

ГЛАВА 2. ИССЛЕДОВАНИЕ ВЗАИМОДЕЙСТВИЯ ЭС И СИМ

2.1. Методика интеграции систем ситуационного, имитационного и экспертного моделирования

В работе предлагается следующая методика интеграции систем ситуационного, имитационного и экспертного моделирования:

1. Выделение класса задач, которые могут решаться совместно системой имитационного моделирования, экспертной системой или ситуационным комплексом.

На первом этапе необходимо определить возможность применения трех рассматриваемых подходов для решения задачи в целом или отдельных подзадач, а также установить необходимую степень интеграции систем: для задач с небольшой размерностью может оказаться достаточным использование ЭС и СИМ, для плохо формализуемых задач большой размерности, с малым уровнем автоматизации, можно использовать только ситуационный подход.

Если не удастся предварительно определить степень интеграции, то необходимо выполнить последовательно этапы предлагаемой методики. Второй и третий этапы могут выполняться независимо. Если исключается использование уже разработанных ЭС и СИМ, то проектирование можно начинать с четвертого этапа.

2. Интеграция ЭС и СИМ.

Этот этап методики является актуальным для интеграции уже существующих ЭС и СИМ.

2.1. Разработка критериев для выбора наиболее подходящей системы или их комбинации.

При интеграции ЭС и СИМ необходимо определить какая из систем или их комбинация наиболее эффективно справляется с решением задач. В общем случае может существовать множество критериев. В работе предлагается лишь некоторые из них.

Для оценки эффективности ЭС и СИМ существует множество методик и алгоритмов, описанных в литературе. Далее в этой главе представлены алгоритмы и правила оценки временных показателей интегрированной экспертно-имитационной системы.

2.2. Выбор алгоритма взаимодействия.

При интеграции систем могут быть использованы различные алгоритмы взаимодействия, которые зависят от уровня автоматизации и -

выполняемых функций. Алгоритмы взаимодействия влияют на время решения задачи с помощью интегрированной системы.

2.3. Определение временных критериев.

На этом этапе осуществляется оценка временных затрат на решение задачи при интеграции ЭС и СИМ для различных алгоритмов взаимодействия.

2.4. Согласование процесса передачи и преобразования данных.

При интеграции существующих ЭС и СИМ необходимым является преобразование и согласование передаваемой информации, при этом временные затраты на преобразование данных оказывают сильное влияние на время решения всей задачи.

3. Использование ситуационного подхода.

Использование ситуационного подхода оправдано для систем большой размерности, где не требуются точные решения. Более подробно необходимые предпосылки рассматриваются в первой главе.

3.1. Создание семиотической модели предметной области.

Использование ситуационного подхода требует создания ситуационной модели предметной области, которую также называют семиотической, так как она включает описания на формализованных и ограниченно-естественных языках.

3.2. Разработка интерфейса (языка) для взаимодействия с ситуационной (семиотической) моделью.

Для работы с семиотической моделью необходимо разработать соответствующий язык взаимодействия, который позволит вносить изменения в ситуационную модель, осуществлять ее наполнение, извлекать необходимые данные.

4. Интеграция СИМ, ЭС и ССМ.

Данный этап необходим в тех случаях, когда интеграция ЭС и СИМ или реализация только ситуационного подхода являются неэффективными.

4.1. Создание интегрированной (SIE) модели предметной области.

Интеграция ЭС, СИМ и ситуационных систем требует разработки единой модели представления знаний. Ситуационно-имитационно-экспертная (SIE) модель должна включать возможность описания всех элементов ситуационного, имитационного и экспертного подхода.

4.2. Реализация взаимодействия на основе SIE-модели.

Создание единой модели для описания элементов недостаточно для реализации системы (программного комплекса), поэтому необходимо

разработать логику работу, переходы между уровнями, способы включения в систему существующих программных компонент.

Ниже представлена блок-схема предлагаемой методики. Более подробно этапы описаны в последующих главах первой части. Вторая часть посвящена практической реализации и результатам экспериментов, подтверждающих эффективность предлагаемой методики.



2.2. Интеграция ЭС и СИМ

Интеграция систем может быть двух типов: взаимодополняющая и взаимозаменяющая. В первом случае ЭС и СИМ выполняют различные (непересекающиеся) функции, которые необходимы для решения общей

задачи. Во втором случае они могут решать одни и те же задачи, и для них необходимо разработать соответствующие критерии выбора.

Взаимозаменяемые системы

В первой главе уже отмечалось, что существуют задачи, для которых возможно использование как ЭС, так и СИМ. Для определения какой из подходов более эффективный необходимо ввести критерии выбора.

Одним из главных критериев является время решения задачи.

$$KR_1 : \tau^{P3} = \min(\tau_S^{P3}, \tau_E^{P3})$$

где τ^{P3} – время решения задачи;

τ_S^{P3} – время решения задачи с помощью СИМ;

τ_E^{P3} – время решения задачи с помощью ЭС.

Сравним время выполнения различных операций в системе имитационного моделирования и экспертных системах. Пусть в ЭС существует постоянное (или среднее) время для осуществления одного вывода $t_{1в}$.

$$t_{1в} = const \mid t_{1в} = t_{1в.ср.} = \sum t_{1вi} / n,$$

где $t_{1в.ср.}$ – среднее время выполнения одного вывода;

n – количество выводов.

Время полного вывода для данной конкретной ситуации будет равняться произведению количества необходимых выводов на среднее время, затрачиваемое на один вывод.

$$t_{пол.выв.} = m * t_{1в.ср.}$$

где $t_{пол.выв.}$ – время полного вывода;

m – количество необходимых выводов.

Время имитационного моделирования зависит от количества блоков в системе, числа поступающих транзактов, времени задержки в блоках и количества необходимых имитаций.

$$t_{бл.} = const \mid t_{бл.} = t_{бл.ср.} = \sum t_{блi} / r,$$

где $t_{бл.}$ – время задержки в одном блоке;

$t_{бл.ср.}$ – среднее время задержки в одном блоке;

r – количество блоков.

$$t_{мод.} = k_{бл} * t_{бл.ср.} * Tr * I^l$$

где $t_{мод.}$ – время моделирования;

$k_{бл}$ – количество блоков;

¹ В формуле подразумевается, что одновременно обрабатывается только один транзакт.

Tr — количество транзактов;

I — количество имитаций.

Если время моделирования $t_{\text{мод}}$ меньше или равно времени полного вывода, то тогда СИМ эффективнее, чем ЭС, потому что ее результаты более точные и подробные. Но обычно это не так.

$$t_{\text{пол.выв}} \leq t_{\text{мод}}$$

Более того, время моделирования значительно больше, чем время вывода с помощью ЭС.

$$t_{\text{пол.выв}} \ll t_{\text{мод}}$$

Другим критерием может служить качественная оценка задачи. Под качеством можно понимать точность, достоверность, детальность, наглядность и др.

$$KR_2 : \omega^{ps} = \max(\omega_S^{ps}, \omega_E^{ps})$$

где ω^{ps} — качество решения задачи;

ω_S^{ps} — качество решения задачи с помощью СИМ;

ω_E^{ps} — качество решения задачи с помощью ЭС.

Критерий качества можно представить с помощью следующих оценок (характеристик).

$$\omega^{ps} = \{W, Ex, D \dots\}$$

где W — критерий важности;

Ex — критерий точности;

D — критерий детальности.

Сопоставим каждой оценке весовой коэффициент, тогда качественная оценка будет определяться следующей формулой:

$$\omega^{ps} = k_w * W + k_{ex} * Ex + k_d * D \dots$$

где k_w — весовой коэффициент важности;

k_{ex} — весовой коэффициент точности;

k_d — весовой коэффициент детальности.

Третьим критерием может служить простота в обращении и надежность, которые достигаются за счет использования более дешевой и простой в эксплуатации системы. Для выбора по этому критерию необходимо, чтобы система удовлетворяла следующим требованиям.

$$KR_3 : \forall (z_i \in Z) [(\tau_S^{ps} < \tau^{kp}) \& (\omega_S^{ps} > \omega^{kp})] \Rightarrow SIM$$

$$\forall (z_i \in Z) [(\tau_E^{ps} < \tau^{kp}) \& (\omega_E^{ps} > \omega^{kp})] \Rightarrow ES$$

где z_i – подзадача общей задачи Z ;

τ^{np} – максимальное допустимое время решения задачи;

ω^{np} – минимально допустимое качество решения задачи;

SIM – система имитационного моделирования;

ES – экспертная система.

Взаимодополняющие системы при взаимозаменяемости

Такого рода системы представляют собой интегрированные оболочки, где для решения задач используются различные подходы. Выбор такой конфигурации взаимодействия обусловлен невыполнением выше перечисленных критериев.

Если задачу можно выполнить с помощью ЭС и СИМ, то решение нужно разбить на следующие группы подзадач:

1. Решение подзадач с помощью ЭС;
2. Передача данных между ЭС и СИМ;
3. Решение подзадач с помощью СИМ.

$$\tau_{SE}^{p3} = \sum_{i=1}^n \tau_i^E + \sum_{j=1}^m \tau_j^S + \sum_{k=1}^l \tau_k^{Per}$$

где τ_i^E – время решения подзадачи с помощью ЭС;

τ_j^S – время решения подзадачи с помощью СИМ;

τ_k^{Per} – время передач данных между ЭС и СИМ;

n – количество подзадач, решаемых ЭС;

m – количество подзадач, решаемых СИМ;

l – количество передач данных между ЭС и СИМ.

