

一种基于多 UIO 的一致性测试序列优化方法^{*}

吴永^{1,2} 贺前华² 涂用军^{1,2}

(华南理工大学电子与信息学院 广州 510640)¹ (广东科学技术职业学院电子工程系 广州 510640)²

摘要 在通信协议一致性测试中,设计出测试费用低、检错能力强的测试序列是关键工作之一。文章基于乡间中国邮递员遍历算法,利用网络流技术从一个状态的多 UIO 序列中选择适当的一个,使有向图 G^{test} 的度: $\Delta(G^{test}) = \sum_{i=1}^n |\xi(v_i)|$ 最小,从而达到减少测试序列的总时间费用的目的。

关键词 一致性测试,测试序列,多 UIO 序列,有限状态机

An Optimization Method for Protocol Conformance Test Sequence Based on Multiple UIO

WU Yong^{1,2} HE Qian-Hua² TU Yong-Jun^{1,2}

(Dept. of Electronic Engin., Guangdong Vocational Institute of Science and Technology, Guangzhou 510640)¹

(College of Electronic and Information Engineering, South China Univ. of Sci. and Tech., Guangzhou 510640)²

Abstract A key problem in protocol conformance test is to design test sequence which cost low and detect fault easily. Based on the traversal algorithm of rural Chinese postman, a suitable UIO sequence is chosen from multiple UIO sequence by using the technology of network stream, which makes the measurement of: $G^{test}; \Delta(G^{test}) = \sum_{i=1}^n |\xi(v_i)|$ smallest and helps to reduce the cost of the test sequence.

Keywords Conformance test, Test sequence, Multiple UIO sequences, Finite-state machine

由于不同的协议软件开发者对协议标准的理解不同,实现过程又可能存在这样或那样的错误,因此我们对完成的协议实现必须进行一致性测试(Conformance Testing)。协议一致性测试已成为通信和分布式系统协议工程中非常活跃的领域之一。当前,协议一致性测试的研究主要有两个方面:测试系统的构造和测试序列的生成。其中最困难的是如何生成检错能力最强、测试费用最小的测试序列。

论文首先介绍一种根据 UIO 获得最小长度测试序列的算法,即基于乡间中国邮递员遍历算法生成测试序列的方法。然后对它加以改进,得到测试费用更低的一种优化测试序列算法。

1 预备知识

设 $G=(V,E)$ 是协议有限状态机 CFSM 的有向图表示,图的每边表示为三元组 $(v_i, v_j; L_k)$, 对应的时间费用是 $C(v_i, v_j; L_k)$ 。

G 的一个行程(Walk)定义为一个有限的非空连续边序列: $W=(v_1, v_2; L_1)(v_2, v_3; L_2) \cdots (v_{i-1}, v_i; L_{i-1})$, 行程的总时间费用等于各边时间费用的总和: $\sum_{j=1}^{i-1} C(v_j, v_{j+1}; L_j)$ 。

一个遍历(Tour)定义为一个开始和终止于同一个顶点的行程; G 的一个欧拉遍历(Euler tour)定义为包含 E 中每边恰好一次的一个遍历; G 的一个邮递员遍历(Postman tour)则定义为包含 E 中每一边至少一次的一个遍历; G 的一个费用最小的邮递员遍历叫中国邮递员遍历(Chinese postman tour); E 的一个子集 E_c 的一个邮递员遍历叫乡间邮递员遍

历(Rural postman tour)^[3]。

如果有向图 G 中的任何一对不同顶点 v_i 和 v_j 之间都存在一个起点为 v_i , 终点为 v_j 的行程, 则称 G 是强连通的。如果 G 对应的无向图是连通的, 则称 G 是弱连通的。

由 E 的一个子集 E_c 导出的一个子图叫 G 的边导子图^[2], 子图顶点是 E_c 中边端点, 记作 $G[E_c]$ 。

顶点 v_i 的入度和出度分别表示为 $d_{in}(v_i)$ 和 $d_{out}(v_i)$ 。如果图 G 的每一个顶点 v_i 满足 $d_{in}(v_i) = d_{out}(v_i)$, 则称 G 是对称的。

