表格语言的分析比较

陈怡海 缪淮扣

（上海大学计算机工程与科学学院 上海 200072）（上海市计算机软件评测重点实验室 上海 201114）

摘 要 表格语言具有可读性和可理解性的优点，它能非常精确地表示软件系统需求。在过去的30多年间，表格语言已经成功地应用于多个安全关键嵌入式软件的开发中。准确了解这些表格语言的特性，对表格语言的研究及推广有重要的指导意义。对3种不同的表格语言进行了详细的综述和讨论，从不同角度分析比较了其异同点，并提出了进一步的研究方向。

关键词 表格方法，形式方法，规格说明验证与确认，工具支持

中图法分类号 TP311.5 **文献标识码** A

Comparison of Tabular Notations

CHEN Yi-hai MIAO Huai-kou

(School of Computer Engineering and Science, Shanghai University, Shanghai 200072, China)

(Shanghai Key Laboratory of Computer Software Evaluating & Testing, Shanghai 201114, China)

Abstract Tabular notations are both readable and convenient. They allow representing the specifications of systems in a very compact and precise manner. They also make checking such important properties as consistency and completeness natural and relatively easy. In the past 30 years, tabular notations have been successfully applied in several safety-critical software systems. This paper presented a fairly comprehensive survey comprising three variants of tabular notations. The paper analyzed and compared all these three variants of tabular notation in details. Moreover, the paper discussed the challenges behind using tabular notations to derive an implementation of a working real-time system and presented some solutions. Finally it also attempted to help the reader navigate the vast literature in the field, to highlight differences and similarities between variants, and to reveal research trends and promising avenues for future exploration.

Keywords Tabular notations, Formal methods, Specification verification and validation, Tool support

1 引言

随着计算机硬件设备性能的迅速提高以及嵌入式系统应用领域的不断拓宽，嵌入式系统软件的规模和复杂性急剧增加，软件已经成为嵌入式系统的主要使能部件。各类嵌入式系统广泛地应用于我们生活的各个领域，例如飞机控制系统、医疗系统、汽车控制系统、核电站系统等，然而这些系统的失效会产生灾难性的后果，例如法国阿丽亚娜五型运载火箭的发射失败，造成约三亿六千万美元的经济损失和一年的发射计划延期[1]。在1985年至1987年期间，由于软件的故障，Therac 25放射治疗仪造成了6起医疗事故并导致了3名病人由于接受了过量辐射而死亡[2]。由于安全气袋软件的设计错误，如果前排副驾驶座没有乘客，当发生碰撞时，这一侧的头部气囊不会打开，致使后排乘客在车辆发生碰撞时遭遇危险，因此2011年通用汽车公司召回了50500辆凯迪拉克SRX豪华SUV轿车[3]。

众所周知，大多数软件错误是在需求阶段产生的，而修正需求阶段的错误所需的代价非常昂贵。Boehm的研究表明：改正在产品交付应用后所发现的一个需求方面的缺陷比在需求阶段改正这个错误要多付出10～100倍的成本[4]。目前软件开发过程的不尽人如意的“技术水平”在于我们没能开发良好的设计文档，而质量糟糕的文档导致许多错误产生并降低了软件开发和使用效率[5]。因此，提高安全关键嵌入式软件开发质量的关键在于提高需求文档的质量并尽早发现各类需求错误，而复杂嵌入式软件的需要又特别难以说明和确认。在过去的数十年间，研究人员提出了多种基于数学的形式方法来提高需求文档的质量，虽然形式方法在学术界和工业界得到了推广和应用，但是形式规格说明往往较难理解和验证。表格语言(Tabular notations)或简称为表格(Tables)，最早在1977年由David Lorge Parnas教授首先提出，它是一种采用多维可视化表格化结构来表示数学表达式的形式化定义方法。表格方法已经成功地应用于美国空军A-7E飞机的需求规格说明的构造和验证[6]，在加拿大安大略省惠灵顿核电站关机系统的安全认证项目中[7]，目前已有越来越多的软件开发人员应用表格表达式来对各类软件系统进行建模。在过去的30多年间，研究人员提出了多种基于表格表达式的规格说