Можно несколько изменить первый критерий:

$$KR_1' : \tau^{p3} = \min(\tau_S^{p3}, \tau_E^{p3}, \tau_{SE}^{p3})$$

где τ^{p3} – время решения задачи;

τ_S^{p3} – время решения задачи с помощью СИМ;

τ_E^{p3} – время решения задачи с помощью ЭС;

τ_{SE}^{p3} – время решения задачи с помощью ЭС и СИМ.

В основном время выполнения задачи с помощью ЭС существенно меньше, чем время работы СИМ. Чтобы минимизировать время необходимо максимально увеличить число задач для ЭС, уменьшить число передач данных и задач для СИМ.

$$\tau_{SE}^{p3} = \sum_{i=1}^n \tau_i^E + \sum_{j=1}^m \tau_j^S + \sum_{k=1}^l \tau_k^{Per} \approx n * \tau_{cp}^E + m * \tau_{cp}^S + l * \tau_{cp}^{Per}$$

$$\tau_{cp}^E < \tau_{cp}^S \Rightarrow \min(\tau_{SE}^{ps}) = \begin{cases} \min(m) \\ \max(n) \\ \min(l) \\ Cond \end{cases}$$

где $Cond$ – условия, связывающие m, n, l ;

τ_{cp}^E – среднее время решения подзадачи с помощью ЭС;

τ_{cp}^S – среднее время решения подзадачи с помощью СИМ;

τ_{cp}^{Per} – среднее время передачи данных между ЭС и СИМ;

n – количество подзадач, решаемых ЭС;

m – количество подзадач, решаемых СИМ;

l – количество передач данных между ЭС и СИМ.

Из-за необходимости передачи данных между ЭС и СИМ важно учитывать соответствующие задержки. Выбор системы для решения подзадачи обусловлен рассмотренными критериями. Добавим время передачи данных в соответствующие критерии.

$$\tau_{cp}^E + k^{Per} * \tau_{cp}^{Per} < \tau_{cp}^S$$

$$k^{Per} = \frac{l}{\min(n, m)}; \max(k^{Per}) = 2;$$

$$1 \leq l \leq (n + m - 1), l \in Var^l \Rightarrow k^{Per} \in \left[\frac{1}{\min(n, m)}, \frac{n + m - 1}{\min(n, m)} \right] \Rightarrow$$

$$\Rightarrow k^{Per} \in \left[\frac{1}{m}, 1 + \frac{n-1}{m} \right], \text{ при } n > m; k^{Per} \in \left[\frac{1}{n}, 1 + \frac{m-1}{n} \right], \text{ при } n < m$$

где k^{Per} – коэффициент передачи данных;

τ_{cp}^E – среднее время решения подзадачи с помощью ЭС;

τ_{cp}^S – среднее время решения подзадачи с помощью СИМ;

τ_k^{Per} – среднее время передачи данных между ЭС и СИМ;

n – количество подзадач, решаемых ЭС;

m – количество подзадач, решаемых СИМ;

l – количество передач данных между ЭС и СИМ.

Взаимодополняющие системы

При взаимодополняющей интеграции вопрос выбора одной из систем не возникает. Тем не менее, существует множество вариантов взаимодействия, которые оказывают влияние на передачу данных (τ^{Per}), и совместную (параллельную) работу систем (τ^{3d}). Особенности взаимодействия будут подробно рассмотрены далее в этой главе.

2.3. Варианты взаимодействия ЭС и СИМ

Взаимодействие СИМ и ЭС может быть различным в зависимости от поставленных задач, необходимости использования существующих программных продуктов, параметров технических средств и возможности автоматизации передачи информации. Можно выделить пять основных вариантов взаимодействия.

Первый вариант

Первый вариант взаимодействия двух систем является самым простым в реализации. Передача данных между СИМ и ЭС осуществляется полностью через пользователя (рис 2.1.а) или через нескольких пользователей (рис 2.1.б).

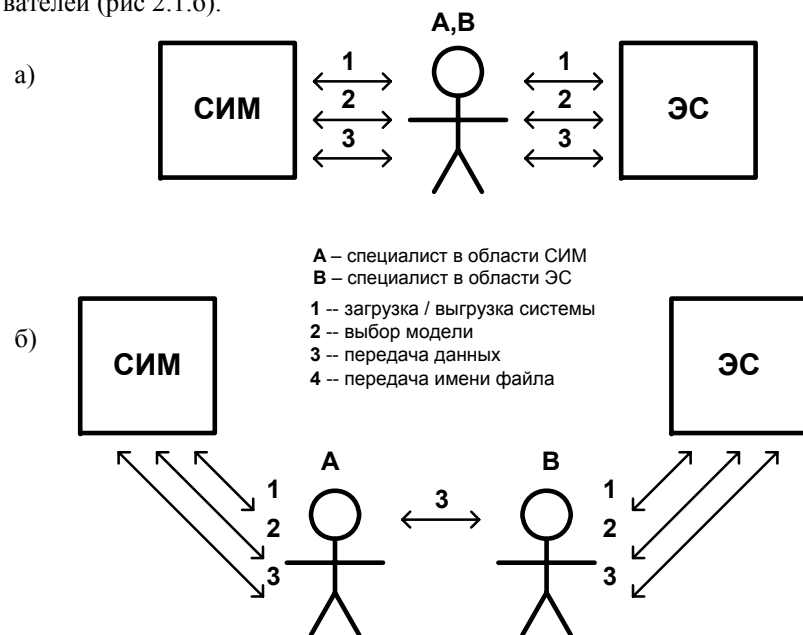


Рис. 2.1. Передача данных через одного или нескольких пользователей.

Одной из причин выбора такого варианта взаимодействия является случай, когда одна из систем является доминирующей во временном использовании, т.е. время использования одной системы значительно больше времени эксплуатации второй системы.

$$\tau^e \gg \tau^s \text{ или } \tau^s \gg \tau^e$$

где τ^e , τ^s — время использования ЭС и СИМ соответственно.

Второй причиной является сложность или ненадежность процесса автоматизации. Например, при работе с несколькими связанными ЭВМ, информация может не передаваться из-за ошибок и разрывов в каналах связи. В некоторых случаях информация требует обязательной интерпретации человеком (субъектом), поэтому в таких системах, а также при использовании ненадежных каналов связи указанный способ взаимодействия необходим в качестве основного или запасного варианта передачи данных.

В рамках этого варианта существует 4 возможные комбинации, которые отличаются различными временными затратами на передачу данных (τ^{Per}), и совместную (параллельную) работу систем (τ^{3o}).

$$\tau^{Per} = \tau_{Sim_Sub}^3 + \tau_{ES_Sub}^3 + \tau^{1-1} + \tau^{1-2} + \tau^2$$

где $\tau_{Sim_Sub}^3$ — время передачи данных м/у СИМ и пользователем;

$\tau_{ES_Sub}^3$ — время передачи данных м/у ЭС и пользователем;

τ^{1-1} — время загрузки/выгрузки системы;

τ^{1-2} — время переключения м/у системами;

$\tau^2 = const$ — время выбора модели (одинаково для 4 вариантов).

$$\tau^{3o} = \tau^{\parallel},$$

где τ^{\parallel} — задержка при параллельной работе двух систем.

Вариант 1(a). Этот вариант предполагает наличие двух ЭВМ и одного пользователя, который является специалистом в области ЭС и СИМ. Загрузка и выгрузка систем может производиться только один раз, поэтому затрачиваемое на это время может не учитываться. Задержки, вызываемые переключением между системами и параллельной (одновременной) работой отсутствуют, так как системы располагаются на различных ЭВМ.

$$\left. \begin{array}{l} \tau^{1-1} \rightarrow 0 \\ \tau^{1-2} \rightarrow 0 \end{array} \right\} \Rightarrow \tau_{1a}^1 \rightarrow 0$$

$$\tau^3 = \tau_{Sim_Sub}^3 + \tau_{ES_Sub}^3 = 2k_a * \tau_{int}^3;$$

$$\tau^{Per} = 2k_a * \tau_{int}^3 + \tau^2; \tau^{3d} = 0;$$

где $\tau_{1\sigma}^1$ – время загрузки / выгрузки или переключения системы;

τ^3 – время передачи данных м / у системами и пользователем;

k_g – коэффициент задержки для заданного варианта взаимодействия;

τ_{int}^3 – минимальное время передачи данных.

Вариант 1(б). Этот вариант предполагает наличие одной ЭВМ, одного пользователя и последовательную работу ЭС и СИМ. Загрузка и выгрузка систем должна производиться каждый раз при переключении. Время на передачу данных увеличивается по сравнению с предыдущим вариантом, так как пользователь должен запомнить (или записать) все данные, которые необходимо передать.

$$\left. \begin{array}{l} \tau^{1-1} \uparrow \\ \tau^{1-2} \rightarrow 0 \end{array} \right\} \Rightarrow \tau_{1\sigma}^1 = \tau^{1-1}$$

$$\tau^3 = \tau_{Sim_Sub}^3 + \tau_{ES_Sub}^3 = 2k_g * \tau_{int}^3; k_g > k_a$$

$$\tau^{Per} = 2k_g * \tau_{int}^3 + \tau^2 + \tau^{1-1}; \tau^{3d} \rightarrow 0;$$

где $\tau_{1\sigma}^1$ – время загрузки / выгрузки или переключения системы;

k_g – коэффициент задержки для заданного варианта взаимодействия.