推论 1 有向图 G 有欧拉遍历当且仅当 G 是强连通且对称的^[4]。

2 根据 UIO 获得最小时间费用测试序列的一种算法

文[1]中,作者给出了一种根据 UIO 序列获得最小时间费用测试序列的一种算法。算法步骤如下:

Step-1: 有协议有限状态机 CFSM 对应的图 G , 构造有向图 $G^{test}=(V^{test}, E^{test})$, 其中 $V^{test} \equiv V$, $E^{test} \equiv E \cup E_{test}$, $E_{test} = \{(v_i, v_k; L_i @ UIO_j) : (v_i, v_j; L_i) \in E, Tail(UIO_j) = v_k\}$ 。

$\forall (v_i, v_k; L_i @ UIO_j) \in E_{test}$, 其费用表示为: $C(v_i, v_k; L_i @ UIO_j) = C(v_i, v_k; L_i) + C(UIO_j)$ 。

Step-2: 由 G^{test} , 构造有向图 $G^*=(V^*, E^*)$, 其中 $V^* \equiv V^{test}$, E^* 通过复制每一边 $(v_i, v_j; L_k) \in E^{test}$ 共 $\chi(v_i, v_j; L_k)$ 次得到。要求 E 中的边在 E^* 中出现 0 次或多次, 而 E_{test} 中的每一边至少在 E^* 中出现一次, 次数多少在于保证 E^* 中边的总费用: $C(\chi) = \sum_{(v_i, v_j; L_k) \in E^{test}} C(v_i, v_j; L_k) \chi(v_i, v_j; L_k)$ 最小, 且 E^*

^{*} 国家自然科学基金资助项目(60172064); 高等学校博士学科点专项基金资助项目。吴永 博士研究生, 工程师, 主要研究方向: 通信系统及其软件测试理论与方法、信息处理; 贺前华 教授, 博士生导师, 主要研究方向: 通信系统及信息处理、集成电路设计。

中每一顶点的入度等于出度。 G^* 叫做 G^{test} 的对称增广^[1]。

定理 1 如果边导子图 $G[E_{test}]$ 是图 G^{test} 的弱生成图, 那么 G^{test} 的任何对称增广 G^* 是强连通的, 从而有欧拉遍历^[3]。

Step-3: 找出 G^* 的一个欧拉遍历。

定理 2 G^{test} 对称增广 G^* 的一个欧拉遍历 P 对应 G^{test} 的一个中国邮递员遍历^[3]。

由前所述, 求解协议的最小费用测试序列转化为求 G^{test} 的覆盖 E_{test} 的一个中国邮递员遍历, 进一步转化为求 G^{test} 的对称增广 G^* 的一个欧拉遍历, 因此最终转化为求 G^{test} 对称增广 G^* 。而由 G^{test} 求对称增广 G^* , 关键是确定 $\chi(v_i, v_j; L_k)$ 的取值。

对于 $\forall v_i \in G^{test}$, 设 $d_{in}^{E_{test}}(v_i) = |\{(v_j, v_i; L_k) : (v_j, v_i; L_k) \in E_{test}\}|$, $d_{out}^{E_{test}}(v_i) = |\{(v_i, v_j; L_k) : (v_i, v_j; L_k) \in E_{test}\}|$, 定义索引 $\xi(v_i) = d_{in}^{E_{test}}(v_i) - d_{out}^{E_{test}}(v_i)$ 。

下面的两步可确定 $\chi(v_i, v_j; L_k)$ 的取值:

Step-1: 从 G^{test} 构造有向图 $G_F = (V_F, E_F)$, $V_F = V^{test} \cup \{s, t\}$, s 和 t 分别为 G_F 的 start 和 sink 顶点, $E_F = E^{test} \cup \{(s, v_i) : v_i \in C\} \cup \{(v_j, t) : v_j \in D\}$, 这里 $C \subseteq V_F$ ($D \subseteq V_F$) 是 G^{test} 中具有非 0 索引的顶点集。为简便计, E 中的自环在 E_F 被删除。当 G^{test} 中顶点对 (v_i, v_j) 之间如果有多条边, E_F 只取费用最小的一边。设边 (s, v_i) 费用为 0 和容量^[5] $\gamma(s, v_i) \equiv \xi(v_i)$, 边 (v_j, t) 费用为 0, 容量为 $\gamma(v_j, t) \equiv -\xi(v_j)$, E_F 中其余的边在 G_F 有和 G^{test} 一样的费用和无限容量。

Step-2: 定义函数:

$$\chi(v_i, v_j; L_k) = \begin{cases} F(v_i, v_j; L_k) + 1, & \text{if } (v_i, v_j; L_k) \in E_{test} \\ F(v_i, v_j; L_k), & \text{if } (v_i, v_j; L_k) \in E \end{cases}$$

F 为 G_F 的最小费用最大流。 F 的值由最小费用最大流算法确定^[5]。

定理 3 函数 χ 形成了 G^{test} 的最小费用对称增广^[3]。

3 多 UIO 序列优选算法

前面介绍的方法只用到了顶点的一个 UIO 序列。注意到一个顶点的 UIO 序列可能不止一个, 用最佳方法从这些 UIO 序列构成的最终测试序列可减少总时间费用。

从前面可以发现一个重要现象, G^{test} 顶点索引变量的减少导致了相应的测试序列总时间费用的减少。理想情况下, G^{test} 是对称的, 不必求对称增广 G^* , 此时, E 中的边不必包含在 G^* 且测试序列的费用就是 E_{test} 中的边的费用。

定义 G^{test} 的度为: $\Delta(G^{test}) = \sum_{i=1}^n |\xi(v_i)|$, 设状态 S_j 有 $r_j > 0$ 个具有不同的尾状态最小长度 UIO 序列。多 UIO 序列的选择问题如下:

给定 $G, MUO_i = \{UIO_i^1, \dots, UIO_i^r\}$ 是顶点 v_i ($i = 1, 2, \dots, n$) 的 UIO 序列集。对每一边 $(v_j, v_i; L_i) \in E$, 找出一 $UIO_i^r \in MUO_i$, 使得 $\Delta(G^{test})$ 最小。

已知 G 中顶点 v_i 的入度和出度分别为: d_{in}^E 和 $d_{out}^E(v_i)$ 。考虑 G 中入度为 $d_{in}^E(v_i)$ 的顶点 v_i , MUO_i 恰好有 $d_{in}^E(v_i)$ 个 UIO 序列必须分配给进入 v_i 的边; 每一边 $(v_j, v_i; L_i) \in E$ 产生一边 $(v_i, v_k) \in E_{test}$, 其中 $v_k = Tail(UIO_i^r)$, ($UIO_i^r \in MUO_i$)。因此多 UIO 的选择问题转化为对每一边 $(v_j, v_i; L_i) \in E$, 选择一个适当的值 α , 使得 $\Delta(G^{test})$ 最小。

给定 G , 构造 $G_M = (V_M, E_M)$, 其中:

$$V_M = \{s, t\} \cup V_L \cup V_R, V_L = V_R = V = \{v_1, \dots, v_n\};$$

$$E_M = E_s \cup E_T \cup E^M; E_s = \{(s, v_i) : v_i \in V_L\};$$

$$E_T = \{(v_j, t) : v_j \in V_R\};$$

$$E^M = \{(v_i, v_j) : \text{存在 } UIO_i^r, \text{ 使得 } HEAD(UIO_i^r) = v_i, Tail(UIO_i^r) = v_j\}.$$

设每一边 $(s, v_i) \in E_s$ 有 0 时间费用和容量 $\gamma(s, v_i) = d_{in}^E(v_i)$, 边 $(v_j, t) \in E_T$ 有 0 时间费用和容量 $\gamma(v_j, t) = d_{out}^E(v_j)$, $(v_i, v_j) \in E^M$ 有 $C(UIO_i^r)$ 时间费用和无穷容量。

