

ПОИСК РЕШЕНИЯ НА ДВУХУРОВНЕВОЙ ОБЪЕКТНОЙ МОДЕЛИ СЛОЖНОЙ СИСТЕМЫ

М.П. Силич

Рассматриваются вопросы иерархического принятия решений в задачах проектирования сложных систем. Описана двухуровневая объектная модель, включающая модель материнской подсистемы и модели дочерних подсистем. В модели отражены межуровневые и внутриуровневые зависимости в виде функциональных отношений между атрибутами подсистем. Предлагается алгоритм выбора оптимальной комбинации вариантов подсистем, основанный на способе координации путем изменения локальных ограничений в частных моделях подсистем.

Введение

Иерархическое принятие решений – наиболее распространенный способ преодоления проблемы большой размерности модели в задачах проектирования сложных систем. Вместо выбора глобального решения на единой модели, описывающей всю систему в целом, осуществляется выбор оптимальных решений на моделях отдельных подсистем и согласование локальных выборов. Кроме сокращения размерности многоуровневая оптимизация имеет ряд других преимуществ. Во-первых, это возможность учитывать локальные критерии, отражающие собственные цели подсистем, что особенно актуально для социальных систем, т.к. исполнители решений могут иметь интересы, не совпадающие с целями системы. Во-вторых, такой подход позволяет использовать различные виды частных моделей подсистем (математические модели, модели в виде совокупности правил-продукций, модели экспертных оценок и т.д.) и соответственно различные методы поиска решений.

Многоуровневые задачи выбора рассматриваются в работах [1-6]. Большинство авторов сводят задачу многоуровневого выбора к совокупности задач выбора на двухуровневой модели, в которых согласовываются варианты подсистем, полученных в результате декомпозиции одной материнской подсистемы. Стратегия «прохождения» иерархии при этом может быть различной – «сверху – вниз», «снизу – вверх» или смешанной. Используются два основных способа координирующего воздействия на модели подсистем: влияние на локальные критерии и изменение локальных ограничений. В свою очередь, среди методов, использующих второй способ координации, можно выделить процедуры, направленные преимущественно на удовлетворение межуровневых ограничений и процедуры, предназначенные для учета внутриуровневых взаимодействий, т.е. для сохранения баланса входов-выходов подсистем одного уровня. Ко второй подгруппе, в

частности, относятся методы: «прогнозирование взаимодействий», «согласование взаимодействий», «оценка взаимодействий» [1] и др.

В данной статье рассматриваются вопросы поиска оптимальной комбинации вариантов подсистем, учитывающей внутриуровневые и межуровневые зависимости, в условиях ограниченных суммарных затрат на реализацию выбранных вариантов. Двухуровневая координационная модель формируется с использованием методики объектного моделирования, описанной в [6]. Для поиска решения на модели предлагается алгоритм, основанный на способе координации путем изменения локальных ограничений в моделях подсистем, а именно на методе распределения допустимого уровня затрат в сочетании с методом «распространения ограничений».

Двухуровневая объектная модель

Модель включает в себя n частных моделей дочерних подсистем s_i и одну модель материнской подсистемы s_0 .

Модель отдельной подсистемы формируется на основе некоторого выбранного класса $c = c(s_i)$, определяющего структуру описания подсистемы. Класс содержит имя, множество атрибутов $A(c)$ и множество методов $P(c)$. Каждому атрибуту $a_j \in A(c)$ сопоставлен домен $D(a_j)$ – множество его возможных значений. На множестве атрибутов класса могут быть установлены отношения зависимости (dependence)

$R^d = (a_1, \dots, a_n) \in A(c) \times K \times A(c)$, отражающие взаимовлияние атрибутов. На основании этих зависимостей формируются процедуры, определяющие значения одних атрибутов через значения других, и представленные, например, в виде множества правил-продукций, либо в виде формулы, либо в виде последо-

вательности операций. Эти процедуры включаются в состав методов класса $P(c)$.

