

ОБ ИСПОЛЬЗОВАНИИ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ ДЛЯ
ИЗУЧЕНИЯ ФИЗИКИ

В.М. Шпаков (Санкт-Петербург)

Введение.

Среди существующих разновидностей моделирования основным в настоящее время становится компьютерное моделирование, а в его составе — имитационное моделирование процессов и динамических систем. Наряду с использованием имитационного моделирования при разработке технических и производственных систем, оно также находит применение в науке и образовании. Рассмотрим требования, предъявляемые к методам и средствам моделирования с точки зрения его эффективного использования для изучения физики. При проведении моделирования можно выделить следующие этапы:

- разработка физической модели;
- разработка компьютерной программы реализации модели;
- валидация модели и проведение модельного эксперимента;
- интерпретация результатов моделирования.

Физическая модель системы разрабатывается с помощью, чаще всего, средств элементарной математики и математического анализа. Программа создается с помощью универсальных (Delphi, C или др.) или специализированных средств программирования (MATLAB, Scilab) [1]. Для уменьшения трудоемкости разработки на этом этапе желательно, чтобы язык программирования был ориентирован на язык спецификации физической модели, то есть — на язык элементарной математики и математического анализа. Также желательно, чтобы получаемая программная спецификация модели была прозрачна с точки зрения ее соответствия физической модели. Это позволяет облегчить проведение экспериментов и модификацию модели. При проведении экспериментов должны быть обеспечены приемлемые точность и быстродействие реализации процессов. Имеет также значение удобство управления процессом моделирования. Под этим понимается возможность остановки, и возобновления моделирования в произвольные моменты времени по командам оператора, остановки при возникновении специфицированных ситуаций. Важными являются удобства осуществления вариации параметров системы и модификации модели. Потребность в модификации модели возникает при проведении ее валидации, а также может возникать в ходе проведения экспериментов как вследствие желания повысить адекватность модели системе, так и вследствие желания модифицировать систему. При интерпретации результатов моделирования оценки параметров и координат системы должны получаться непосредственно из результатов без дополнительной их обработки.

Цель настоящего доклада состоит в том, чтобы на конкретном примере показать достоинства технологии реализации физических моделей, основанной на транзитивном логико-динамическом подходе к спецификации взаимодействующих гибридных процессов [2]. Этот подход кратко состоит в следующем. Для представления состояний непрерывных процессов задается множество вещественных переменных X и для представления состояний дискретно-событийных процессов и режимов гибридных процессов — множество логических переменных Q . Кроме того, создается множество предикатов от непрерывных состояний G . Таким образом, множество логических переменных состоит из двух подмножества $W = Q \cup G$. Для спецификации процессов необходимо определить

функции переходов следующих типов: $\sigma : W \rightarrow Q \times \{False, True\}$, $\delta : W \times X \rightarrow X$, $\gamma : X \rightarrow G \times \{False, True\}$. Функции перехода σ и δ задаются с помощью совокупности продукционных правил вида «Условие \rightarrow Действие» [3]. В качестве условий используются элементарные конъюнкции логических переменных, а в качестве действия — присвоение значений *True* определенным дискретным состояниям и режимам процессов. Такие конъюнкции интуитивно понятным образом интерпретируются как логико-динамические ситуации S . С учетом сказанного функция σ может быть представлена с помощью совокупности трансформационных правил следующего вида:

$$S_j \rightarrow q'_{j_1}, \dots, q'_{j_i}, \dots, q'_{j_m} \quad (1)$$

где S_j — текущая ситуация, $q'_{j_i} \in Q \times \{True\}$, $m = 1, \dots, |Q|$. Штрихи в данном случае обозначают сдвиг по времени между возникновением ситуации и изменением значений состояний процессов.

Для задания функции перехода δ необходимо определить алгоритмы вычисления транзитивных отношений между непрерывными состояниями для каждого режима. Это может быть сделано с помощью использования алгоритмов и процедур реализации элементарных динамических процессов, элементарных функций и арифметических операций. Основными элементарными процедурами являются интегрирующие, дифференцирующие, реализующие аperiodические и колебательные звенья, а также звенья чистого запаздывания. Транзитивное отношение τ определяется параметрами элементарного динамического звена и длительностью интервала времени Δt между текущим и следующим состояниями процесса. В случае вынужденного процесса оно также зависит от внешнего воздействия x . Для задания зависимости непрерывных состояний от логики развития гибридного процесса алгоритмы вычисления этих отношений включаются в исполнительные части правил, условными частями которых являются требуемые значения соответствующих режимов. В принятых обозначениях эти правила имеют вид:

$$S_j \rightarrow x'_k = \tau_k(x_k, x, \Delta t), \quad x_k, x \in X, \quad (2)$$

где S_j — ситуация, определяющая режим, τ_k — транзитивное отношение для состояния x_k . Сложные отношения формируются совокупностями правил (2), реализующими соединения элементарных процессов, прямые и обратные связи между ними.