Вариант 1(в). Этот вариант предполагает наличие одного ЭВМ, одного пользователя и параллельную работу ЭС и СИМ. Время на передачу данных уменьшается по сравнению с предыдущими двумя вариантами, так как пользователь может пользоваться буфером обмена. Временные задержки загрузки и выгрузки систем сменяются на задержки переключения, которые значительно меньше. Возможно многократное переключение, которое способствует увеличению эффективности передачи данных. При запуске одной системы из другой необходимо предварительно сохранить информацию о текущем состоянии вывода (имита-

$$\left. \begin{array}{l} \tau^{1-1} \rightarrow 0 \\ \tau^{1-2} \uparrow \end{array} \right\} \Rightarrow \tau_{1\sigma}^1 = \tau^{1-2}$$

$$\tau^3 = \tau_{Sim_Sub}^3 + \tau_{ES_Sub}^3 = 2k_g * \tau_{int}^3; k_g < k_a < k_g$$

$$\tau^{Per} = 2k_g * \tau_{int}^3 + \tau^2 + \tau^{1-2} + \tau^{1-2} + \tau^{save}; \tau^{3d} = \tau^{||};$$

ции).

где τ_{1a}^1 – время загрузки / выгрузки или переключения системы;

k_e – коэффициент задержки для заданного варианта взаимодействия;

τ^{save} – время, необходимое на сохранение.

Скорость выполнения операций зависит от объема памяти, производительности процессора, пропускной способности системных и периферийных шин, операционной системы и др. Время задержки на выполнение операций τ^{\parallel} различается в зависимости от их типа и количества, поэтому для оценки временных затрат необходимо вычислить τ^{\parallel} среднее для всех, типовых или наиболее частых операций.

$$\tau^{\parallel} = \overline{\tau^{\parallel}} = \left(\sum_{i=1}^n \tau_i^{\parallel} \right) / n,$$

где τ_i^{\parallel} – время выполнения операции

n – количество операций

Иногда следует выбирать другие критерии оценки, например, максимальное время выполнения операции:

$$\tau^{\parallel} = \max(\tau_1^{\parallel}, \tau_2^{\parallel}, \dots, \tau_i^{\parallel}, \dots, \tau_n^{\parallel})$$

Вариант 1(з). Этот вариант предполагает наличие двух ЭВМ и двух пользователей и аналогичен по затратам варианту 1(а), за исключением увеличения времени передачи данных между двумя пользователями.

$$\left. \begin{array}{l} \tau^{1-1} \rightarrow 0 \\ \tau^{1-2} \rightarrow 0 \end{array} \right\} \Rightarrow \tau_{1z}^1 \rightarrow 0$$

$$\tau^3 = \tau_{Sim_Sub}^3 + \tau_{ES_Sub}^3 + \tau_{Sub_Sub}^3 = 2k_a * \tau_{int}^3 + k_{sub_sub} * \tau_{int}^3 = k_e * \tau_{int}^3;$$

$$\tau^{Per} = k_e * \tau_{int}^3 + \tau^2; \quad k_e = 2k_a + k_{sub_sub} \rightarrow k_e; \quad \tau^{3a} = 0;$$

где τ_{1z}^1 – время загрузки / выгрузки или переключения.

Второй вариант

Второй вариант взаимодействия СИМ и ЭС подразумевает возможность запуска одной системы из другой или переключение между ними. Он имеет все преимущества первого варианта, но позволяет сделать “прозрачным” для пользователя переход между СИМ и ЭС (см. рис. 2.2.).

Третий вариант.

Третий вариант взаимодействия позволяет осуществлять передачу данных между ЭС и СИМ с использованием файлов. Одна из систем

сохраняет данные в виде файла. Другая система по команде пользователя считывает информацию о входных параметрах, производит обработку и получает выходные данные. Схема взаимодействия представлена на рис. 2.3.

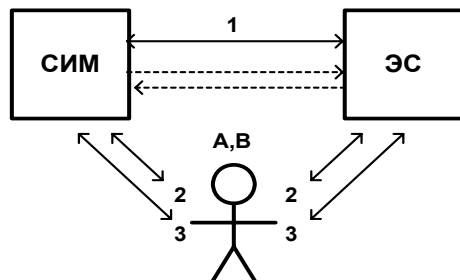


Рис. 2.2. Второй вариант взаимодействия.

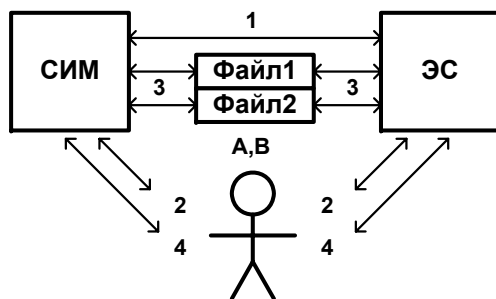


Рис. 2.3. Третий вариант взаимодействия.

Третий вариант автоматизирует передачу данных между СИМ и ЭС, а пользователь осуществляет управление файлами (сохранение, перенесение, открытие) и выбирает подходящую модель (логического вывода или имитационного моделирования).

Важным показателем такого взаимодействия является механизм доступа к разделяемым данным Sh , который определяет время и возможность доступа к данным. При работе ЭС и СИМ с пересекающимися данными задержка по определению прав доступа и разрешению конфликтов может сильно увеличить временные затраты τ^{sh} , а в некоторых случаях приводить к сбоям в работе компьютера.

$$\tau^{Per} = \tau_{Sim_HDD}^3 + \tau_{ES_HDD}^3 + \tau^{1-1} + \tau^{1-2} + \tau^2 + \tau^4;$$

где $\tau_{Sim_HDD}^3$ – время чтения/записи данных СИМ;

$\tau_{ES_HDD}^3$ – время чтения/записи данных ЭС;

τ^4 – время передачи имени файла;

$$\tau^{3d} = \tau^{||} + \tau^{sh},$$

где τ^{sh} – задержка при доступе к разделяемым данным.

Автоматизированная работа с файлами может быть осуществлена в четырех основных режимах:

а). Симплексный — информация передается только в одном направлении через один файл.

б). Полудуплексный — информация передается поочередно в двух направлениях через один файл.

в). Дуплексный — информация передается в двух направлениях через два различных файла.

г). Динамический — информация передается в двух направлениях через несколько файлов.

$$\tau_{simplex}^{sh} < \tau_{p_duplex}^{sh} < \tau_{duplex}^{sh} < \tau_{dinam}^{sh}$$

$$\tau^3 = \tau_{Sim_HDD}^3 + \tau_{ES_HDD}^3 < \tau_{Sim_Sub}^3 + \tau_{ES_Sub}^3$$

где $\tau_{simplex}^{sh}$ – τ^{sh} для симплексного режима;

$\tau_{p_duplex}^{sh}$ – τ^{sh} для полудуплексного режима;

τ_{duplex}^{sh} – τ^{sh} для дуплексного режима;

τ_{dinam}^{sh} – τ^{sh} для динамического режима.

Если используется вариант с последовательным запуском систем, т.е. с их выгрузкой и освобождением ресурсов. Тогда $\tau^{sh} = 0$.

При равномерном обращении к файлам время задержки дуплексного варианта приблизительно в два раза больше, чем при полудуплексном способе, но при этом значительно возрастает количество передаваемой информации.

$$2 * \tau_{p_duplex}^{sh} \cong \tau_{duplex}^{sh}; \tau_{p_duplex}^3 > \tau_{duplex}^3$$

где $\tau_{p_duplex}^3$ – τ^3 для полудуплексного режима;

τ_{duplex}^3 – τ^3 для дуплексного режима.

Четвертый вариант

Четвертый вариант взаимодействия ЭС и СИМ включает все возможности третьего. Помимо передачи данных он позволяет осуществлять автоматизированный выбор модели для СИМ или ЭС, которую обычно выбирает пользователь. Если установить соответствующие критерии выбора, то эту задачу можно автоматизировать (см. рис. 2.4).

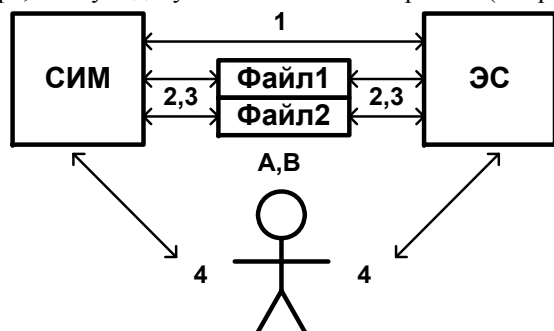


Рис. 2.4. Четвертый вариант взаимодействия.