G_M 上的一个流 F 定义为一个函数 $F: E_M \rightarrow Z^+$, Z^+ 是一个非负整数集合, 并满足如下条件:

$$(1) \forall v_i \in V_L, F(s, v_i) = \sum_{(v_i, v_j) \in E^M} F(v_i, v_j);$$

$$(2) \forall v_i \in V_R, F(v_j, t) = \sum_{(v_i, v_j) \in E^M} F(v_i, v_j);$$

$$(3) \forall (s, v_i) \in E_s, F(s, v_i) \leq \gamma(s, v_i);$$

$$(4) \forall (v_j, t) \in E_T, F(v_j, t) \leq \gamma(v_j, t).$$

流 F 的费用是: $C(F) = \sum_{v_i, v_j \in E^M} C(UIO_i^r) F(v_i, v_j)$ 。

给定 G_M 上一个最小费用最大流 F^* , 使用如下多 UIO 选择算法 Algorithm-select-MUIO 可构造 G^{test} 。

算法 Algorithm-select-MUIO;

Step-1: $\forall v_i \in V$, 计算 $F^*(v_i, v_j)$, $j = 1, 2, \dots, n$ 及 $|UIO_i^r|$, $\alpha = 1, 2, \dots, r$ 。

Step-2: 将 $F^*(v_i, v_j)$ 由大到小排列:

$$F^*(v_i, v_{j_1}) \geq \dots \geq F^*(v_i, v_{j_{d_{in}^E(v_i)}}) \geq \dots \geq F^*(v_i, v_{j_n}), \text{ 将}$$

$|UIO_i^r|$ 由小到大排列: $|UIO_i^1| \leq \dots \leq |UIO_i^r|$ 。

Step-3: 求顶点 v_i 入边集 $E_{in}(v_i) = \{(v_j, v_i), j = 1, 2, \dots, d_{in}^E(v_i)\}$, 将 UIO_i^1 赋给 E_{in} 中的 $F^*(v_i, v_{j_1})$ 条不同边, UIO_i^2 赋给 E_{in} 中的 $F^*(v_i, v_{j_2})$ 条不同边, 直到 E_{in} 中每边都赋给了一个 $UIO_i^r \in MUO_i$ 。 $(v_i, v_j) \in E^M$ 是 G_M 中的边并表示 UIO_i^r 。

4 算法举例

假设 SDL 描述的 CFSM 对应的有向图 $G = (V, E)$ 如图 1(a) 所示, 各边费用相等, “ri/null” 表示重置边。由 G 构造有向图 $G_M = (V_M, E_M)$ 如图 1(b) 所示。CFSM 顶点 v_i 的多 UIO 如表 1 所示。下面以此为例来解释多 UIO 优选算法。

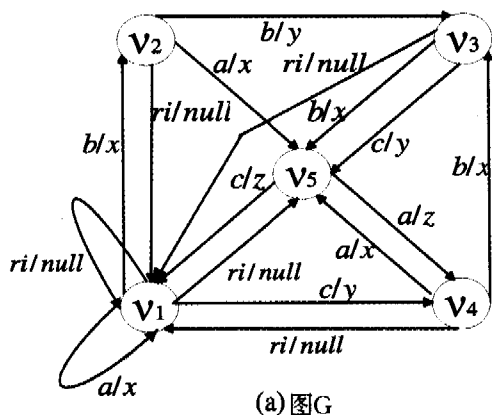
给定 G_M 上的最小费用最大流 F^* , 对顶点 v_1 求得: $F^*(v_1, v_1) = 1, F^*(v_1, v_2) = 3, F^*(v_1, v_3) = 2, F^*(v_1, v_4) = 0, F^*(v_1, v_5) = 1$, 如图 1(b) 中 E^M 上边的数字所标。由大到小排列为:

$$F^*(v_1, v_2) > F^*(v_1, v_3) > F^*(v_1, v_1) = F^*(v_1, v_5) > F^*(v_1, v_4),$$

