which is defined as the difference between the number of edges in E into v_i and the number of edges in E out of v_i:

$$K(v_i) = d_{in}(v_i) - d_{\mathrm{out}}(v_i). \qquad (14)$$

If $K(v_i) = 0$ for all $i = 1, 2, 3, \ldots, n$ then no edge in E need be included in G; however, if $K(v_i) \neq 0$ then some edges in E must be replicated for G to be symmetric. The minimum-cost symmetric augmentation for a *Chinese* postman path(or tour) can similarly be solved by a minimum-cost maximum flow algorithm with a complexity of $O(|E|\cdot|V|log|V|)$ [2].

Theorem 1: If an edge-induced subgraph $G[E_{tss}]$ of $G(V, E)$ is a weakly connected spanning graph of G, then any minimum-cost symmetric augmentation graph G^* of G is strongly connected and hence G^* has an *Euler* tour.

Proof of Theorem 1: Assume that E_{tss} is weakly connected edge-induced spanning subgraph of G, but that there exists a minimum-cost symmetric augmentation graph G^* of G which is not strongly connected. Then, there exists a bipartition of V^* into two non-empty sets V_x and V_y such that there is no edge from V_y to V_x. Since E_{tss} is weakly connected spanning subgraph of G^* as well as of G, then there exists an edge $(v_j, v_k ; i_p/o_q) \in E_{tss}$ from $v_j \in V_x$ to $v_k \in V_y$ in G^*.

For any directed graph,

$$\sum_{vj\in V*} d_{in}(v_j) - \sum_{vj\in V*} d_{\mathrm{out}}(v_j) = 0; \qquad (15)$$

therefore, for G^*

$$\sum_{vj\in Vx} d_{in}(v_j) - \sum_{vj\in Vx} d_{\mathrm{out}}(v_j) \leq -1. \qquad (16)$$

Since G^* is symmetric, $d_{in}(v_j) = d_{\mathrm{out}}(v_j)$, for all $v_j \in V^*$. Therefore,

$$\sum_{vj\in Vx} d_{in}(v_j) = \sum_{vj\in Vx} d_{\mathrm{out}}(v_j) \qquad (17)$$

thus,

$$\sum_{vj\in Vx} d_{in}(v_j) - \sum_{vj\in Vx} d_{\mathrm{out}}(v_j) = 0, \qquad (18)$$

which is a contradiction. □

Therefore, we can improve an algorithm for solving the CPP(Chinese Postman Problem) in order to adapt it, when a minimum-length test sequence is required to be found during the protocol testing, as follows. The CPP(Chinese Postman Problem) on a directed graph $G(V, E)$, in the general case that G is not symmetric, can be considered as consisting of two major parts: that is, 1) duplicate each edge in E so that the resulting graph G^* is symmetric while minimizing the sum of the lengths of the duplicated edges, 2) find an *Euler* tour of the resulting symmetric graph G^*; hence, a postman tour of minimum length of the original graph G. Part 1) of the problem can be stated as a linear programming problem to be solved by minimizing $Cost(R)$ subject to $R \geq 0$ and R is an integer. Part 2) of the problem can be reduced to a minimum-cost tour of G^* that starts and ends at the same vertex and contains every edge exactly once.

4. Adaptive Application of Q.2931 to Protocol Test Suite Generation

The Q.2931 signaling protocol specifies the procedures for call control of the establishment, maintenance and clearing of network connections at the B-ISDN User Network Interface (UNI) [9, 10, 11, 12]. Fig.1 depicts the B-ISDN signalling protocol stacks for the UNI operation of an Asynchronous Transfer Mode (ATM) network. For the UNI, the Q.2931 connection control protocol is used to set up and clear a connection between users and the B-ISDN network. It operates over an ATM Adaptation Layer (AAL) designed for especially for Q.2931, which is called the Signalling AAL (SAAL). ITU-T recommends Q.2931 for the UNI signalling, and the ATM Forum UNI Signalling Specification(version 4.0) is based on Q.2931 [9, 10].

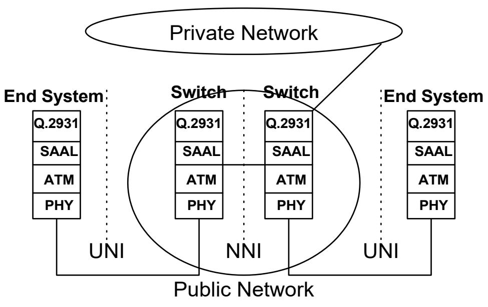

Fig. 1. B-ISDN signalling protocol stacks relating to Q.2931

Table 1 describes The ATM UNI signalling interface provides ten timers at the network side and ten timers at the user side.