Среди атрибутов выделяется подмножество, так называемых, базовых атрибутов $\{a_j\}^B \subset A(c)$, в которое включаются первичные характеристики. К их числу относятся: «конструктивные» параметры, определяющие принципы организации и функционирования подсистемы; «управляемые» параметры, на которые можно непосредственно воздействовать при управлении системой; «внешние возмущения», характеризующие условия функционирования. Будем выделять также подмножество целевых атрибутов $\{a_j\}^G \subset A(c)$, в которое включаются показатели эффективности подсистемы. Значения целевых атрибутов определяют целевое состояние. Они соответствуют заданной комбинации значений базовых атрибутов, являясь их следствием.

На базе класса может быть создано множество объектов $O(s_i) = \{o_k \mid c(o_k) = c(s_i)\}$, соответствующих различным вариантам реализации подсистемы. Каждый объект содержит конкретные значения атрибутов. В общем случае вместо конкретного значения может быть указана область значений.

бутов, характеризующие некоторую реализацию одной подсистемы, могут зависеть от значений атрибутов, характеризующих реализации других подсистем. Другими словами, между атрибутами классов, сопоставленных различным подсистемам, могут существовать отношения зависимости. Будем различать следующие типы межподсистемных отношений зависимости:

1. Межуровневые зависимости, связывающие атрибуты материнской подсистемы s_0 с атрибутами дочерних подсистем. Как правило, данный вид зависимостей представляется в виде функциональных отношений, разрешенных относительно атрибутов материнской подсистемы: $a_0 = f(a_1, K a_i, K)$, $a_0 \in A(c(s_0)), a_i \in A(c(s_i))$.

Примером является зависимость затрат материнской системы от затрат отдельных подсистем, определяемая как сумма.

2. Внутриуровневые зависимости, отражающие взаимное влияние атрибутов подсистем одного уровня (баланс входов-выходов подсистем). Пример такого рода зависимостей: объем продукции, производимой некоторой подсистемой, должен быть равен объему продукции, поступающей на вход другой подсистемы, умноженному на коэффициент потерь, возникающих при передаче.

На рис. 1 представлен пример двухуровневой модели, включающей межуровневые и внутриуровневые зависимости.

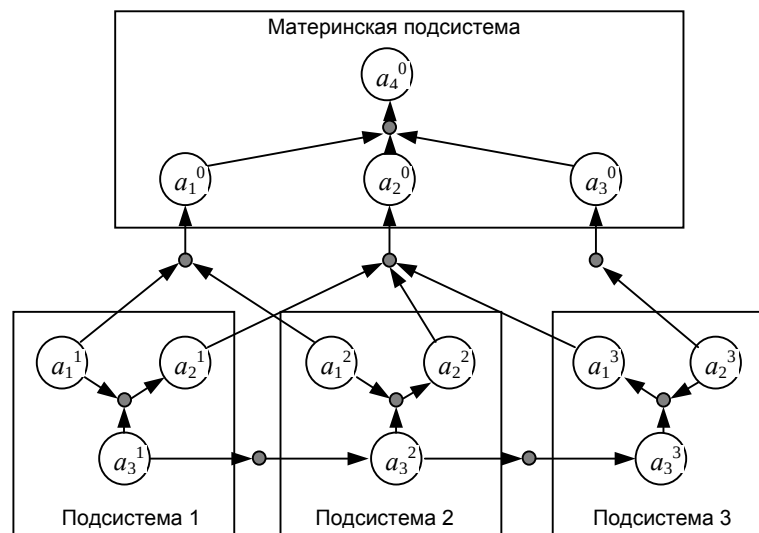


Рис. 1. Двухуровневая объектная модель

Как правило, сначала задаются значения базовых атрибутов, затем, исходя из заданных данных, определяются значения остальных атрибутов, в том числе с помощью методов класса $P(c)$.

Поскольку подсистемы взаимосвязаны друг с другом, очевидно, что значения атри-

Задача поиска оптимального решения

Рассмотрим задачу выбора такого сочетания вариантов дочерних подсистем, которое не выводит за рамки ограничений, наклад-

дываемых выбранным вариантом материнской подсистемы, не нарушает баланс входов-выходов дочерних подсистем и обеспечивает наилучшее значение глобального критерия эффективности. К ограничениям, накладываемым материнской подсистемой, относятся затраты на реализацию выбранного варианта, являющиеся аддитивной функцией затрат дочерних подсистем. В качестве глобального критерия эффективности, т.е. критерия для выбора комбинации вариантов дочерних подсистем выступает критерий эффективности материнской подсистемы.

