

Синтез конфигурации изделия на этапах жизненного цикла¹

*К.И. Столяров,
асп.,
М.В. Овсянников,
доц., к.т.н., mvo50@mail.ru,
МГТУ им. П.Э. Баумана, г. Москва*

Автоматизация процесса синтеза конфигурации позволит сделать более эффективным данный процесс и позволит расширить задачу от простого формирования пригодной конфигурации до инструмента для быстрой и гибкой кастомизации изделия за счет применения более сложных закономерностей, правил и ограничений в форме онтологии предметной области. Особенность данного подхода заключается в оперировании, при решении спектра задач управления конфигурацией, широким многообразием реальных параметров объектов с применением методов построения онтологии проблемной области и программирования в ограничениях. Все виды ограничений задачи существуют в виде единого пространства аксиом и разрешаются (вычисляются) или проверяются решателем как единая система выражений.

В результате исследования были разработаны онтология предметной области синтеза конфигурации, алгоритмы и программная реализация макета CMSolver и проведено его тестирование.

Automating the process of configuration synthesis makes this process more efficient and extends the task from simple formation of suitable configuration to building a tool for quick and flexible customization of the product through the use of more complex laws, rules and restrictions in selected configuration. The peculiarity of this approach consists in operating, while performing the spectrum of configuration management tasks, a wide variety of real parameters of objects by using the methods of constructing domain ontology and constraint programming. All types of constraints for the problem above exist in the form of a unified space of axioms and are solved (calculated) or checked by the solver as a system of expressions.

The CMSolver that implements the proposed approach is intended to solve the problems of synthesizing product configurations.

Задача синтеза конфигурации сложного наукоемкого изделия решается на разных этапах производственного цикла от эскизного проектирования главным конструктором до комплектования изделия на этапе реализации менеджером.

Проблема учета конкретных требований при проектировании технических объектов приводит к необходимости решать задачу выбора структуры изделий или систем.

Нахождение оптимального состава компонентов затрудняется сложностью алгоритма расчета по проектным формулам и большой номенклатурой выбираемых компонентов. При проектировании необходимо рассчитывать разнородные параметры, такие как технические характеристики, параметры надежности, стоимость и технологичность. Дополнительную проблему создает необходимость отслеживать и учитывать при проектировании появление современных аналогов компонентов.

Требования к изделию можно представить в виде набора данных, содержащего всю информацию необходимую для выделения из пространства вариантов семейства модификации, удовлетворяющей заданным требованиям.

Задача управления конфигурациями и управления изменениями сложных высокотехнологичных изделий является частным случаем задачи структурного синтеза.

При большом числе комбинируемых компонентов или вариантов выбора и при значительной сложности вычисляемых и проверяемых зависимостей, ограничений и требований задача переходит в разряд трудно решаемых даже для многих существующих автоматизированных методов, не говоря уже о ручном способе поиска решений.

Основная проблема существующих подходов к автоматизации решения данного класса задач это необходимость в трудоемкой программной реализации алгоритмов поиска решений и поддержке разработанного программного средства для учета всех изменений в методике расчета.

В качестве цели задачи синтеза конфигурации принимается получение допустимой конфигурации в соответствии с установленными требованиями и приближающей характеристики конфигурации к оптимальным в соответствии с заданной целевой функцией.

В частности, такие задачи возникают при производстве изделий для требований конкретного заказчика, создания систем компьютерных, энергетических, при построении процедур обслуживания и др.

Например, при решении задачи проектирования районной электрической сети разрабатываются различные варианты схем электрических сетей и отбираются наиболее подходящие по техническим и экономическим требованиям. Выбирается электрическое оборудование (силовые трансформаторы, компенсирующие и распределительные устройства и т.д.) для осуществления надёжного электроснабжения потребителей в различных режимах загрузки.

При решении задачи проектирования подстанций осуществляется выбор главной схемы электрических соединений и выбор основного оборудования проектируемой подстанции (силовых трансформаторов, сборных шин и гибких токопроводов, кабелей, коммутационной аппаратуры, выключателей и разъединителей, измерительных приборов, изоляторов, распределительного устройства и т.д.). При проектировании выполняются расчеты различных технико-экономических параметров, среди которых расчеты мощности, токов короткого замыкания, приведённых затрат. В процессе определения целесообразности выбора того или иного варианта проводят проверку на энергетическую и экономическую эффективность задействованных технических решений.

¹ Работа (доклад) выполнена при финансовой поддержке РФФИ, грант № 18-07-01311

Применение для описания данных задачи онтологии свойств, позволяет применять методы программирования в ограничениях и разработать новые алгоритмы решения широкого класса инженерных задач в частности задачи управления конфигурацией.

Обычно выделяют 4 уровня: метаонтология, онтология верхнего уровня, онтология предметной области и прикладная онтология (рис. 1).