Table 1. The Q.2931 timers

Timer Network Side	Timer User Side	Cause for Start	Normal Stop
T301		Not supported in this Implementation Agreement	
T303	*T303*	*SETUP* sent	*CONNECT, CALL PROCEEDING,* or *RELEASE COMPLETE* received
T308	*T308*	*RELEASE* sent	*RELEASE COMPLETE* or *RELEASE* received
T309	*T309*	*SAAL* disconnection	*SAAL* reconnected
T310	*T310*	*CALL PROCEEDING* received	*CONNECT* or *RELEASE* received
	T313	*CONNECT* sent	*CONNECT ACKNOWLEDGE* received
T316	*T316*	*RESTART* sent	*RESTART ACKNOWLEDGE* received
T317	*T317*	*RESTART* received	Internal clearing of call references
T322	*T322*	*STATUS ENQUIRY* sent	*STATUS, RELEASE,* or *RELEASE COMPLETE* received
T398	*T398*	*DROP PARTY* sent	*DROP PARTY ACKNOWLEDGE* or *RELEASE* received
T399	*T399*	*ADD PARTY* sent	*ADD PARTY ACKNOWLEDGE, ADD PARTY REJECT* or *RELEASE* received

The connection setup procedure begins by a user issuing the *SETUP* message. This message is sent by the calling user to the network and is relayed by the network to the called user. Timer *T303* is invoked when ATM issues a *SETUP* message to the network on the local side of network and is invoked by network node at the remote side when it passes the *SETUP* message to the user. Upon receiving the *SETUP* message, the network returns a *CALL PROCEEDING* message to the initiating user, forwards the *SETUP* message to the called user, and waits for the called user to return a *CALL PROCEEDING* message.

Timer *T303* is stopped when the remote end user returns a *CALL PROCEEDING* message. The *CALL PROCEEDING* message is used to indicate that the call has been initiated and no more call establishment information is needed, nor will any be accepted. Upon receiving the *CALL PROCEEDING* message, the local user and remote network node turn off their *T303* timers and turn on their *T310* timers. The timer waits for the *CONNECT* message to be sent to either party.

Table 2. The testing transition edges of Q.2931

T_m	Testing Transition Edges
T_1	*setup.req/SETUP*
T_2	*SETUP/setup.ind*
T_3	*2nd.T303/release.ind*
T_4	*release.resp/RELEASE COMPLETE*
T_5	*RELEASE COMPLETE (or RELEASE)/release.conf*
T_6	*RESTART ACKNOWLEDGE/nu*
T_7	*1st.T303/SETUP*
T_8	*proceeding.req/CALL PROCEEDING*
T_9	*setup.resp/CONNECT*
T_{10}	*CALL PROCEEDING/proceeding.ind*
T_{11}	*CONNECT/setup.conf & CONNECT ACKNOWLEDGE*
T_{12}	*setup.resp/CONNECT*
T_{13}	*CONNECT/setup.conf & CONNECT ACKNOWLEDGE*
T_{14}	*T310/RELEASE & release.ind*
T_{15}	*release.req/RELEASE*
T_{16}	*1st.T308/RELEASE*
T_{17}	*2nd.T308/RESTART*
T_{18}	*1st.T316 & 2nd.T316/RESTART*
T_{19}	*T313/RELEASE & release.ind*
T_{20}	*CONNECT ACKNOWLEDGE/ setup.complete.ind*

The connection release procedure entails only timer *T308*. Either the network or the user can invoke the connection release by sending the *RELEASE* message to the respective party. This operation turns on timer *T308* which remains on until the *RELEASE COMPLETE* message is received. If *T308* expires for the first time, the *RELEASE* message is retransmitted. The receiving network and receiving user are required to transmit the *RELEASE COMPLETE* message as a result of receiving the *RELEASE* message.

Table 2 describes the testing transition edges of the state transition diagram of Q.2931, and Fig.2 shows the experimental result of an adaptive application of the CPP to Q.2931, which should find a minimum-cost tour that starts and ends at the same vertex and contains every edge exactly once.

The resulted test suite: [*SETUP/setup.ind, release.resp/RELEASE COMPLETE, setup.req/SETUP, 2nd.T303/release.ind, setup.req/SETUP, 2nd.T303/release.ind, setup.req/SETUP, CALL PROCEEDING/proceeding.ind, T310/RELEASE & release.ind, RELEASE COMPLETE (or RELEASE)/release.conf, setup.req/SETUP, CONNECT/setup.conf & CONNECT ACKNOWLEDGE, release.req/RELEASE, 1st.T308/RELEASE, RELEASE COMPLETE (or RELEASE)/release.conf, SETUP/setup.ind, release.resp/RELEASE COMPLETE, setup.req/SETUP, CONNECT/setup.conf & CONNECT ACKNOWLEDGE, release.req/RELEASE, RELEASE COMPLETE (or RELEASE)/release.conf, setup.req/SETUP, 1st.T303/SETUP, 1st.T303/SETUP, CONNECT/setup.conf & CONNECT ACKNOWLEDGE, release.req/RELEASE, 2nd.T308/RESTART, 1st.T316 & 2nd.T316/RESTART, 1st.T316 & 2nd.T316/RESTART, 1st.T316 & 2nd.T316/RESTART, RESTART ACKNOWLEDGE/nu, setup.req/SETUP, CALL PROCEEDING/proceeding.ind, CONNECT/setup.conf & CONNECT ACKNOWLEDGE, release.req/RELEASE, RELEASE COMPLETE (or RELEASE)/release.conf, setup.req/SETUP, CALL PROCEEDING/proceeding.ind, T310/RELEASE & release.ind, RELEASE COMPLETE (or RELEASE)/release.conf, SETUP/setup.ind, proceeding.req/CALL PROCEEDING, setup.resp/CONNECT, CONNECT ACKNOWLEDGE/setup.complete.ind, release.req/RELEASE, RELEASE COMPLETE (or RELEASE)/release.conf, SETUP/setup.ind, setup.resp/CONNECT, setup.resp/CONNECT, setup.resp/CONNECT, CONNECT ACKNOWLEDGE/setup.complete.ind, release.req/RELEASE, RELEASE COMPLETE (or RELEASE)/release.conf, SETUP/setup.ind, setup.resp/CONNECT, CONNECT ACKNOWLEDGE/setup.complete.ind, release.req/RELEASE, RELEASE COMPLETE (or RELEASE)/release.conf, SETUP/setup.ind, setup.resp/CONNECT, T313/RELEASE & release.ind, RELEASE COMPLETE (or RELEASE)/release.conf*]