Таким образом, модель материнской подсистемы должна содержать выбранный обобщенный вариант реализации $o(s_0)$, определяемый значениями (областями допустимых значений) атрибутов. Среди целевых атрибутов должен быть атрибут, выступающий критерием эффективности, а также атрибут, отражающий затраты материнской подсистемы, для которого должна быть определена допустимая область: $z \leq z^*$, где z^* - верхняя граница суммарных затрат.

Частные модели дочерних подсистем содержат целевые состояния в виде совокупности локальных критериев эффективности и локальных ограничений, а также функциональные отношения между атрибутами. Все критерии эффективности подсистем лучше выразить в одних единицах, например, в баллах, отражающих оценку качества функционирования. При этом в рамках подсистемы критерий может быть интегральным и определяться через значения некоторых частных критериев, например, с помощью функции свертки частных критериев. Ограничения формируются исходя из выбранного варианта материнской подсистемы и прогнозируемых внешних условий.

Для каждой пары дочерних подсистем s_1 и s_2 , имеющих взаимосвязи, устанавливаются функциональные зависимости между атрибутами этих подсистем:

$a_{j1}^1 R^d a_{j2}^2, a_{j1}^1 \in A(c(s_1)), a_{j2}^2 \in A(c(s_2))$. Соответствующие методы определения значений атрибутов включаются в состав классов.

Межуровневые зависимости представлены функцией затрат и функцией качества. Первая связывает атрибут материнской подсистемы z , отражающий глобальные затраты на реализацию комбинации вариантов, с атрибутами z_i , отражающими локальные за-

траты дочерних подсистем: $z = \sum_i^n z_i$.

Вторая связывает глобальный критерий эффективности K , в качестве которого выступает один из атрибутов материнской подсистемы, с локальными критериями K_i дочерних подсистем. Будем рассматривать случай, когда функция качества является монотонной, т.е. при увеличении значений локальных критериев увеличивается и значение глобального критерия. При этом вклад локальных критериев в улучшение глобального критерия может быть различным. Если все критерии являются количественными и выражены в одних единицах, то функция качества может быть выражена следующим образом:

$$K = \sum_i^n v_i \cdot K_i, \text{ где } v_i - \text{«вес» соответствующего}$$

локального критерия, $\sum_i^n v_i = 1$.

Необходимо найти сочетание локальных вариантов дочерних подсистем $\langle o(s_1), o(s_2), \dots, o(s_n) \rangle$, обеспечивающее наилучшее значение глобального критерия эффективности, сохраняющее баланс входов - выходов подсистем и удовлетворяющее ограничению, наложенному на суммарные затраты.

Задачу можно решить двумя способами. Первый предполагает перебор всех возможных комбинаций вариантов дочерних подсистем. Для каждой комбинации на основе известных значений атрибутов дочерних подсистем определяются значения атрибутов материнской подсистемы (интегрированных показателей) с использованием межуровневых зависимостей. Затем проверяется выполнение ограничений, накладываемых дочерними подсистемами друг на друга, и ограничений, накладываемых выбранным вариантом материнской подсистемы. Комбинация вариантов, удовлетворяющая всем ограничениям и обеспечивающая наилучшее значение глобального критерия, считается оптимальной. Основными недостатками данного метода являются не учет локальных критериев эффективности и большое количество формируемых комбинаций вариантов подсистем. Общее число комбинаций вычисляется по

формуле: $\prod_i^n m_i$, где $m_i (i = \overline{1, n})$ - количество вариантов i -той подсистемы.

Более предпочтительным методом решения задачи нахождения оптимального решения является координатный. Этот путь предполагает нахождение для каждой под-

системы оптимального в смысле локального критерия эффективности варианта. Но для того, чтобы в результате получить непротиворечивое и оптимальное в смысле глобального критерия сочетание вариантов, в модели дочерних подсистем вносятся определенные изменения, называемые координирующими воздействиями. Способы выработки этих воздействий определяются выбранными способами координации.