到稿日期：2013-05-23 返修日期：2013-08-02 本文受国家自然科学基金项目(61073050，61170044)资助。

陈怡海(1972－)，男，博士，副教授，主要研究领域为软件工程、软件认证等，E-mail：yhchen@staff.shu.edu.cn；缪淮扣(1953－)，男，教授，博士生导师，主要研究领域为形式方法、软件需求工程、软件测试、软件体系结构、软件模式等。

明方法。准确了解这些基于表格表达式的建模方法的特性，对表格方法的研究及推广有重要的指导意义。本文首先介绍了表格方法的历史背景知识及其基本原理，针对现有的不同的基于表格的规格说明技术，从多个角度进行分析比较，指出各自的优缺点，并提出了进一步的研究方向。本文第 2 节至第 6 节，分别从语法、语义、应用和工具支持等不同角度介绍这 3 种不同的表格语言；最后总结全文并指出了进一步的研究内容和方向。

2 Parnas 表格方法

表格表达式能方便地表示复杂的数学表达式，与传统的数学表达式相比，表格表达式更容易理解和使用。这种方法最早由 Parnas 教授提出并得到了成功的应用，因此被称为 Parnas 表格或表格表达式(Tabular Expressions)。Parnas 教授所倡导的表格方法，其核心思想包含 3 个方面内容：(1)文档驱动的软件开发方法，软件开发首先从定义精确的需求文档开始[5]；(2)采用函数/数学关系定义文档内容；(3)使用表格表达式对需求进行描述。这 3 个方面是缺一不可的有机的整体，而表格表达式是 Parnas 方法的外在可视部分。

文档是指具有官方地位或权威并可作为证据使用的文字描述，而文档驱动的软件开发方法是指采用精确(数学的)、高度结构化的文档来记录软件设计决策，并指导检验人员和测试人员进行验证和测试，对程序员进行约束，并且在软件交费使用以后，可作为软件维护人员的参考文档。完整的文档需要包括“系统需求文档”、“系统设计文档”、“软件需求文档”、“软件行为文档”、“模块接口规格说明”、“模块内部设计文档”、“用户关系文档”、“软件测试规格说明”等[5]。

文档驱动方法的核心是采用四变量模型作为状态机模型对系统需求进行形式定义[8]。图 1 为四变量模型的示意图，它将目标系统的行为表示为 4 种变量之间的数学关系，这 4 种变量分别是：系统监测变量(MON)、系统控制变量(CON)、系统从输入设备中读入的变量(INPUT)、系统向输出设备输出的变量(OUTPUT)。

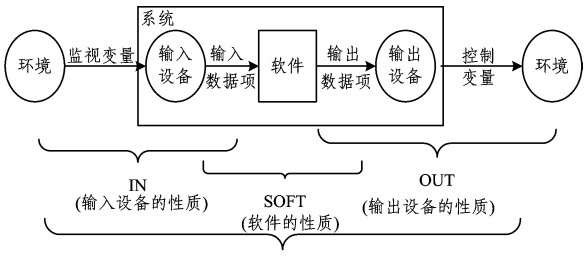

图 1 四变量模型示意图

四变量模型中，变量之间的关系由 REQ、NAT、IN 和 OUT 4 个关系进行定义。NAT 关系描述环境对系统行为的自然约束，REQ 关系将系统需求定义为系统需要建立或维护的监视变量和控制变量之间的关系。IN 关系定义了监视变量和输入变量之间的关系，而 OUT 关系则定义了输出变量和控制变量之间的关系。整个软件系统的性质 SOFT 可以从 REQ、NAT、IN 和 OUT 中蕴含推导出来。