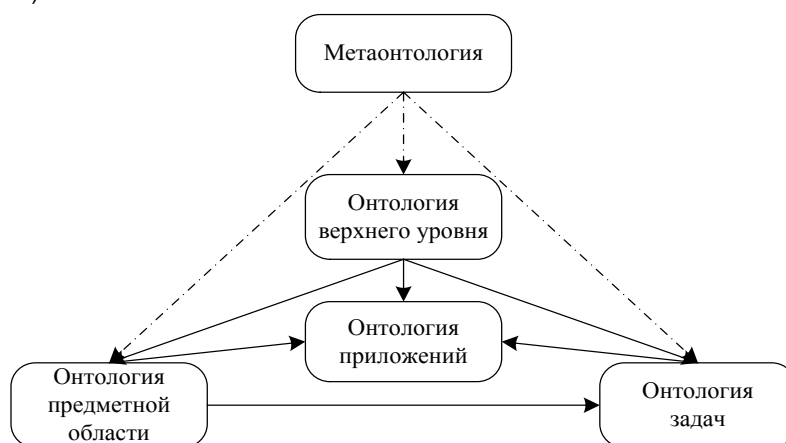


рис. 1. Структура иерархии онтологий

Метаонтология призвана обеспечить как точную, математическую спецификацию онтологий нижнего уровня, установление взаимосвязей между ними и формальный анализ их свойств.

Онтология представления в общем случае включает язык спецификации и язык описания функций интерпретации.

Ключевым моментом в проектировании онтологии является выбор соответствующего языка спецификации онтологий (Ontology specification language).

В настоящее время основными являются RDF и OWL, которые используются в SW и ISO 15926.

Для формальных онтологий в литературе рассмотрены следующие основные модели описания интерпретаций:

Формализованный естественный язык

Язык исчисления предикатов первого и второго порядка

Язык KIF - язык исчисления предикатов второго порядка для представления непроцедурных знаний, в котором процедурные знания представляются правилами переписывания термов.

Пользователь формальное описание своей задачи строит из фрагментов, хранящихся в библиотеке Ontolingua, как из блоков, состыковывая их различным образом и придавая некоторые значения некоторым переменным. В результате строится формальное описание задачи пользователя, которое с точки зрения математической логики также представляется в виде теории, то есть в виде набора переменных задачи, имен предикатов, имен функций и набора аксиом и уравнений задачи. Вопрос задачи может состоять в определении значений некоторых переменных или области истинности некоторой формулы или в проверке истинности некоторой формулы в построенной теории для задачи. [1]

Специальный язык описания математических выражений

Программные модули, реализующие соответствующие функции преобразования.

Пример - работы Евгенева, в которых разработан метод, позволяющий интегрировать продукционные базы знаний, выполняющие функции САЕ систем, с геометрическими знаниями, основанными на параметризованных 3D моделях САД систем. [2]

В данной работе используется метод описания интерпретаций, основанный на ограничениях **Constreine method**. Основной недостаток – невозможность удобной формальной записи интерпретаций (ограничений) в базе знаний не в программном коде.

Поэтому в Онтологии ЖЦП предполагается использовать в качестве языка описания интерпретаций модифицированный язык описания функций "EXPRESS", предложенный в стандарте ISO 10303 (STEP).

При использовании ПО для решения задачи синтеза конфигурации Бухановым С.А. был разработан специализированный алгоритм формальной трансляции ограничений.

Модель предметной области представляются в этом случае как неупорядоченная совокупность отношений, которые соответствуют связям, существующим между переменными задачи. Эти отношения, называемые общим термином "ограничения", могут иметь вид уравнений, неравенств, логических выражений и т.п.

Удовлетворение ограничений (Constraint satisfaction) является одной из ветвей программирования в ограничениях и имеет дело с проблемами, определенными на конечных доменах.

Решению задачи синтеза конфигурации методом CSP посвящен ряд работ [3, 4]. Предлагаемый метод решения имеет два основных отличия:

- Метод основан на онтологической модели проблемной области. Предлагаемый подход от классического метода CSP отличается применением его на структурированных онтологических структурах в условиях обобщенных ограничений.
- Метод реализует обобщенный алгоритм поиска решения. Особенность данного подхода заключается в оперировании, при решении спектра задач управления конфигурацией, широким многообразием реальных параметров объектов с применением методов программирования в ограничениях.

Описание природы каждого параметра является частью функциональной конфигурации. Каждый такой параметр отражает какое-либо свойство объекта и может содержать сложную систему значений, таких как набор дискретных значений, диапазоны, исключаящие диапазоны и логические выражения.

Ограничения содержатся в функциональной конфигурации и являются основой онтологии предметной области задачи. Все виды ограничений задачи существуют в виде единого пространства аксиом и разрешаются (вычисляются) или проверяются решателем как единая система выражений. Ограничения имеют форму логических или функциональных выражений, а также таблиц значений. Функциональные выражения имеют вид уравнения, приравненного нулю. Функциональные выражения могут быть использованы решателем для вычисления любого аргумента при остальных известных.