Функция γ является логической по определению, ее реализация связана с вычислением неравенств следующего вида:

$$(x_{j_1} \geq (a_k + x_{j_2})) \wedge (x_{j_3} \leq (b_k + x_{j_4})) \rightarrow g_k, \quad (3)$$

где $x_{j_1}, \dots, x_{j_4} \in X$, $g_k \in G$, a_k и b_k — границы диапазона. Таким образом, совокупность взаимодействующих гибридных процессов может быть специфицирована путем формирования базы трансформационных правил (1, 2, 3).

Реализация процессов, специфицированных трансформационными правилами, производится интерпретатором правил. Исполняющая процедура интерпретатора в цикле сканирует базу правил и выполняет специфицированные действия. На каждом шаге цикла производится продвижение времени на определенную величину, определяющую масштаб моделирования по времени. В СПИИРАН разработан прототип среды моделирования,

реализующий описанный подход [4]. Среда помимо интерпретатора правил содержит редакторы векторов состояния процессов, редакторы правил, текстовые, графические и анимационные средства визуализации процессов. Рассматриваемое в следующем разделе моделирование динамической системы было выполнено с помощью данного прототипа среды.

Модель маятника на пружине.

Один конец пружины закреплен на шарнире, на ее другом конце находится груз. Разработаем физическую модель процессов, возникающих в этой системе при отклонении пружины от вертикали и изменении ее длины. При этом воспользуемся прямоугольной системой координат. Изменение координат центра масс груза происходит под действием веса P и силы пружины F_n . Вес груза не меняется. При изменении положения груза изменяется величина и направление силы пружины. На рис.1 представлены отношения между положением груза и действующими на него силами.

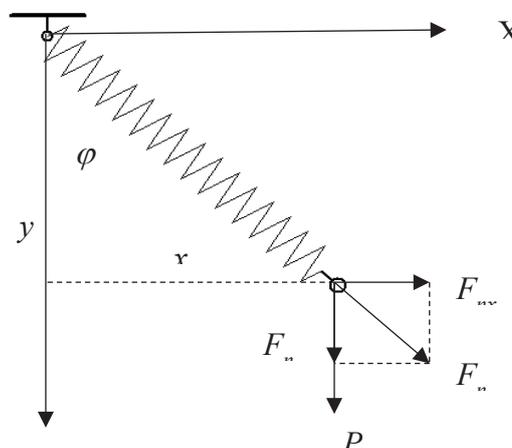


Рис. 1. Отношения между действующими силами.

Начальные данные: угол φ , масса груза m , вес P , жесткость пружины c , длина пружины в ненапряженном состоянии L_0 , реальная начальная длина пружины R , начальное значение силы пружины F_n . По начальным данным находят новые координаты груза на первом шаге цикла обработки правил. Для этого вычисляются составляющие по координатным осям действующей на груз силы. Затем последовательно вычисляются составляющие ускорения, скорости и координаты. По новым координатам вычисляются новые значения длины пружины и действующей силы, значение угла. Таким образом, процесс изменения положения груза может быть представлен последовательностью следующих вычислений: $\sin \varphi$; $\cos \varphi$; $F_{nx} = F_n * \sin \varphi$; $F_{ny} = F_n * \cos \varphi$; $F_x = F_{nx}$; $F_y = F_{ny} + P$; $A_x = F_x / m$ — ускорение по оси x ; $A_y = F_y / m$ — ускорение по оси y ; $V_x = \int A_x dt$ — скорость по оси x ; $V_y = \int A_y dt$ — скорость по оси y ; $x = \int V_x dt$ — новое значение координаты по оси x ; $y = \int V_y dt$ — новое значение координаты по оси y ; $R = \sqrt{x^2 + y^2}$ — новое значение длины пружины, $\Delta L = L_0 - R$ — деформация пружины; $F_n = -c * (L_0 - R)$ — новое значение силы пружины; $\operatorname{tg} \varphi = x / y$ — тангенс угла; $\varphi' = \operatorname{arctg}(\operatorname{tg} \varphi)$ — новое значение угла. Приведенную совокупность функциональных и интегральных отношений можно назвать физической моделью процессов. При реализации процессов последовательность этих вычислений