Это важно при работе пользователя с большим количеством типовых моделей, при наличии у него недостаточных знаний или ограничений прав на принятие решения. Если время, необходимое для принятия решения τ^{kp} , меньше потраченного пользователем на выбор модели τ^2 , то следует автоматизировать этот процесс.

$$\tau_4^2 < \tau_1^2; \tau_1^2 \cong \tau_2^2 \cong \tau_3^2$$

где $\tau_1^2 - \tau^2$ для первого варианта взаимодействия;

$\tau_2^2 - \tau^2$ для второго варианта взаимодействия;

$\tau_3^2 - \tau^2$ для третьего варианта взаимодействия;

$\tau_4^2 - \tau^2$ для четвертого варианта взаимодействия.

Информация о модели передается вместе с исходными данными через файл и не должна рассматриваться как управляющая.

Пятый вариант

Пятый вариант взаимодействия ЭС и СИМ реализует передачу данных в реальном режиме времени без управления пользователем. Осуществляется автоматический выбор модели и передача данных. Информация передается не только перед запуском, но и в процессе работы про-

грамм. Динамический обмен данными может быть односторонним и двухсторонним. Схема взаимодействия представлена на рис. 2.5.

Такой вариант не только позволяет автоматизировать взаимодействие СИМ и ЭС, но сделать его полностью автоматическим. Снижаются требования к пользователю, так как ему необходимо работать только с одной из них.

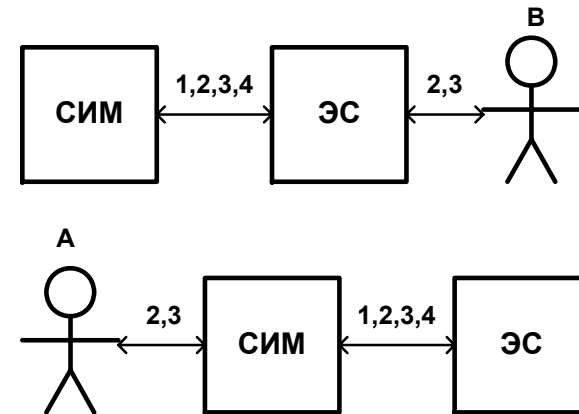


Рис. 2.5. Пятый вариант взаимодействия.

Такая передача информации может быть реализована через специальные технологии (DCOM, CORBA, ACTIVE-X и др.), которые обеспечивают управление программами извне и позволяют передавать данные в область памяти, используемой другой программой.

Автоматическое взаимодействие может быть использовано только для строго детерминированных задач, поэтому создаваемые системы должны включать другие варианты взаимодействия как альтернативные или запасные.

$$\tau^{Per} = \tau_{Sim_ES}^3 + \tau^{1-1} + \tau^{1-2} + \tau^2 + \tau^4$$

$$\left. \begin{array}{l} \tau^{1-1} \rightarrow 0 \\ \tau^{1-2} \rightarrow 0 \end{array} \right\} \Rightarrow \tau_{1e}^1 \rightarrow 0; \tau^4 \rightarrow 0;$$

$$\tau^3 = \tau_{Sim_ES}^3 = \min(\tau^3); \tau^2 = \min(\tau^2)$$

где $\tau_{Sim_ES}^3$ – время передачи данных м/у СИМ и ЭС;

$$\tau^{30} = \tau^{\parallel} + \tau^{sh}.$$

2.4. Алгоритмы взаимодействия СИМ И ЭС

В данном параграфе приведены алгоритмы взаимодействия СИМ и ЭС. Все множество вариантов взаимодействия можно объединить в три группы:

- последовательное взаимодействие;
- параллельное взаимодействие;
- последовательно-параллельное взаимодействие.

Последовательный алгоритм подразумевает поочередное использование каждой из двух систем: ввод исходных данных в СИМ (ЭС), получение выходных данных, передача их в ЭС (СИМ) в качестве исходных.

Если рассматривать систему как ориентированный граф (автомат), то запуск системы можно интерпретировать как вход, а последующие изменения как переходы между вершинами-состояниями (рис. 2.6).

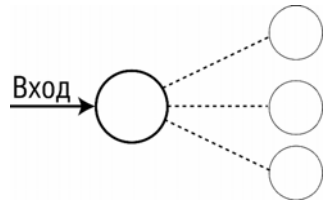


Рис. 2.6.

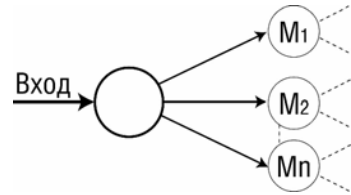


Рис. 2.7.

Можно сопоставить множеству существующих моделей множество графов (автоматов), совокупность которых представляет единый граф. Запуск программы для выбора модели обычно осуществляется один раз (рис. 2.7), но для удобства представления можно распараллелить вход (рис. 2.8). Далее объединим вершины состояния запуска системы и выбора модели в одну (рис. 2.9).

$$\{M1, M2, \dots, Mn\} \rightarrow Mi$$

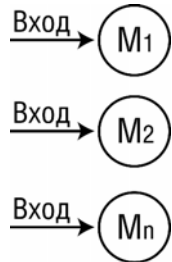


Рис. 2.8.

Следующим этапом является передача входных данных, который можно рассматривать как переход системы в нулевое состояние:

$$C_{тек} = C_0; \quad O1.P1 = A1; \quad \dots O1.Pn = An \quad \dots Ok.Pm = A1,$$

где A_i — внешние данные, т.е. данные, введенные пользователем;

O_i — объекты (переменные, устройства и др.) конкретной выбранной модели. Начальные значения параметров для них определяются пользователем;

P_i — параметры, свойства.

В случае, когда пользователь не может ввести исходные данные для некоторых параметров, необходимо присвоить им априорные или нулевые значения. Если предположить, что существует все множество исходных состояний, представленных вершинами графа, то ввод исходных данных можно рассматривать как выделение подграфа (рис. 2.10).

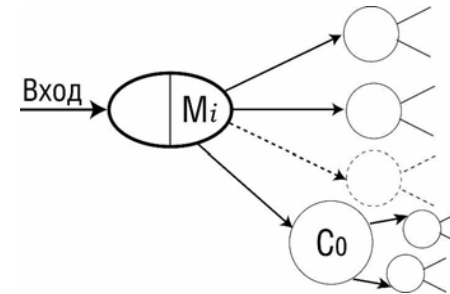
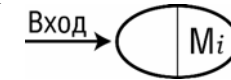


Рис. 2.10.

Рис. 2.9.



$O1.P2 = A2$



Рис. 2.11.

Предположим, что каждому входному состоянию C_0 соответствует единственное состояние C_s , тогда $C_0 \rightarrow C_s$ — переход из начального состояния модели в конечное. Переходы осуществляются с использованием функции перехода между объектами или состояниями под действием управляющей функции. После моделирования имеем следующие результаты.

$$C_s = \{O1.P1, \dots O1.Pn, \dots Ok.Pm\}$$

Конечное состояние моделирования характеризуется значениями всех объектов и переменных. Часть этих значений является выходными па-

раметрами $\{D\}_x$ или, другими словами, результатами моделирования, где x — количество исследуемых параметров.

Следующий шаг связан с вводом данных в ЭС и осуществлением логического вывода на основе данных, введенных пользователем, в том числе полученных в результате моделирования. Начальный этап запуска ЭС и выбора модели аналогичен этапу, связанному с СИМ.

Открытие и выбор модели логического вывода

$$\{M1, M2, \dots, Mn\} \rightarrow Mi$$

Ввод исходных данных пользователем

$$B1.P1=F(C1); B1.P2=F(C2); \dots B1.Pn=F(Cn) \dots Bk.Pm=F(Cl),$$

где Ci — внешние данные, т.е. данные, введенные пользователем;

Bi — гипотезы, определяющие вероятность факта или решения;

Pi — параметры, свойства гипотезы.

Для присвоения параметров гипотезы используются функции преобразования (программные процедуры Pr и/или математические формализмы H). Обычно программные процедуры включают в себя конкретные вопросы с вариантами ответов. Для каждого преобразования могут использоваться различные наборы H .

Множество входных данных $\{C\}_i$ для ЭС пересекается с множеством $\{D\}_x$ и характеризует переход к начальному состоянию (рис. 1.12).

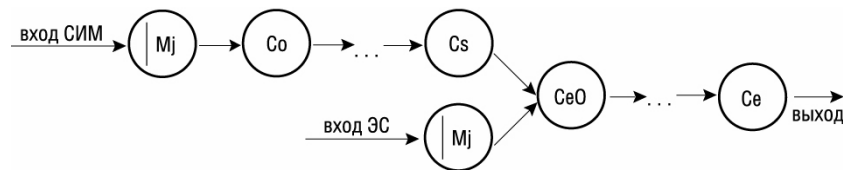


Рис. 2.12.

$Ce_0 \rightarrow Ce$ — переход из начального состояния модели в конечное. (Все переходы осуществляются с использованием правил логического вывода, изменения гипотез и применением программных процедур под действием управляющей функции). Получаем следующие результаты вывода:

$$Ce = \{B1.P1, \dots B1.Pn, \dots Bk.Pm\}$$

Конечное состояние вывода характеризуется значениями свойств исследуемых объектов, получением фактов и (или) наиболее вероятных

гипотез, что является решением S. Два варианта последовательного алгоритма взаимодействия представлены на рис. 2.13.