可采用 Parnas 表格对上述关系进行描述。Parnas 表格主要有 10 种不同类型，常见类型有正规函数表、反转函数表、向量函数表等[9,10]。图 2 上方定义的函数，其对应的语义等价的表格表达式显示在下方。

$$f(x,y)=\begin{cases} x^2+y^2, & (((x<1)\wedge(y<0))\vee((x=1)\wedge(y=0))\vee((x>1)\wedge(y>0))) \\ x^2-y^2, & (((x=1)\wedge(y<0))\vee((x>1)\wedge(y=0))\vee((x<1)\wedge(y>0))) \\ x+y, & (((x>1)\wedge(y<0))\vee((x=1)\wedge(y>0))\vee((x>1)\wedge(y<0))) \end{cases}$$

$f(x,y)=$

	$x<1$	$x=1$	$x>1$
$y<0$	x^2+y^2	x^2-y^2	$x+y$
$y=0$	$x+y$	x^2+y^2	x^2-y^2
$y>0$	x^2-y^2	$x+y$	x^2+y^2

图 2 表格表达式

已有多种不同的方法对表格表达式的形式语义进行了定义。Parnas 最早给出了 10 种表格表达式的形式语法和语义[10]；Janicki 等人在关系演算基础上，定义了表格表达式的语义模型[11-13]；为了能对表格表达式进行形式推理和表达式转换，Kahl 提出了基于复合的表格形式语法和语义[14]。然而这些方法局限于某些类型的表格表达式，缺乏可扩展性。在前人研究工作基础上，金英教授和 Parnas 教授[15,16]定义了表格表达式的通用模型，该模型可以将任何类型的表格表达式转换为等价的一般表达式。一个具体的表格表达式类型可以定义为：

- 表达式的具体构成定义：给出表格表达式的维数、条件网格、结果网络等表格构成信息。
- 限制方案(restriction schema)：是一个谓词表达式，用于描述这一类表格表达式共同具有的约束条件，包括网格个数、网络的索引集合和表的索引集合、包含的表达式类型，以及其他需要满足的条件约束谓词。
- 辅助函数(auxiliary functions)：一些函数定义，这些函数将在定义约束谓词和求值方案中用到。
- 求值方案(evaluation schema)：是一个扩展的表达式，用于描述该类表达式的计算过程的策略或者方案，它给出了对一个具体的表达式类型如何进行求值的形式语义。

关于表达表达式通用模型的进一步定义，可以参考文献[6,7]。

在表格支持工具方面，麦克马斯特大学的研究人员开发了表格工具集 TTS[17]。TTS 工具集具有表格创建、编辑、一致性分析等功能，由于开发时间较早，已不再对其进行维护。Peters 等人开发了基于 Eclipse 插件的表格编辑原型工具[18]，工具采用基于 XML 的开放数学文档格式(OMDoc)作为表格的形式模型，并使用 XSLT 技术将 XML 格式表格表达式输出到 PVS 定理证明工具。近期，Mark 等人[19]在 Matlab/Simulink 的基础上开发了图形化表格编辑工具，工具支持一维和二维表格输入、单个/多个输出，支持代码生成，并集成了 CVC3 SMT 求解器和 PVS 定理证明工具来检查表格的不相交性和完备性条件。

3 SCR 方法

软件成本降低方法 SCR(Software Cost Reduction)[20-23]是由 David Parnas 教授和美国海军研究实验室 C. L. Heitmeyer教授领导的开发小组开发的对实时嵌入式安全关键系统进行说明分析的形式方法，它是一种基于表格表达式、事件驱动的需求定义方法。从 1978 年至今，SCR 方法已经成功地应用到各种不同类型的嵌入式安全关键控制系统的开发过程