выполняется в цикле. Приведенная совокупность вычислений выполняется персональным компьютером за единицы или даже доли микросекунд. Если изменения состояний системы за такой промежуток времени можно считать малыми, то моделирование такой системы может производиться в реальном масштабе времени. В противном случае необходимо установить величину приращения времени на каждом шаге цикла вычислений, обеспечивающую требуемую точность реализации процессов. При этом процессы будут реализованы в модельном масштабе времени, определяемом этим приращением времени. Приведенные выше вычисления представлены на рис. 2 в виде совокупности трансформационных правил в формате редактора прототипа среды моделирования. Во 2-ой, 5-ой и 6-ой колонках находятся идентификаторы используемых переменных, в 3-ей — названия процедур. В качестве условий всех правил использована логическая константа $EverTrue = True$.

№	Переменная	Процедура	Коэффициент	Аргумент 1	Аргумент 2	Ситуация / cot
1	Sin(fi)	Синус	1.00	fi угол		Ever True
2	Cos(fi)	Косинус	1.00	fi угол		Ever True
3	Fn * sin(fi)	Умножение	1.00	Fn сила пружины	Sin(fi)	Ever True
4	Fn * cos(fi)	Умножение	1.00	Fn сила пружины	Cos(fi)	Ever True
5	F x	Пропорциональная	-1.00	Fn * sin(fi)		Ever True
6	F y	Сумма / разность	1.00	P вес	Fn * cos(fi)	Ever True
7	ускорение Ax	Деление	1.00	F x	m масса	Ever True
8	ускорение Ay	Деление	1.00	F y	m масса	Ever True
9	скорость V x	Интеграл	1.00	ускорение Ax		Ever True
10	скорость V y	Интеграл	1.00	ускорение Ay		Ever True
11	X	Интеграл	1.00	скорость V x		Ever True
12	Y	Интеграл	1.00	скорость V y		Ever True
13	X * X	Умножение	1.00	X	X	Ever True
14	Y * Y	Умножение	1.00	Y	Y	Ever True
15	R * R	Сумма / разность	1.00	X * X	Y * Y	Ever True
16	R длина текущая	Степень	1.00	R * R		Ever True
17	L0 - R деформация	Сумма / разность	1.00	L0 длина нулевая	R длина текущая	Ever True
18	Fn сила пружины	Умножение	-1.00	c жесткость	L0 - R деформация	Ever True
19	X / Y	Деление	-1.00	X	Y	Ever True
20	fi угол	Арктангенс	1.00	X / Y		Ever True

Рис. 2. Правила трансформации состояний модели маятника на пружине.

Нетрудно видеть, что имеется практически однозначное соответствие между физической моделью и программой ее реализации.

Валидацию модели можно провести следующим образом. Если в начальных условиях задать угол $\varphi = 0$, то этот маятник превращается в пружинный маятник, период колебаний

которого $T = 2 \cdot \pi \cdot \sqrt{m/c}$. В модельном эксперименте при $c = 10$ и $m = 2$ был получен период

колебаний пружинного маятника $T = 2.812$ сек. Расчетное значение $T = 2 \cdot \pi \cdot \sqrt{0,2} = 2,810$ сек. Если заменить пружину жестким невесомым стержнем ($c = 10000000$), то получим

математический маятник, период колебаний которого $T = 2 \cdot \pi \cdot \sqrt{L/g}$. При $L = 2$ и $g = 10$ в

эксперименте при начальном угле $\varphi = 5^\circ$ также были получены колебания с периодом 2,812 сек. Валидация модели также проводилась с помощью проверки выполнения закона сохранения энергии. Для этого в базу правил добавлялись правила для вычисления

кинетической энергии и изменения потенциальной энергии. Проверка проводилась для различных моментов времени. Расхождения в значениях имелись только в пятом знаке, т.е. не превышали 0,01%.

На рис. 3 приведен график процесса изменения угла при следующих значениях параметров: $m = 2$, $L_0 = 2$, $g = -10$ и начальных условий $\varphi = 30$, $R = 2$.