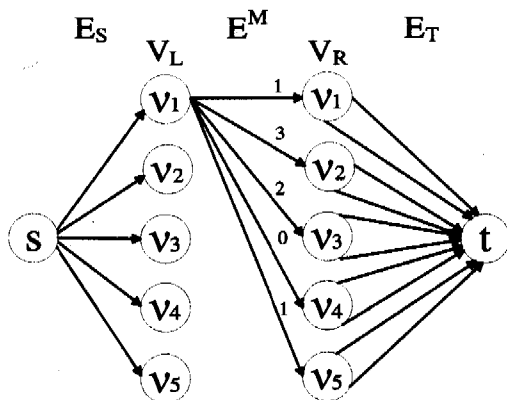
$$d_{in}^E(v_1) = 7, \text{ 而 } |UIO_1^1| = |UIO_1^2| = |UIO_1^3| = |UIO_1^4| = |UIO_1^5| = 2.$$

顶点 v_1 的入边集 $E_{in}(v_1) = \{(v_1, v_1; a/x), (v_2, v_1; ri/null), (v_3, v_1; ri/null), (v_4, v_1; ri/null), (v_5, v_1; c/z), (v_5, v_1; ri/null)\}$ 。由于 $|UIO_i^1|$ ($i = 1, 2, \dots, 5$) 相同, 故将 UIO_1^1 赋给 $\{(v_1, v_1; a/x), (v_2, v_1; ri/null), (v_3, v_1; ri/null)\}$, UIO_1^2 赋给 $\{(v_1, v_1; ri/null), (v_4, v_1; ri/null)\}$, UIO_1^3 赋给 $\{(v_5, v_1; c/z)\}$, UIO_1^4 赋给 $\{(v_5, v_1; ri/null)\}$ 。

其它顶点的做法和顶点 v_1 一样。由于 V_L 中每个顶点到 V_R 中某些顶点及 V_R 中每个顶点到 t 存在无穷容量, $F^*(s, v_i) = \gamma(s, v_i) = d_{in}^E(v_i)$, $v_i \in V_L$, 这就保证了 G 中每个顶点都赋给了一个 UIO 序列。



(a) 图G



(b) 图GM

图 1

表 1 CFSM 各顶点的多 UIO 序列

状态	UIO 序列	UIO 尾状态	MUIO
v_1	a/x a/x	v_1	UIO_1^1
	a/x b/x	v_2	UIO_2^1
	c/y b/x	v_3	UIO_3^1
	a/x c/y	v_4	UIO_4^1
	b/x a/x	v_5	UIO_5^1

5 算法仿真实验

算法仿真是在 Simu-BlueTSGT 实验环境下进行的。

(上接第 259 页)

显然,分别使基于历史的局部搜索法和利用辅助表实现的快速查找算法比从原始查找方法快了好多。在生成 A2B 的数据时候,基于历史的局部搜索法显然优于利用辅助表实现的快速查找算法。但是当还需生成 B2A 数据的时候,前者却比后者慢得多。究其原因,是因为在使用前者的时候,一旦发现所要的数据没有在邻域当中时,算法必须查找整个非均匀三维查找表的四面体。鉴于此种情况,结合两种算法,即在生成 A2B 数据的时候,采用基于历史的局部搜索法;在生成 B2A 数据的时候,采用利用辅助表实现的快速定位算法。最后整个的生成过程所需的时间为 18.532s。

结论 考虑到生成 Profile 数据之间的相关性,本文提出了基于历史的局部搜索算法,该算法利用历史数据的信息来加速生成新数据,从而有效提高了 Profile 的生成速度。利用辅助表的快速定位方法,避开一部分判断四面体的开销,通过粗定位和细定位进一步提高了生成 Profile 数据的效率。但

Simu-BlueTSGT 是 BlueTSGT 系统的一个仿真系统,直接调用 C 语言书写 CFSM 对应的有向图的中间形式。为简化仿真,在仿真实验中,直接输入本文算法解释实例的有向图的中间形式,仿真结果如下:

启动多 UIO 优选算法生成的测试序列为:

ri/null@a/x@a/x@a/x@b/x@b/x@b/x@c/z@ri/null
@a/x@b/x@ri/null@a/x@b/x@a/x@c/z@c/y@b/x@c/y
@a/z@b/x@c/y@ri/null@b/x@a/x@c/z@c/y@b/x@ri/
null@a/x@b/x@b/y@b/x@a/z@a/x@a/z@b/x@b/x@a/
z@ri/null@c/y@b/x@c/y@c/z

不启动多 UIO 优选算法生成的测试序列为:

ri/null@b/x@b/y@c/y@c/z@a/x@b/x@a/x@a/z@
b/x@c/y@c/z@b/x@a/x@ri/null@b/x@a/x@a/z@a/x@
c/y@c/z@b/x@c/y@a/z@b/x@b/x@c/z@b/x@b/y@b/x
@c/z@b/x@a/x@c/z@b/x@ri/null@b/x@a/x@a/z@b/x
@ri/null@b/x@a/x@a/z@b/x@b/x@c/z@ri/null@b/x@
a/x@a/z@ri/null@b/x@a/x@c/z

长度分别为 44 和 55。

可见,简化后的算法确实优化了测试序列的费用。

结论 本论文参考了基于乡间中国邮递员遍历算法利用状态的 UIO 序列获得测试序列的方法,考虑到使用此方法生成测试序列时, G^{test} 的度: $\Delta(G^{test}) = \sum_{i=1}^n |\xi(v_i)|$ 的减少导致了相应的测试序列总时间费用的减少,以及一个状态的 UIO 序列不止一个的事实,提出了一种给每个状态选择一个适当的 UIO 序列的优选算法,使得 $\Delta(G^{test})$ 最小,从而使得最终生成的测试序列总时间费用减少,达到优化测试序列费用的目的。

参考文献

- 1 Aho A V, Dabura A T, Lee D, Uyar M U. An optimization technique for protocol conformance test generation based on UIO sequences and rural Chinese postman tours. In: Protocol Specification, Testing and Verification, VIII, S. Aggarwal and K. K. Sabnani, eds. New York: Elsevier North-Holland, 1998. 75~86
- 2 Bondy J A, Murty U S R. Graph Theory With Applications. New York: Elsevier North Holland, 1976
- 3 Aho A V, Dabura A T, Lee D, Uyar M U. An optimization technique for protocol conformance test generation based on UIO sequences and rural Chinese postman tours. In: Proc. IEEE INFOCOM, 1997
- 4 Miller R E, Paul S. On the Generation of Minimal-Length Conformance Tests for Communication Protocols. In: Proc. of IEEE Int'l Communication Conference. Atlanta: IEEE Computer Society, 1993
- 5 周炯磐著. 通信网理论基础. 北京邮电大学出版社, 1991. 124~134

需要注意的是在基于历史的局部搜索法中邻域确定得不合适可导致搜索时间的增加,因此下一步的工作重点可放在如何确定适当的邻域来最大化该算法的效率。

参考文献

- 1 Kang H R. Printer-Related Color Processing Techniques. In: Proceedings of SPIE, 1995, 2413, 410~419
- 2 Kang H R. Comparisons of Three-Dimensional Interpolation Techniques by Simulations. In: Proceedings of SPIE, 1995, 2414, 104~114
- 3 International Color Consortium. ICC Profile Specification. USA: ICC, 2001
- 4 史瑞芝. 用查表插值法实现 CIE Lab 与 CMYK 颜色空间的转换. 测绘学院学报, 2002, 19(3): 224~226
- 5 乔新亮. 面向彩色打印的 3D-LUT 优化技术. [学位论文]. 西安: 西安电子科技大学, 2002
- 6 Sonka M, Hlavac V, Boyle R. Image Processing, Analysis, and Machine Vision. (2nd Version). Beijing: PPTPH & Thomson Learning, 2002