Последовательно-параллельное взаимодействие подразумевает параллельную работу ЭС и СИМ, но последовательное взаимодействие

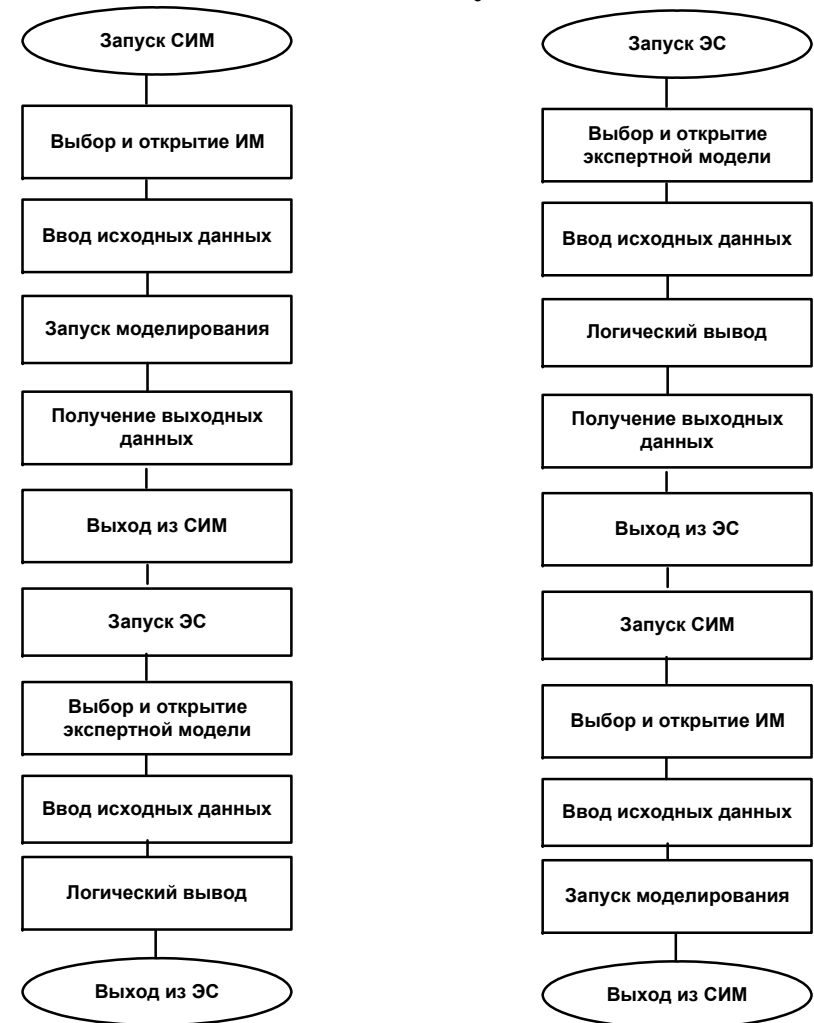
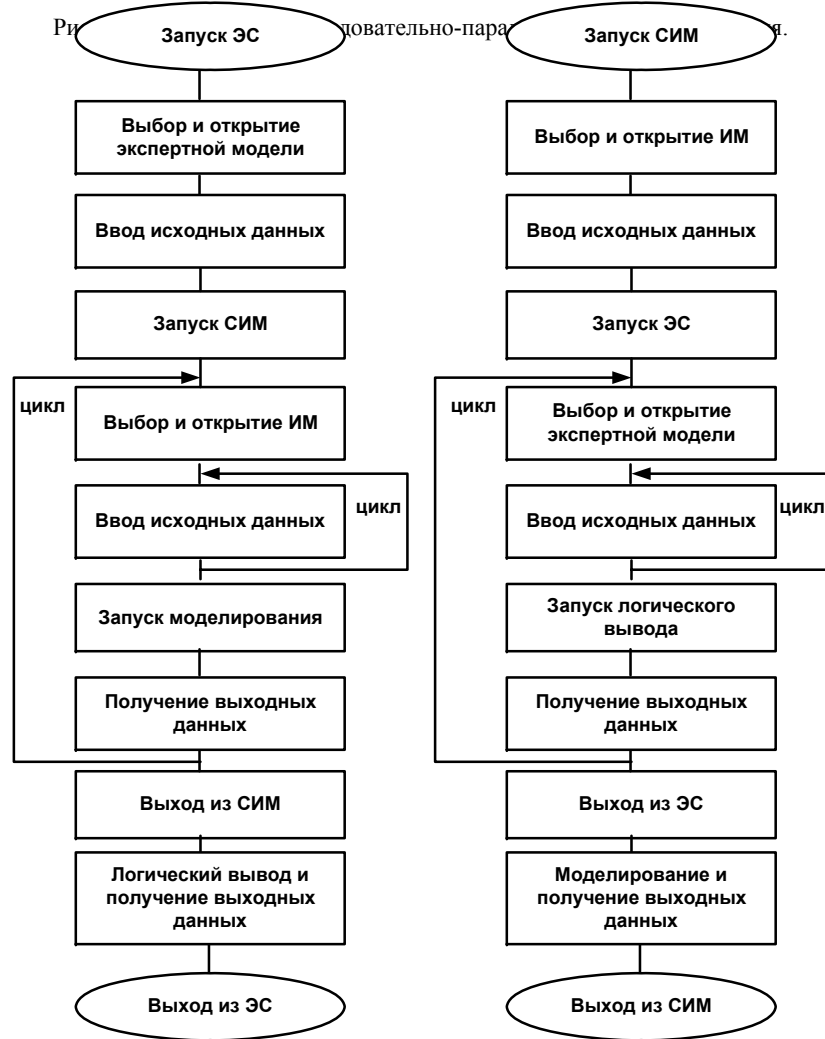


Рис. 2.13. Последовательные алгоритмы взаимодействия.

пользователем. Во время работы одной из них осуществляется запуск другой, не выходя из программы, но приостанавливая действия. Осуществляется моделирование результатов в СИМ (ЭС) и возврат в ЭС (СИМ). Алгоритм такого взаимодействия представлен на рис. 2.14.



При параллельном взаимодействии ЭС и СИМ работают вместе и при нехватке данных обращаются друг к другу, что определяет необходимость согласовывать передачу данных. Алгоритм приведен на рис. 2.15.

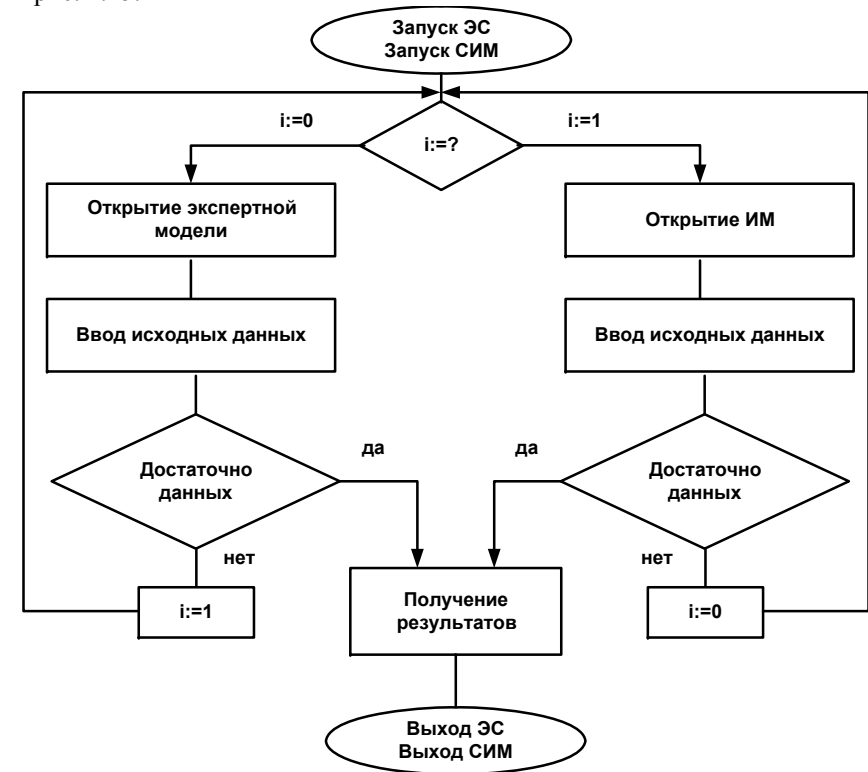


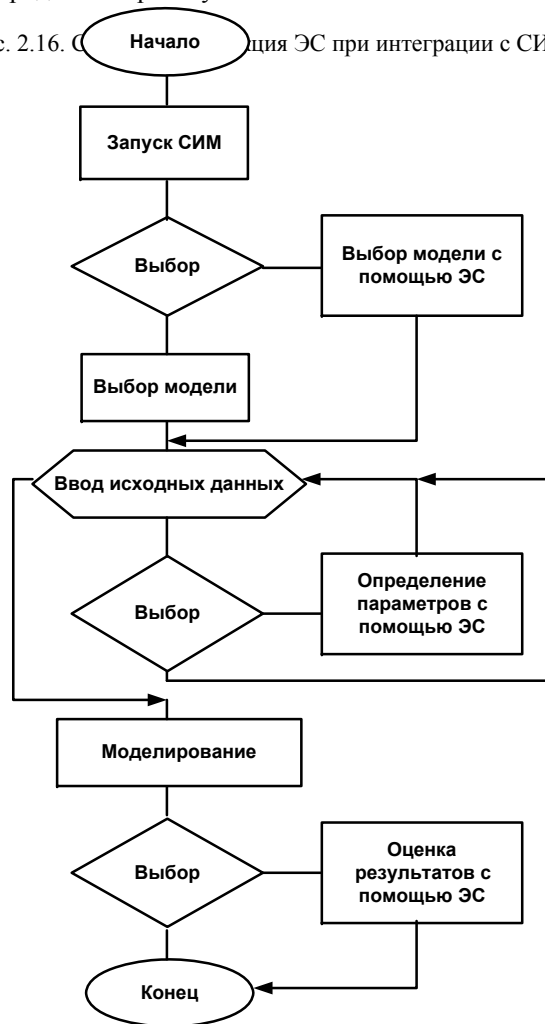
Рис. 2.15. Алгоритм параллельного взаимодействия.