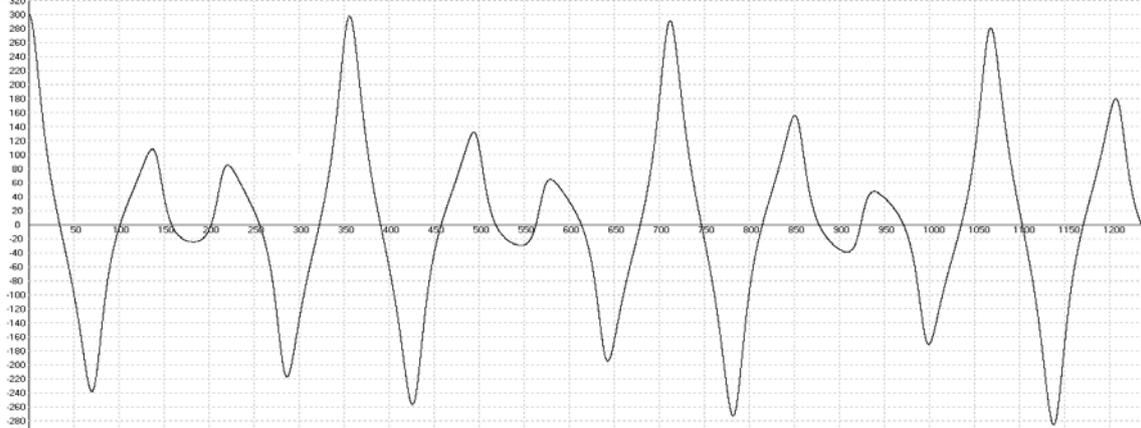


Рис.3. Процесс изменения угла маятника на пружине.

Процессы изменения угла φ и радиуса R в данной системе описываются системой двух обыкновенных нелинейных дифференциальных уравнений второго порядка [5]:

$$\frac{d^2 R}{dt^2} = R \cdot \left(\frac{d\varphi}{dt} \right)^2 - \frac{c}{m} (R - L) + g \cdot \cos \varphi, \quad \frac{d^2 \varphi}{dt^2} = -\frac{2}{R} \cdot \frac{dR}{dt} \cdot \frac{d\varphi}{dt} - \frac{g}{R} \cdot \sin \varphi.$$

Компьютерная реализация процессов, специфицированных этими уравнениями, была также выполнена с помощью трансформационных правил. Графики процессов, реализованных с помощью этих правил совпадают с графиками процессов, полученных на основании физической модели.

Модель маятника на пружине может быть легко преобразована в модель маятника на резинке. Резина отличается от пружины тем, что сила возникает только при ее растяжении. Для модификации модели достаточно изменить условную часть правила 18 (рис 2), а именно, заменить логическую константу *EverTrue* на предикат от длины резинки $R > L_0$. При этом, конечно, надо сформировать правило (3) для вычисления текущего значения этого предиката. Это правило в формате редактора прототипа среды моделирования имеет вид, представленный на рис. 4.

№	Предикаты		Переменные		Переменные	Константы	
1	$R > L_0$	IF	R длина текущая	>=	L0 длина нулевая	+	0.00 &

Рис. 4. Правило для спецификации предиката $R > L_0$.

Выводы

Технология компьютерной реализации взаимодействующих гибридных процессов, основанная на использовании трансформационных правил, обладает рядом достоинств, делающих ее применение для имитационного моделирования физических систем перспективным. Среди этих достоинств основными являются близость языка спецификации моделей языку физики, эффективность способов модификации моделей и

способов управления процессом моделирования, возможность эффективной программной реализации моделей.

Литература

1. Коткин Г.Л., Черкасский В.С., Компьютерное моделирование физических процессов с использованием MATLAB, учебное пособие, Новосибирск, НГУ, 2001. <http://www.exponenta.ru/educat/systemat/cherkassky/main.asp>.
2. Шпаков В.М. Исполняемые спецификации транзитивных моделей технологических процессов, Мехатроника, автоматизация, управление, № 3, 2004. С. 38-45.
3. Alur R., Henzinger T.A., Lafferriere G., Pappas G. J., 2000. Discrete Abstractions of Hybrid Systems, *Proceedings of the IEEE. No. 88*.
4. Шпаков В.М., Прототип среды моделирования структурированных совокупностей взаимодействующих процессов // Сборник докладов конференции “Имитационное моделирование. Теория и практика”, Санкт-Петербург, 19 – 21 октября 2005.-Т.II, сс. 292 – 295.
5. Старжинский В.М., Теоретическая механика. — М.: Наука, 1980.