Обобщенная модель системы автоматов

И.Б. Бурдонов, А.С. Косачев

Институт системного программирования, Москва, Россия

igor@ispras.ru, kos@ispras.ru

Доклад посвящен моделированию, композиции и детерминизму составных систем. Компонент системы моделируется конечным автоматом с несколькими входами и выходами, а взаимодействие – синхронным обменом сообщениями по симплексным каналам связи, связывающими выходы со входами и называемыми *соединениями*. Переход автомата предъявляет требования к соообщениям на всех входах и выходах автомата и отдельно указывает часть входов и выходов, по которым сообщения принимаются или посылаются. Синхронность означает, что требования автоматов, связанных соединением, должны быть согласованы.

Задан конечный *алфавит сообщений* *M*, *∅∉M*, и *M∅ = M∪{∅}.* *Автомат* в алфавите *M* – это набор *A=(M,I,J,S,T,s0)*, где *I* – множество *входов*, *J* – множество *выходов*, *S* – множество *состояний*, *T*⊆*S×X×P×Y×Q×S* – множество *переходов*, где *X={x|x:I→M∅}* – множество *стимулов, Y={y|y:J→M∅}* – множество *реакций*, *P=2I* – для приёма сообщений, *Q=2J* – для посылки сообщений, *s0*∈*S* – *начальное состояние*, и выполнены условия: 1. *I*∩*J=*∅; 2. передаются непустые сообщения: *∀(s,x,p,y,q,t)∈T p⊆x‑1(M) & q⊆y-1(M)*; 3. автомат всюду определён по стимулам: *∀s∈S ∀x∈X ∃p,y,q,t (s,x,p,y,q,t)∈T*; 4. приём сообщений выполним независимо от посылки сообщений: *∀(s,x,p,y,q,t)∈T ∀q`⊆y-1(M) ∃t` (s,x,p,y,q`,t`)∈T*; 5. множества *I*, *J* и *S* конечны. Для *a=(s,x,p,y,q,t)* обозначим *sa=s*, *xa=x*, *pa=p*, *ya=y*, *qa=q*, *ta=t*. Для *A=(M,I,J,S,T,s0)* обозначим *IA=I*, *JA=J*, *SA=S*, *TA=T*, *s0A=s0*.

Пусть *V* – конечное множество автоматов в алфавите *M*. Входы и выходы автоматов разные: *∀A,B∈V (A≠B ⇒ IA∩IB=∅ & IA∩JB=∅ & JA∩JB=∅)*, состояние автомата – множество, и состояния разных автоматов не пересекаются: *∀A,B∈V ∀sA∈SA ∀sB∈SB (A≠B ⇒ sA∩sB=∅)*. *Система автоматов* – это набор ***R****=(M,V,E)*, где *E:EDom → EIm* – биекция, определяющая *соединения*, где *EDom⊆J****R***, *EIm⊆I****R***, *J****R****=∪{JA|A∈V}* – множество всех выходов всех автоматов, *I****R****=∪{IA|A∈V}* – множество всех входов всех автоматов. Для ***R****=(M,V,E)* обозначим: *V****R****=V*, *E****R****=E*. Вход *i∈I****R****\EIm* и выход *j∈J****R****\EDom* – *внешние*, связывающие систему с её окружением.

Вводится композиция в духе CCS [1], но с учётом соединений. Обозначим: для функции *f* и множества *N*: *f/N = f \ {(z,f(z)) | z∈N∩****Dom****(f)}*; для отношения эквивалентности *a~b = (a&b)∨(¬a&¬b)*; для множества переходов *T*: *States(T) = {sa|a∈T}∪{ta|a∈T}*. Пусть *A,B∈V*, *(j,i)∈E*, *j∈JA*, *i∈IB*.

Условие композиции переходов *f(a,j,i,b) = a*∈*TA & b*∈*TB & (A=B ⇒ a=b) & ya(j)=xb(i) & (j*∈*qa ~ i*∈*pb)*.

Композиция переходов *a[j,i]b = (sa∪sb, xa∪xb/{i}, pa∪pb\{i}, ya∪yb/{j}, qa∪qb\{j}, ta∪tb)*.

Композиция множеств переходов: *G[j,i]H* *=* *{a[j,i]b|a∈G & b∈H & f(a,j,i,b)}*.

Композиция автоматов: *A[j,i]B =* *(M,IA*∪*IB\{i},JA*∪*JB\{j},{s0A*∪*s0B}∪States(TA[j,i]TB),TA[j,i]TB,s0A*∪*s0B)*.

Композиция системы: ***R****[j,i] = (M,V\{A,B}∪{A[j,i]B},E\{(j,i)})*.

Композиция ассоциативна. Композиция ***R****^* системы ***R*** по последовательности всех её соединений не зависит от их порядка, не имеет соединений, и все её автоматы не связаны между собой соединениями, т.е. все их входы и выходы внешние. Для использования системы в качестве компонента другой системы, она докомпоновывается до одного автомата с помощью композиции без соединения (для несвязанных автоматов).

*a[]b = (sa∪sb,xa∪xb,pa∪pb,ya∪yb,qa∪qb,ta∪tb)*, *G[]H* *=* *{a[]b|a∈G & b∈H}*,

*A[]B = (M,IA*∪*IB,JA*∪*JB,{s0A*∪*s0B}∪States(TA[]TB),TA[]TB, s0A*∪*s0B)*, ***R****[A,B] = (M, V\{A,B}∪{A[]B}, E)*.

Автомат *A* *детерминирован*, если 1. состояние *s* и стимул *x* однозначно определяют приём сообщений *p* и реакцию *y* и 2. одинаково помеченные переходы из одного состояния совпадают. Если в системе ***R*** все автоматы детерминированные, разбиты на два класса: класс «вершин» и класс «дуг», каждое соединение связывает автоматы разных классов и в каждом автомате класса «дуг» реакция зависит только от состояния (не зависит от стимула), то композиционный автомат ***R****^* детерминирован.

Преложенную модель можно использовать для тестирования детерминированных составных систем. Если ошибки могут быть только в компонентах, а все соединения правильные, тестирование сводится к проверке правильности переходов каждого автомата. Однако автомат тестируется только как часть системы, что похоже на тестирование в контексте [2]. Предполагается, что известно, какими должны быть автоматы (с точностью до изоморфизма), и именно это проверяется. Тест наблюдает как состояния автоматов, так и передаваемые сообщения. Такие предположения оправданы, например, при имитационном тестировании аппаратуры (simulation-based verification) [16].

Литература

1. Milner R. Communication and Concurrency. Prentice-Hall, 1989.

2. Revised Working Draft on “Framework: Formal Methods in Conformance Testing”. *JTC1/SC21/WG1/Project 54/1, ISO Interim Meeting, ITU-T on*. Paris. 1995.

3. A. S. Kamkin, M. M. Chupilko. Survey of modern technologies of simulation-based verification of hardware. *Programming and Computer Software*, 2011, vol. 37 (3), pp. 147–152.