中，包括航空系统、空间站系统、电话网络系统、核电站控制系统等。

SCR 方法同样基于四变量模型，然而 SCR 方法只采用 NAT 关系和 REQ 关系来定义系统的行为。在 SCR 方法中，除了监视变量和控制变量以外，还引入了模式、模式类、条件、事件和项这些结构来表示系统的行为。模式是系统状态集合，模式类是系统状态集合的划分，每个划分称为该类中的一个模式。条件是定义在监视变量、项或模式上的一个谓词。当一个条件的值从“真”变为“假”时，产生一个事件，反之亦然。引入符号“ @T(c)”用于表示条件 c 为真，而符号“ @F(c)”则表示条件 c 为“假”。项是在监视变量、模式、其他项上定义的辅助函数。

SCR 方法采用了 3 种不同的表格对系统进行描述，分别是条件表格(Condition tables)、事件表格(Events tables)和模式转换表格(Mode transition tables)。每一种表格将一个独立变量(控制变量、模式类或者是项)定义为一个数学函数。

条件表格将变量定义为系统的模式和条件的函数。表 1 显示了典型的条件表格。表 1 的每一列显示了系统的一个特定模式 m_j，表格中在“条件”标题下面的第 k 列说明了当条件 $c_{j,k}$ 成立时 r_i 的值为 v_k。

表 1　条件表格的典型格式

Modes	Conditions				
m_1	$c_{1,1}$	…	$c_{1,k}$	…	$c_{1,p}$
…	…	…	…	…	…
m_j	$c_{j,1}$	…	$c_{j,k}$	…	$c_{j,p}$
…	…	…	…	…	…
m_n	$c_{n,1}$	…	$c_{n,k}$	…	$c_{n,p}$
r_i	v_1	…	v_k	…	v_p

事件表格将变量定义为模式和事件的函数，表 2 显示了一个事件表格。在表 2 中，当系统处于特定的模式 m_j 时，如果事件 $e_{j,k}$ 发生，r_i 的值由 v_k 说明。

表 2　事件表格的典型格式

Modes	Events				
m_1	$e_{1,1}$	…	$e_{1,k}$	…	$e_{1,p}$
…	…	…	…	…	…
m_j	$e_{j,1}$	…	$e_{j,k}$	…	$e_{j,p}$
…	…	…	…	…	…
m_n	$e_{n,1}$	…	$e_{n,k}$	…	$e_{n,p}$
r_i	v_1	…	v_k	…	v_p

模式转换表用于说明模式类的转换关系，将系统的下一个模式定义为当前模式和事件的函数。表 3 显示了一个典型的事件转换表。每一行描述了一个激活表格从左侧的模式 m_i 到右侧的模式 m_{ij} 的转换。

表 3　模式转换表的典型格式

Current Modes	Event	New Modes
m_1	$e_{1,1}$	$m_{1,1}$
	…	…
	$e_{1,i}$	$m_{1,i}$
	…	…
	e_{1,k_1}	m_{1,k_1}
…	…	…
m_n	$e_{n,1}$	$m_{n,1}$
	…	…
	$e_{n,i}$	$m_{n,i}$
	…	…
	e_{n,k_1}	m_{n,k_1}

经过多年开发，SCRtool 是一个成熟的 SCR 方法支持工具集[21]，工具集中有 4 个基本工具：规格说明编辑器、一致性检测器、依赖图浏览器和模拟器，规格说明编辑器的功能是创建表格规格说明，而一致性检测器可用于检查规格说明的语法和类型正确性、可确定性、案例覆盖等性质。依赖图浏览器可以显示规格说明中各个部分之间的依赖关系，模拟器对系统进行模拟运行，显示规格说明中的监视变量、控制变量、术语和模式类，以确认规格说明符合用户意图。