При интеграции ЭС и СИМ компоненты могут выполнять различные функции. Наиболее широкое применение получила интеграция ЭС в СИМ в качестве справочной функции. При этом экспертная система помогает выбрать имитационную модель, подобрать и ввести исходные данные, а также оценить результаты, получаемые в результате имитации системы. Примеры систем, использующих такой алгоритм взаимодействия, приводятся в [Крапухина, 1995; Фураева, 1999]. Алгоритм в

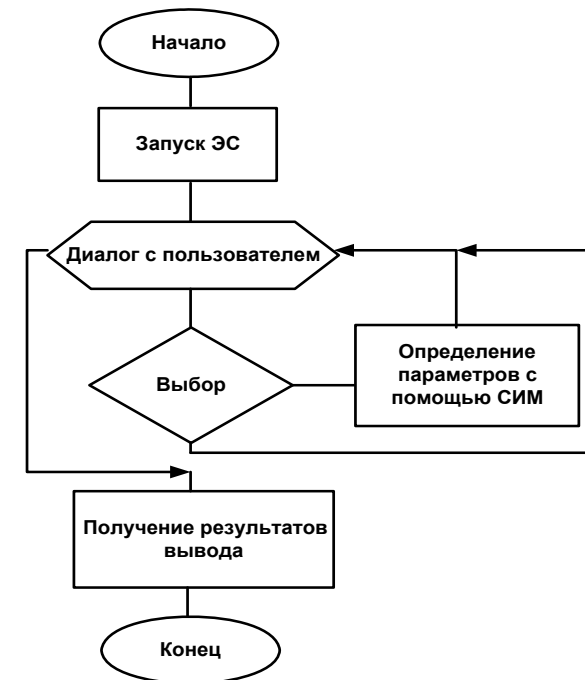
общем виде (при использовании ЭС во всех случаях) представлен на рис. 2.16.

В известной экспертной системе G2 есть модуль, осуществляющий имитационное моделирование наблюдаемого объекта в реальном времени. Если данные приходят с запозданием или отсутствуют в случае сбоя, то ЭС продолжает работу на основе показателей СИМ.

Рис. 2.16. Структура алгоритма функционирования ЭС при интеграции с СИМ.



При такой интеграции СИМ выполняет функцию процедуры извлечения знаний (данных) в ЭС. Алгоритм такого рода взаимодействия



приведен на рис. 2.17.

Рис. 2.17. Функция извлечения знаний СИМ при интеграции с ЭС.

При тесной интеграции ЭС и СИМ можно реализовать механизм свертки этапов процесса моделирования. При таком взаимодействии ЭС должна осуществлять наблюдение за процессом имитационного моделирования и накапливать статистику о типовых ситуациях и процессах. В случае повторного возникновения ЭС приостанавливает работу СИМ и осуществляет свертку, т.е. производит необходимые изменения параметров или структуры модели. Алгоритм такого взаимодействия приведен на рис. 2.18.

Надо отметить, что данный алгоритм является упрощенным, и для его реализации необходимо создать единую модель представления знания, или разработать соответствующие правила преобразования. Эти вопросы более подробно раскрываются в следующем параграфе и главе 4.

На сегодняшний день разработка эффективных алгоритмов свертки является одной из ключевых задач имитационного моделирования.

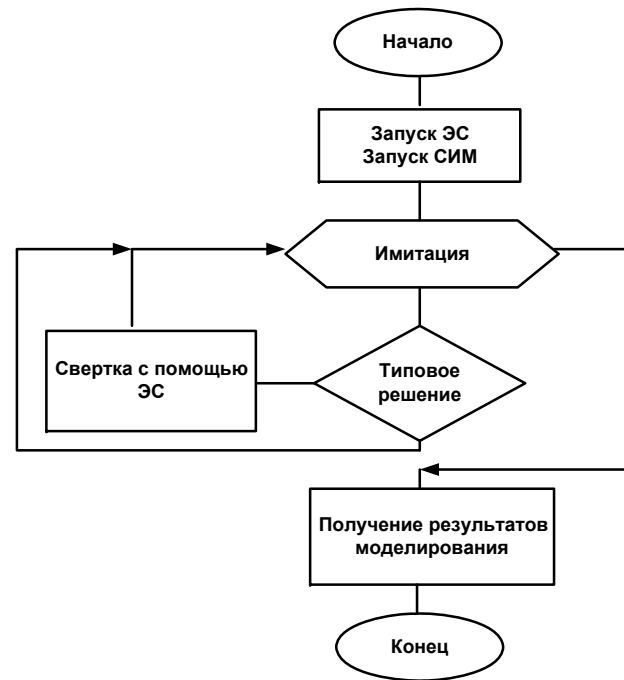


Рис. 2.18. Функция свертки ЭС при интеграции с СИМ.

2.5. Способы передачи видов данных из СИМ в ЭС

Основные результаты имитационного моделирования

Имитационное моделирование предназначено для получения характеристик системы (производительности, надежности), изучения проте-

кающих процессов, анализа статистических данных и выявления оптимальной конфигурации сложных систем.

Можно выделить четыре типа данных, которые получаются в результате имитационного моделирования: параметры объектов P , статистические данные SD , события E и состояния системы C . Приведенные типы данных являются составными, т.е. определяются несколькими значениями простейших характеристик объектов и значений переменных. Под простейшими характеристиками подразумеваются такие свойства объектов, которые не могут быть разделены на составные части. Возможность разделения характеристик на части определяется проектировщиком системы.

Различные типы данных могут иметь общие простейшие характеристики, но при этом рассматриваться независимо друг от друга. Приведем определения основных типов выходных данных, получаемых в результате имитационного моделирования.

P — параметры объектов или значения переменных, полученных в результате моделирования. Значения доступны только после окончания процесса моделирования. В основном, это временные характеристики выполнения операции или группы операций.

E — события, которые происходят при изменении параметров объектов или значений переменных. Общее количество событий за весь процесс моделирования вычисляется на основе количества параметров и переменных, принимаемых значений и числа дискретных интервалов времени моделирования (модельного времени). Обычно выделяются лишь некоторые события, которые контролируют группу параметров и осуществляются при достижении критических значений.

SD — статистические данные, которые отражают параметры объектов или значения переменных для различных интервалов времени. Обработка статистических данных может осуществляться во время моделирования. В этом случае могут использоваться такие параметры, как дисперсия, математическое ожидание, функция распределения.

C — состояние модели или системы. Состояние может определяться текущими значениями параметров, произошедшими событиями, временем моделирования, а также структурой связей и объектов модели.

Способы передачи информации о состоянии

Для передачи информации о состоянии предлагается использовать три способа: постоянная запись всех или выделенных характеристик

состояния в файл, запись в файл осуществляется только по внешней команде, запись в файл происходит после возникновения событий.

1-й способ. Запись информации о состоянии СИМ в файл² осуществляется с заданной частотой времени ν . ЭС производит чтение результатов, среди которых выбирает нужные моменты времени.

$$\nu = 1/T_{\text{диск}}$$

Минимальное время дискретизации $T_{\text{диск}}$ определяется исходя из суммы времени записи информации в память T_w , времени чтения T_r , времени доступа и разрешения конфликтов T_{sh} , времени перехода из одного состояния в другое (СИМ или ЭС) T_{on} .

$$T_{\text{диск}} = T_w + T_r + T_{sh} + T_{on}$$

$$\nu = 1/(T_w + T_r + T_{sh} + T_{on})$$

Время перехода из одного состояния в другое СИМ (T_{on}) в общем случае не постоянно. Оно изменяется в соответствии с протяжкой модельного времени, поэтому время дискретизации должно выбираться либо как минимальное из всех промежутков, либо максимальное, либо должно динамически изменяться и синхронизироваться в двух системах.

$$T_{on} = \max(T_{on_j}); T_{on} = \min(T_{on_j})$$

В общем случае состояние характеризуется всеми параметрами объектов и значениями переменных, поэтому данные могут передаваться полностью для каждого момента времени (или в каждый момент — для динамического взаимодействия).

$$M.T0; M.O.P1 = \dots, M.On.Pm = \dots$$

$$M.T1; M.O.P1 = \dots, M.On.Pm = \dots$$

...

$$M.Tn; M.O.P1 = \dots, M.On.Pm = \dots$$

где n — количество объектов;

m — количество свойств объекта;

$T0, T1 \dots Tn$ — периоды времени между началом и окончанием передачи информации о состоянии;

$$T1 = T0 + T_{\text{диск}}$$

Чаще всего необходима только часть всей информации о состоянии, поэтому можно записывать только выделенную часть характеристик

² В качестве файла может выступать база данных.