Алгоритм координации

Предлагаемый алгоритм нахождения оптимального решения на двухуровневой объектной модели сложной системы основан на способе координации через изменение локальных ограничений, т.е. через изменение множества допустимых решений. Ограничения задаются на основе межподсистемных зависимостей. Координируемыми параметрами при этом являются атрибуты, входящие в эти зависимости.

Поскольку межуровневые зависимости представлены аддитивной функцией затрат, предлагается использовать метод распределения, в соответствии с которым общий ресурс (затраты) распределяется между всеми дочерними подсистемами и таким образом формируются ограничения на локальные затраты. Полный перебор всех возможных координирующих воздействий предполагает перебор всех возможных разбиений множества значений затрат на n подмножеств, соответствующих каждой из дочерних подсистем. Очевидно, что допустимые уровни локальных затрат должны задаваться не случайным образом, а целенаправленно.

Для учета внутриуровневых зависимостей может быть использован метод «распространения ограничений» [5]. Суть этого метода состоит в том, что выбор оптимальных вариантов для каждой из дочерних подсистем осуществляется последовательно, причем для первой подсистемы решение находится без учета межподсистемных зависимостей, а для остальных – с учетом уже выбранных вариантов для предыдущих подсистем. В зависимости от того, в какой последовательности выбираются подсистемы при решении локальных задач, результат может быть различным. Полное количество возможных последовательностей определяется, как число перестановок и равно $n!$, где n – количество дочерних подсистем. Полный перебор координирующих воздействий в данном случае также оказывается неприемлемым.

В силу монотонности функции качества следует стремиться к нахождению оптимальных в смысле локальных критериев вариантов дочерних подсистем в условиях ограничений. Причем, большую «свободу» в выборе оптимального варианта следует предоставить подсистемам, которые дают больший вклад в критерий подуровня. Суть предлагаемого алгоритма состоит в том, что допустимый уровень затрат распределяется между дочерними подсистемами в соответствии с их «весом». Кроме того, при формировании последовательности перебора подсистем по методу «распространения ограничений» также учитывается важность подсистем.

Рассмотрим последовательность шагов алгоритма.

Шаг 1. Определение «веса» v_i каждой дочерней подсистемы, отражающего ее вклад в критерий подуровня. Для этого рекомендуется использовать метод парных сравнений. Сумма весов должна быть равна 1.

Шаг 2. Первоначальное распределение уровня допустимых затрат между подсистемами пропорционально их «весу». Для каждой дочерней подсистемы формируется ограничение $z_i \leq z_i^*$, где $z_i^* = v_i \cdot z^*$. Ограничение вносится в частную модель подсистемы.

Шаг 3. Формирование последовательности перебора подсистем Q . Последовательность формируется по «слоям». Первый «слой» составляет подсистема, имеющая наибольший «вес». Если имеется несколько подсистем с одинаковым максимальным «весом», то они располагаются в произвольном порядке. Для каждой из этих подсистем определяются подсистемы второго «слоя», связанные с ней межподсистемными зависимостями. Они располагаются следом за подсистемами первого слоя в порядке убывания «веса». Затем последовательно перебираются подсистемы второго «слоя» и для каждой из них определяются подсистемы, связанные с ней, но ранее не включенные в последовательность и т.д., пока все подсистемы не будут включены в последовательность. Если имеются подсистемы, не связанные с другими подсистемами, то они помещаются в конец последовательности в порядке убывания «веса».

Результатом является цепочка подсистем, определяющая порядок их выбора:

$$Q = \langle s_1, s_2, \dots, s_{n-1}, s_n \rangle.$$

Шаг 4. Выбирается очередная подсистема s_i из последовательности Q (i – номер в последовательности, $i = \overline{1, n}$). Формируются

ограничения от подсистем, связанных с текущей подсистемой функциональными зависимостями. При этом значения атрибутов уже рассмотренных подсистем, соответствующие выбранным вариантам этих подсистем, выступают ограничениями для текущей подсистемы. Атрибуты же еще не рассмотренных подсистем, рассматриваются как свободные переменные.