4　OPG 表格

20 世纪 90 年代中期，加拿大安大略电力公司(OPG)首次使用计算机来控制惠灵顿核电站的关机系统，关机系统是一个实时安全关键嵌入式控制系统，因此其必须满足相关的安全技术标准。为了能获得核能监管委员会的运行许可，在 Parnas 教授的指导下，安大略电力公司在项目中使用了表格方法，该项目是加拿大历史上首次大规模使用形式化方法，表格方法在惠灵顿核电项目中获得极大成功，关机系统成功运行至今没有发现一个错误[7]。事实上，OPG 表格是一种特殊类型的 Parnas 表格，其主要有 4 种：水平条件表格、垂直条件表格、状态转换表格和结构化判定表[24]。图 3 显示了最常用的水平条件表格。

	Condition	*Result* f_name
Condition 1	Sub_condition 1_1	res 1_1
	Sub_condition 1_2	res 1_2
Condition 2	… … …	…
Condition n		res n

图 3　水平条件表格

在惠灵顿项目中，为了能对表格进行形式验证，OPG 开发了相应支持工具[25]。它是一个基于微软 Word 的宏程序。检查过程的步骤如下：1)将表格存储为富文本格式(RTF)；2)使用设计验证工具将 RTF 文件转换成标准的 PVS 输入文件；3)在 PVS 定理证明工具中进行证明，文档中的所有表格必须证明其不相交性和完备性。图 4 显示了需求文档和在验证过程中应用的工具。

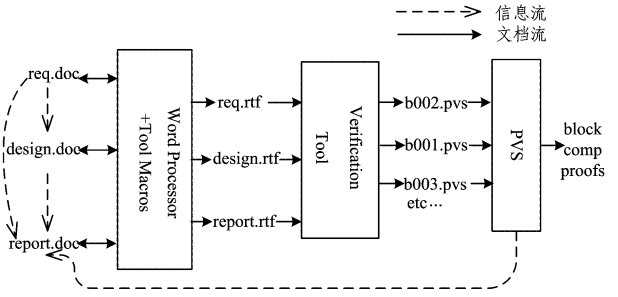

图 4　设计验证过程中的文档和工具

结束语　随着各类安全关键嵌入式系统中软件的规模和复杂性急剧增加，如何对复杂软件系统的需求进行描述和确认是一个具有挑战性的研究难题。研究表明，尽早发现需求阶段产生的错误能提高软件开发的质量并降低开发成本。前面详述了 3 种基于表格的需求规格说明描述方法，过去 30 年来的实践证明上述这些方法都已成功地应用于各类安全关键嵌入式控制软件系统的分析和设计过程中，例如飞机控制系统、核电站系统等。表格表达式的优点在于其可视化的结构

并能清晰地描述数学函数和关系，能对规格说明进行一致性和完备性检查，采用表格语言可以提高规格说明的可理解性和可读性。

然而，表格方法在实际的应用过程中仍然存在诸多需要进一步研究的内容。目前存在着多种表格方法，每种方法均定义了不同类型的表格。然而在实际的应用过程中，究竟应该使用哪些表格，对于某类问题采用何种表格进行描述最为合适？目前为止，表格方法还缺乏统一的语法和语义，这将给表格方法潜在的使用者带来学习障碍。在表格的形式语义方面，David L. Parnas 教授与金英博士定义的表格表达式的最新形式模型完全建立在标准数学概念的基础上，成功地脱离了对具体可视的表格的依赖，具有良好的可扩展性[15]，可以作为未来表格形式语义的统一基础。目前支持表格方法的软件是有待进一步完善的实验室研究项目或者是不公开的专属软件，因此表格方法的进一步推广和应用，需要有功能更强大、更健壮的工具支持，基于开源的社区软件开发方法将是一个很好起点。

参 考 文 献

[1] Le Lann, Gérard. An analysis of the Ariane 5 flight 501 failure-a system engineering perspective[C]// Proceedings of International Conference and Workshop on Engineering of Computer-Based Systems. IEEE, 1997: 339-346

[2] Leveson N G, Turner C S. An investigation of the Therac-25 accidents[J]. Computer, 1993, 26(7): 18-41