состояния. Чем меньше параметров, тем меньше может быть время дискретизации $T_{диск}$.

$$n1 < n; m1 < m \Rightarrow T_{диск1}(n1, m1) < T_{диск}(n, m)$$

2-й способ. Запись информации о состоянии происходит по внешнему сигналу, выдаваемому одной из систем (например, ЭС). Будем называть этот сигнал прерыванием. Для того, чтобы вызов прерывания был произведен вовремя, необходимо контролировать состояние системы. Лучшим вариантом является запись и чтение полной (необходимой) информации о состоянии. Время получение данных будет зависеть только от времени чтения Tr и времени доступа Tsh .

$$T_{зд} = Tr + Tsh$$

После операции чтения/записи система, от которой получена информация о состоянии, может быть приостановлена.

Можно также динамически изменять объем записываемой информации, т.е.

$$m = F1(t), n = F2(t); F1, F2 — \text{дискретные функции.}$$

Такой вариант, хоть и является более сложным, существенно увеличивает быстродействие при большом количестве характеристик n и m . Время непосредственной передачи полной информации о состоянии увеличивается, но уменьшается время обработки контрольных данных.

$$T_{зд} = Tr1 + Tsh1 + Tпер + Tsh2 + Tw + Tr2 + Tsh3$$

где $Tr1$ — время чтения контролируемых данных;

$Tпер$ — время обработки контролируемых данных и передачи запроса о записи полной информации о состоянии;

Tw — время записи полной информации о состоянии;

$Tr2$ — время чтения полных данных;

$Tsh1$ — время доступа перед операцией чтения контролируемых данных;

$Tsh2$ — время доступа после запроса о записи полной информации о состоянии;

$Tsh3$ — время доступа перед операцией чтения полной информации о состоянии.

Уменьшение временных задержек можно достичь при передаче в качестве исходных данных времени прерываний.

3 способ. Запись информации о состоянии S происходит при возникновении внутренних событий. В самой системе контролируются параметры, которые приводят к возникновению событий. Они могут быть выбраны из типовых внутренних событий и заданы в качестве исходных данных. Этот способ позволяет вызывать события не только по времени прерывания, но и по любому другому параметру.

Временные задержки будут зависеть только от времени чтения Tr и времени доступа Tsh .

$$Tзд = Tr + Tsh$$

Способы передачи информации о событии

Передача информации о событии зависит от того, в какой системе они возникают и чем вызваны. Предлагается три основных способа генерации событий:

1. Контроль состояния системы или группы параметров с помощью ЭС и вызов внешних событий для СИМ.
2. Контроль состояния системы или некоторых ее параметров внутри СИМ и вызов внутренних событий.
3. Промежуточным вариантом может считаться способ задания событий при формировании и передаче исходных данных между СИМ и ЭС.

Можно предположить динамическое изменение свойств событий, т.е. контролируемых критических значений параметров и связей. Изменение событий в реальном режиме времени может осуществляться как внутри системы (СИМ), так и по сигналам из внешней системы (ЭС).

Способы передачи информации о статистических данных

Для передачи информации о статистических данных предлагается четыре способа:

1. Внешний анализ данных, полученных на основе нескольких временных (или событийных) отметок. Под отметкой подразумевается момент времени Ti .

$$Ti = T0 + n * T_{диск},$$

где $T_{диск} = const$;

n — количество анализируемых состояний.

2. Внутренний анализ данных, полученных на основе нескольких временных (или событийных) отметок.

3. Накопление статистических данных на каждом этапе (состоянии) внутри системы. Например, такие параметры как дисперсия, математическое ожидание, закон и функция распределения.

4. Накопление статистических данных на каждом этапе (состоянии) во внешней системе.

Третий вариант предпочтительней четвертого, так как увеличивает быстродействие, но иногда невозможен из-за ограниченности одной из систем.

2.6. Преобразование форматов данных из СИМ в ЭС

Рассмотрим преобразование количественных данных на примере характеристик времени. Можно выделить два основных вида преобразования характеристик времени: простое (непосредственное) и количественно-качественное.

Простое или непосредственное преобразование подразумевает наличие в СИМ и ЭС непрерывного представления времени. Следствием этого является то, что каждому моменту времени в СИМ однозначно соответствует только один момент времени в ЭС. Это справедливо и в обратную сторону, т.е. фактически наблюдается однозначное соответствие между временем в ЭС и СИМ.

Количественно-качественное преобразование подразумевает наличие в ЭС дискретного или качественного представления времени, при этом в СИМ используется непрерывное представление. Дискретное представление означает, что используются не все моменты времени, а лишь некоторые, полученные путем округления, усечения или других способов. Обычно каждому дискретному значению времени ставится в соответствии интервал непрерывного времени. Качественное представление отличается от дискретного тем, что интервалу (интервалам) может соответствовать значение других величин, имеющих другую размерность или форму представления.

Простое преобразование

Если модельное время СИМ соответствует времени ЭС, то значение времени ЭС T_e равняется значению времени СИМ T_s .

$$T_e = T_s$$

Часто при моделировании используют протяжку модельного времени, т.е. ускоряют или замедляют его. Кроме того, время масштабирует-

ся в тех случаях, когда СИМ не позволяет использовать дробные значения времени, а переход к меньшей минимальной единице измерения невозможен (например, в GPSS). В этих случаях нужно ввести коэффициент соотношения K , равный произведению коэффициентов протяжки $K1$ и масштаба $K2$.

$$K = K1 * K2$$

Если коэффициент протяжки может изменяться, то необходимо определить его как функцию, зависящую от момента времени.

$$K1 = Func(Ts)$$

Неравномерная протяжка времени рассматривается более подробно при количественно-качественном преобразовании. Примем $K1$ постоянной величиной.

$$K1 = const \Rightarrow K = const$$

$$Te = K * Ts$$

$$\vec{K} = (K_{t_1}, K_{t_2}, \dots, K_{t_n}),$$

где t_i – временные параметры (характеристики);

n – количество параметров.

Для определения коэффициентов соотношения группе параметров можно воспользоваться векторной записью:

Область значений коэффициента соотношения можно разделить на следующие части:

$K = 1$ — точное совпадение, которое рассмотрено выше;

$K > 1$ — время в СИМ ускорено по сравнению с ЭС;

$0 < K < 1$ — время в СИМ замедлено по сравнению с ЭС;

$K < 0$ — отрицательное время.

Последнее значение может использоваться при суперпозиции задержек и ускорений. При этом K может быть постоянной величиной, но различные объекты СИМ могут изменять знак времени. Это связано с особенностями имитационных моделей, так как некоторые параллельные процессы могут быть представлены последовательно.

Ускорение и замедление модельного времени на отрицательной оси делится на те же промежутки, что и в случае с положительным временем.

$K = 0$ — нулевое значение может считаться ошибкой или характеристикой отсутствия данных.

Если в СИМ используется округление времени, то необходимо выполнять его после преобразования. В противном случае следует добавлять небольшое значение δ .

$$Te = K * Ts + \delta$$

Кроме ошибки округления часто возникает “ошибка деления на 0”, поэтому δ равняется сумме двух ошибок:

$$\delta = \delta 1 + \delta 2,$$

где $\delta 1$ — ошибка округления;

$\delta 2$ — “ошибка деления на 0”

$$\text{при } Ts = 0 \quad |K * Ts + \delta| = 0$$

$$Ts \neq 0 \quad |K * Ts + \delta| > 0$$

Если контроль ошибок не производится, то не рекомендуется использовать $K=0$ как показатель неизвестности данных.

Количественно-качественное преобразование

Качественное представление времени подразумевает наличие в ЭС некоторых дискретных значений, которые оно (время) может принимать:

$$Te = \{Te1, Te2, \dots, Ten\}$$

$n > 0, \quad n \in Z$ — количество дискретных значений. Может принимать только целые и положительные значения.

Интервальное представление определяется границами временного интервала:

$$Ts \Leftrightarrow To \leq Ts \leq Tk$$

To — начало интервала;

Tk — конец интервала.

Равномерное преобразование

Равномерное преобразование возможно в том случае, когда размеры интервалов одинаковы, т.е. каждое последующее дискретное значение (и интервал) может быть получено из предыдущего следующим образом:

$$Te_n = Te_{n-1} + T_{ш}$$

$T_{ш} = const$ — постоянное значение шага, определяющее размер интервала.

Если интервалы времени в двух системах совпадают, то значение шага можно вычислить следующим образом:

$$T_u = \frac{T_k - T_o}{n}$$

$Te1 = To$ & $Te1 = Tk$, где

$Te1, Ten$ — начало и конец интервала времени в ЭС;

To, Tk — начало и конец интервала времени в СИМ.

Соответствие интервала дискретному значению можно представить так:

$$T_L^{ES} = [T_u * (L-1); T_u * L] = \left[\frac{T_k - T_o}{n} (L-1); \frac{T_k - T_o}{n} L \right]$$

где T_L^{ES} — дискретное значение времени в ЭС;

L — дискретный номер значения, характеризующий номер интервала;

n — количество всех дискретных значений, т.е. общее количество интервалов.