Определяется локальный оптимальный вариант $o(s_i)$ текущей подсистемы на частной модели подсистемы с учетом межподсистемных ограничений и ограничений на локальные затраты. Если нет допустимого решения, то можно вернуться на шаг 1 с тем, чтобы перераспределить заново «веса» подсистем и соответствующие уровни допустимых затрат.

Шаг 5. Перераспределение уровня допустимых затрат между подсистемами. Находится разница между уровнем допустимых затрат для текущей подсистемы и реальными затратами на выбранный оптимальный вариант: $\Delta z = z_i^* - z_i$. Если последовательность дочерних подсистем уже пройдена, то осуществляется переход на шаг 6. Если нет, то «остаток» финансовых ресурсов Δz перераспределяется между оставшимися подсистемами. Для каждой из подсистем $s_j, j = \overline{i+1, n}$, стоящих в последовательности Q следом за текущей подсистемой s_i , к ранее определенному уровню допустимых затрат z_j^* добавля-

ется остаток $\Delta z_j = \Delta z \cdot \frac{v_j}{v_{i+1} + K + v_n}$. Соответ-

ствующие ограничения на локальные затраты вносятся в частные модели подсистем. Затем – переход на шаг 4.

Шаг 6. Если уже перебраны все подсистемы из последовательности дочерних подсистем, то проверяется, имеется ли «остаток». Если остатка нет, то на этом работа алгоритма заканчивается.

Если $\Delta z > 0$, то принимается решение, либо закончить поиск, считая найденную комбинацию вариантов подсистем приемлемой, либо перераспределить остаток между подсистемами и пройти еще одну итерацию. В первом случае работа алгоритма заканчивается, во втором осуществляется перераспределение. Для последней подсистемы s_n уровень допустимых затрат устанавливается равным реальным затратам на выбранный оптимальный вариант $z_n^* = z_n$. Суммарная

величина затрат за вычетом затрат для n -той подсистемы распределяется между всеми остальными подсистемами $s_j (j = \overline{i, n-1})$ пропорционально их «весу» (без учета «веса» последней подсистемы):

$$z_j^* = (z^* - z_n^*) \cdot \frac{v_j}{v_1 + K + v_{n-1}}. \text{ Затем переходим}$$

на шаг 4.

Рассмотренный алгоритм не гарантирует нахождения глобального оптимума, однако он позволяет находить вариант, близкий к оптимальному, за приемлемое время в случае, когда решение задачи оптимизации методом полного перебора нецелесообразно. Данный алгоритм сочетает основные идеи «волнового» алгоритма координации и алгоритма «от безусловно оптимального» [6].

Алгоритм может быть использован не только для случая, когда межуровневые ограничения сводятся к ограничению на суммарные затраты, но и для любых ограничений аналогичного вида, когда задается допустимая область изменения некоторого атрибута материнской системы, являющегося аддитивной функцией соответствующих атрибутов дочерних подсистем. Например, когда ограничения накладываются на некоторые совместно используемые ресурсы.

СПИСОК ЛИТЕРАТУРЫ

1. Месарович М., Мако Д., Такахара И. Теория иерархических многоуровневых систем. – М.: Мир, 1973. – 344 с.
2. Михалевич В.С., Волкович В.Л. Вычислительные методы исследования и проектирования сложных систем. – М.: Наука, 1982. – 286 с.
3. Трахтенгерц Э.А. Методы генерации, оценки и согласования решений в распределенных системах поддержки принятия решений // Автоматика и телемеханика. – 1995. – №4. – С. 3-52.
4. Воловик М.А. Многокритериальный коллективный выбор сложных альтернатив // Вестник Красноярского государственного технического университета, посвященный 65-летию проф. Б.П.Соустина. – 1998. – С. 244-251.
5. Planning with constraints (MOLGEN: Part 1) // Artificial Intelligence. – 1981. - №16. – pp.11-139.
6. Силич М.П. Системная технология: объектно-ориентированный подход. – Томск: Том. гос. ун-т систем управления и радиоэлектроники, 2002. – 224 с.