Fig. 2. The resulted test suite derived from the adaptive application of the CPP to Q.2931

Table 3 describes a comparative test sequence length of the existing methods and our method from the experimental result of its application to the B-ISDN Q.2931.

Table 3. A comparative test sequence length from the experimental result of an adaptive application to the B-ISDN Q.2931

Methods	Test Sequence Length
T-method*	25 input/output pairs
W-method	109 input/output pairs
DS-method	67 input/output pairs
UIO-method	48 input/output pairs
Our method	42 input/output pairs

*T-method has an incomplete fault coverage.

5. Conclusions

In this paper, we propose on an adaptive application of the CPP(Chinese Postman Problem) algorithm to the protocol test sequence generation. Also, we present an example of this CPP algorithm to B-ISDN Q.2931 call connection control procedure for the purpose of showing how it can be adapted to generate a test suite for conformance testing of a communication protocol.

From the experimental result of its application to the Q.2931, depicted in Table 3, the proposed method does evidently have a good capability of optimization in a test sequence, while the existing methods such as transition tour (T-method), characterization set (W-method), distinguishing sequence (DS-method), unique input/output (UIO-method) does not provide an optimum length of the test sequence when it may be generated. Thus, our method does shorten the total length of a test sequence because the state identification sequences and test subsequences should be minimized when they might be derived, and also because the total length of a test sequence should be minimum-costed based on the optimization technique when it may be generated.

The proposed method has an advantage of an optimization technique which finds a minimum cost of test suite from a standardized specification, so this optimization technique of the CPP algorithm can be practically applied to a real environment for testing a conformity of a protocol implementation.

REFERENCES

[1] D. Rayner, "OSI Conformance Testing," in Computer Networks and ISDN Systems, vol. 14, no. 1, pp. 79-98, 1987.

[2] D. P. Sidhu and T. K. Leung, "Formal Methods for Protocol Testing: A Detailed Study," in IEEE Transactions on Software Engineering, vol. 15, no. 4, pp. 413-426, 1989.

[3] K. K. Sabnani and A. T. Dahbura, "A Protocol Test Generation Procedure," in Computer Networks and ISDN Systems, vol. 15, no. 4, pp. 285-297, 1988.

[4] Z. Kohavi, Switching and Finite Automata Theory. New York: McGraw-Hill, 1978.

[5] J. Edmonds and E. L. Johnson, "Mathing, Euler Tours, and the Chinese Postman," in Mathematical programming, vol. 5, pp. 88-124, 1973.

[6] Béla Bollobás, Modern Graph Theory, Springer, 1998.

[7] A. Gibbons, Algorithmic Graph Theory, Cambridge: Cambridge Univ. Press, 1985.

[8] R. E. Tarjan, Data Structures and Network Algorithms, Philadelphia: Society for Industrial and Applied Mathematics, 1983.

[9] ITU-T Recommendation Q.2931, Broadband Integrated Services Digital Network (B-ISDN) User-Network Interface (UNI) Layer 3 Specification for Basic Call/Connection Control, 1995.

[10] ATM Forum Technical Committee, ATM User-Network Interface (UNI) Signalling Specification, Version 4.0, 1996.
[11] Uyless Black, ATM, Voulme II Signalling in Broadband Networks, Prentice Hall, 1998.
[12] Harry J. R. Dutton and Peter Lenhard, Asynchronous Transfer Mode (ATM) Technical Overview, Prentice Hall, 1995.

Author Biography

Chul Kim **[Regular member]**

- Feb. 1977 : Yonsei Univ., Electronics Engineering, B.S.
- Aug. 2000 : Yonsei Univ., Computer Science, Ph.D.
- Jul. 1981 ~ Jan. 1984 : Samsung LtD, Researcher
- Feb. 1984 ~ Jan. 1993 : IBM Korea, Engineer
- Mar. 1994 ~ current : Yongin Univ., Dept. of Computer Science, Professor

〈Research Interests〉
Protocol Engineering, Computer Security