В зависимости от способа задания границ интервала можно использовать следующие варианты:

$$T_L^{ES} = [T_u * (L); T_u * (L+1)] = \left[\frac{T_k - T_o}{n} * L; \frac{T_k - T_o}{n} * (L+1) \right]$$

$$T_L^{ES} = [T_u * (L - \frac{1}{2}); T_u * (L + \frac{1}{2})] = \left[\frac{T_k - T_o}{n} * (L - \frac{1}{2}); \frac{T_k - T_o}{n} * (L + \frac{1}{2}) \right]$$

Для унификации введем коэффициент Knp :

$$T_L^{ES} = [T_u * (L - k_{np}); T_u * (L + (1 - k_{np}))] =$$

$$= \left[\frac{T_k - T_o}{n} * (L - k_{np}); \frac{T_k - T_o}{n} * (L + (1 - k_{np})) \right]$$

$Knp = 0$ — при определении интервала в большую сторону;

$Knp = 1$ — при определении интервала в меньшую сторону;

$Knp = 0,5$ — при определении интервала в обе стороны;

В случае несовпадения границ можно вычислить интервал для любого дискретного значения двумя способами:

— округлением, когда все значения, выходящие за рамки одного из интервалов считаются максимальными (минимальными).

$$Te = F2(Ts)$$

$$F2(Ts) = Te1, \text{ при } Ts \leq To$$

$$F_2(T_s) = T_{en}, \text{ при } T_s \geq T_k$$

— усечением области программным способом, когда значения выходящие за рамки не рассматриваются вовсе.

$$T_{e1} = T_o \& (|) T_{en} = T_k$$

$$T_e = F_1(T_s), \text{ } F_1(T_s) \text{ not}(\exists) \text{ при } T_s > T_{en}, T_s < T_{e1}$$

Общий функционал для количественно-качественного преобразования выглядит следующим образом:

$$T_L^{ES} = k_{сж} T_s + \delta = k_{сж} \left[\frac{T_k - T_o}{n} * (L - k_{np}); \frac{T_k - T_o}{n} * (L + (1 - k_{np})) \right] + \delta =$$

Если границы не совпадают:

$$= k_{сж} \left[\frac{Func^3(T_k) - Func^3(T_o)}{n} * (L - k_{np}); \frac{Func^3(T_k) - Func^3(T_o)}{n} * (L + (1 - k_{np})) \right] + \delta$$

$$\text{где } Func^3 = func(choice) = \begin{cases} F_1(T_s), \text{ при } choice = 0; \\ F_2(T_s), \text{ при } choice = 1; \end{cases}$$

Неравномерное преобразование

Неравномерное преобразование характерно для случаев, когда размеры интервалов неодинаковы, т.е. интервал между последующими дискретными значениями изменяется. Например, при расчете времени выполнения заказа в полиграфии: сначала играют роль часы или даже минуты (выпуск газетных фотоформ). Если заказ нельзя реализовать в течение нескольких часов, то время уже рассчитывается днями. В случае создания многостраничной иллюстрированной энциклопедии время измеряется неделями или даже месяцами.

Формально, неравномерное преобразование можно задать изменяющимся значением шага или коэффициента сжатия. Значение шага может изменяться, в основном, из-за параметров экспертной системы. Так, например, в базе знаний могут храниться правила, оперирующие со значением тиража печатной продукции. Если объем выпуска небольшой, то учитывается каждая единица продукции. В случае многотысячных тиражей такой задачи не возникает.

На рис. 2.19 представлен график зависимости значения шага. Обычно функция имеет дискретный характер и постоянно возрастает. Если

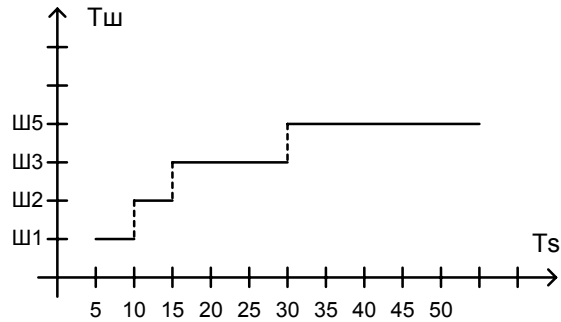


Рис. 2.19. Дискретная зависимость шага.

выражать эту зависимость в виде непрерывной функции, то она будет сначала пропорционально возрастать, а потом примет вид, близкий к логарифмической кривой.

В тех случаях, когда зависимость не только возрастает, но и убывает, то можно говорить о периодичности изменения. Если периоды постоянны, то высота пиков и расстояние между ними также будут постоянными (рис. 2.20).

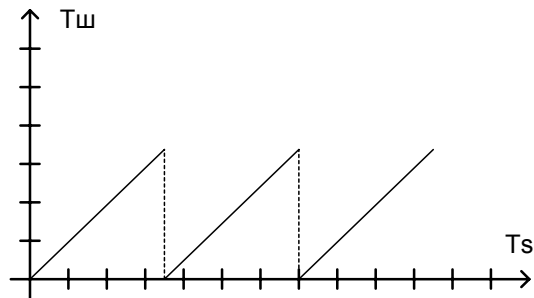


Рис. 2.20. Периодическое изменение шага.

Рассмотрим далее влияние изменения коэффициента сжатия на неравномерное преобразование, которое связано с варьированием модельного времени, точнее сказать с его шагом (размерностью). Если при движении транзакта встречаются сложные в обработке участки (например, параллельная работа с разрешением конфликтов на ресурсы) и простые (последовательная задержка транзакта в блоках), то сложность

вычисления с помощью ЭВМ сильно колеблется. Из этого следует, что в сложных случаях модельное время растягивается, а в простых – сжимается. На рис. 2.21. приведены примеры зависимости коэффициента сжатия 2:1, 1:1, 1:2, 1:3.

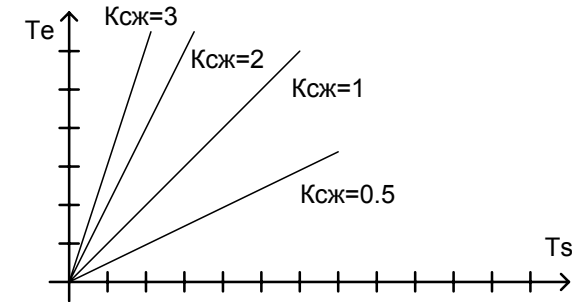


Рис. 2.21. Постоянный коэффициент сжатия.

На рис. 2.22. приведен график изменения коэффициента сжатия для различных этапов моделирования. На участках $[T_0, T_1]$ $K_{сж}=1$; $[T_1, T_2]$ $K_{сж}=0.5$; $[T_2, T_3]$ $K_{сж}=2$;

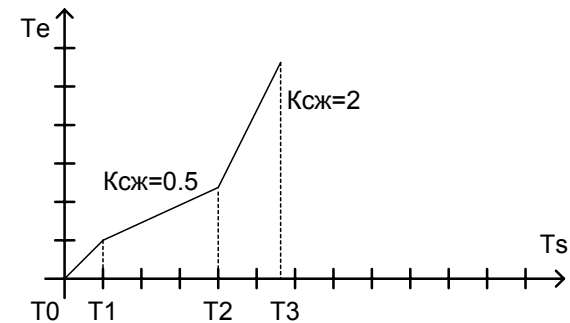


Рис. 2.22. Изменяющийся коэффициент сжатия.

Выводы

В этой главе представлена общая методика интеграции систем имитационного, ситуационного и экспертного моделирования. Ее этапы последовательно раскрываются во второй, третьей и четвертой главах.

В рамках первого этапа методики в главе исследована проблема выбора наиболее подходящей системы для реализации задач. Рассмотрены

случаи отдельной и интегрированной реализации систем. Разработаны критерии выбора для взаимодополняющей и взаимозаменяющей интеграции. При использовании методики необязательно пользоваться предлагаемыми критериями. Можно усовершенствовать или разработать новые критерии.

В рамках второго этапа методики были рассмотрены различные варианты взаимодействия ЭС и СИМ. Было выделено пять вариантов, зависящих от степени автоматизации передачи данных и управления. Самым удобным является вариант полной интеграции двух систем, однако, в случае использования ненадежных каналов передачи данных наименее автоматизированный вариант является предпочтительным. Оптимальным вариантом взаимодействия является возможность передавать информацию на различных уровнях автоматизации в зависимости от ситуации.

Для реализации различных вариантов были разработаны соответствующие алгоритмы: для последовательного, параллельного, последовательно-параллельного взаимодействия. Определены функции и алгоритмы, которые могут выполнять ЭС и СИМ при интеграции (справочная функция ЭС, функция извлечения знаний СИМ, функция свертки ЭС).

Для каждого варианта взаимодействия определены временные задержки, ограничивающие их применение. Рассмотрены случаи параллельной работы, определено время на разрешение конфликтов, загрузку, выгрузку, переключение и передачу данных.

Выделены основные результаты имитационного моделирования. Исследованы возможные способы передачи данных о *состоянии, событиях, статистических данных и элементарных параметрах*.

Исследованы математические преобразования, необходимые для передачи элементарных параметров между ЭС и СИМ. На примере характеристик времени выделены *простое, количественно-качественное, равномерное и неравномерное преобразования*.