[3] Research Z E. GM Recalls Cadillac on SW Glitch[OL]. http://www. zacks. com/stock/news/55570/gm-recalls-cadillac-on-sw-glitch, 2011-07-21

[4] Boehm B, Turner R. Balancing Agility and Discipline: A Guide for the Perplexed [M]. Addison-Wesley/Pearson Education, 2003

[5] Parnas D L. Precise Documentation: The Key to Better Software [C]// Nanz S, ed. The Future of Software Engineering. Berlin Heidelberg: Springer, 2011: 125-148

[6] Alspaugh T A, et al. Software Requirements for the A-7E Aircraft[R]. DTIC Document, 1992

[7] Parnas D L, Asmis G, Madey J. Assessment of safety-critical software in nuclear power plants[J]. Nuclear safety, 1991, 32(2): 189-198

[8] Parnas D L, Madey J. Functional documents for computer systems[J]. Science of Computer Programming, 1995, 25(1): 41-61

[9] Abraham R. Evaluating generalized tabular expressions in software documentation[D]. Dept. of Electronic and Computer Engineering, McMaster University, 1997

[10] Parnas D L. Tabular representation of relations[R]. Technical Report CRL 260[38]. McMaster University Canada, 1992

[11] Janicki R. Towards a formal semantics of Parnas tables[C]// Proceedings of the 17th international conference on Software engineering. ACM: Seattle, Washington, United States, 1995: 231-240

[12] Janicki R, Parnas D L, Zucker J. Tabular Representations in Relational Documents[M]// Brink C, Kahl W, Schmidt G, eds. Relational Methods in Computer Science. Springer Vienna, 1997: 184-196

[13] Janicki R, Wassyng A. Tabular expressions and their relational semantics[M]. Fundamenta Informaticae, 2005, 67(4): 343-370

[14] Kahl W. Compositional syntax and semantics of tables [R]. SQRL Report 15. McMaster University, 2003

[15] Jin Y, Parnas D L. Defining the meaning of tabular mathematical expressions[J]. Science of Computer Programming, 2010, 75(11): 980-1000

[16] 金芝，刘璘，金英. 软件需求工程原理和方法[M]. 北京：科学出版社，2008

[17] Parnas D, Peters D. An Easily Extensible Toolset for Tabular Mathematical Expressions Tools and Algorithms for the Construction and Analysis of Systems [M] // Cleaveland W, ed. Springer Berlin/Heidelberg. 1999: 345-359

[18] Peters, Dennis K, Lawford M. An IDE for software development using tabular expressions[C]// Proceedings of the 2007 conference of the center for advanced studies on collaborative research. IBM Corp., 2007: 248-251

[19] Eles, Colin, Lawford M. A tabular expression toolbox for matlab/Simulink[P]. NASA Formal Methods. Springer Berlin Heidelberg, 2011: 494-499

[20] Heitmeyer C. Formal methods for specifying, validating, and verifying requirements[J]. Journal of Universal Computer Science, 2007, 13(5): 607-618

[21] Heitmeyer, Constance, Archer M, et al. Tools for Constructing Requirements Specification: The SCR Toolset at the Age of Ten [R]. Naval Research Lab Washington DC Center for High Assurance Computing Systems (CHACS), 2005

[22] Heitmeyer C L, Jeffords R D, Labaw B G. Automated consistency checking of requirements specifications[J]. ACM Transactions on Software Engineering and Methodology (TOSEM), 1996, 5(3): 231-261

[23] Heitmeyer C L. Software Cost Reduction[M]. Encyclopedia of Software Engineering, John Wiley & Sons, Inc, 2002

[24] Moum G. Procedure for the Specification of Software Requirements for Safety Critical Software[R]. Report CE-1001-PROC Rev. 2. CANDU Computer systems Engineering Centre of Excellence Procedure, April 2000

[25] Wassyng A, Lawford M. Software tools for safety-critical software development[J]. International Journal on Software Tools for Technology Transfer, 2005, 8(4/5): 337-35