Традиционно существуют следующие виды конфигураций, соответствующие уровням композиционной абстракции изделия:

- функциональная конфигурация (ФК) – шаблоны, правила и принципы формирования проектной конфигурации (ПК);
- проектная конфигурация – определяет безальтернативную структуру изделия, готовую к реализации в виде физической конфигурации;
- физическая конфигурация - структура реального изделия, изготовленного согласно ПК.

В основе декомпозиции сложного изделия при формировании ФК лежит структурный подход. ФК представляет собой специальную систему справочников, которые содержат все необходимое для построения ПК и вычисления параметров компонентов всех уровней. Структуризация изделия внутри справочников происходит на основе функционально-конструктивной принадлежности узлов.

В процессе наполнения справочников происходит естественная функциональная классификация компонентов, определение параметров и, если это необходимо, связывание параметров текущего компонента с параметрами компонентов других классов с помощью аксиом. Механизм аксиом реализует систему разнородных типов ограничений структурирования: на основе прямых правил входимости компонентов, на основе валидации вычисляемых и агрегативных параметров.

Для такой организации данных о компонентах изделия характерна широкая потенциальная взаимозаменяемость компонентов внутри классов. Совокупность данных, составляющих ФК, обладает признаками самособирающейся структуры изделия. При хранении данных ФК применяется онтологический принцип описания структурных элементов, параметров, связей и аксиом.

Система хранения справочных данных предполагает использование метода программирования ограничений при использовании информации ФК и требований заказчика для создания актуальных вариантов ПК.

К параметрам, определенным в ПК, применяются выражения ограничений, описанных в ФК, и требования к значениям параметров. Разрешение ограничений происходит итерационным обходом пространства аксиом с вычислением промежуточных значений параметров, пока не будут найдены и проверены все параметры, перечисленные в требованиях, или не будет зафиксирован факт неполноты или невыполнимости ограничений для проверяемой конфигурации.

Таким образом, можно сформулировать следующие задачи:

1. Определить ПК, которая максимально удовлетворяет ограничениям ФК и требованиям заказчика.

Т.к. ФК задает состав компонентов и описывает ограничения на их свойства, задачу 1 можно свести к специальной задаче программирования в ограничениях.

a. Первый этап - генерация множества вариантов, которые отвечают заданным ограничениям, из сочетания компонентов, хранящихся в справочниках.

b. Второй этап – оптимизация - выбор варианта, по целевой функции.

2. Проверить степень соответствия ПК требованиям и ограничениям ФК.

Результаты проверки проявляются в виде системы оценок степени соответствия, проверяемой ПК отдельным видам требований и данных ФК, таких как: полнота ФК, полнота структуры ПК, выполнимость аксиом, соответствие значений параметров, запас по значениям параметров.

Функциональная конфигурация (ФК) полностью соответствует определению онтологии предметной области. Она представляет собой формализованное описание объектов определенного типа и знаний относительно их свойств.

Изделие состоит из компонентов. Каждый компонент соединен с одним или несколькими другими компонентами физически и может взаимодействовать (влиять на применимость) косвенно с еще несколькими компонентами.

У каждого компонента есть свой набор, который часто зависят (вычисляются) от свойств **входящих** в него суб-компонентов. Такие зависимости можно описать в виде функциональных зависимостей.

Каждое взаимодействие или связь между компонентами можно охарактеризовать через функциональное отношение между свойствами соответствующих компонентов.

Каждая связь может быть реализована в виде n -сторонней функциональной зависимости, т.е. в виде одной и той же зависимости, но выраженной через разные аргументы.

Функциональная конфигурация представляет собой набор данных, на основе которых формируется проектная конфигурация.

Эти данные должны включать в себя информацию, на основе которой можно построить корректную структуру изделия с учетом заданных требований.

ФК должна содержать данные:

- О функциональной декомпозиции изделия в виде иерархической структуры входимости компонентов, образующих конфигурацию.
- Каким-либо образом формально описать закономерности, определяющие возможность включения в конфигурацию конкретного компонента.

Механизм, позволяющий проверить собранную конфигурацию на соответствие требованиям.

Построение ФК является основой для дальнейшего многократного формирования множества проектных конфигураций. Описание ФК должно быть однозначным и непротиворечивым.

Узел верхнего уровня определяет структуру всего изделия и хранится в справочнике готовых конфигураций. Узлы нижних уровней конструктивно независимы, хранятся в классифицированном справочнике компонентов и могут быть использованы во множестве различных изделий.

Некоторые параметры узла верхнего уровня содержат исходные значения параметров, необходимые для синтеза конфигурации (например, обозначение модели автомобиля, конфигурация которой формируется в данный момент).

Эти параметры позволяют выбрать или проверить подходящие узлы из всей номенклатуры компонентов справочника, подходящие по классовому признаку для каждого конкретного ОК.

Множество СвОФК разделено на непересекающиеся подмножества, каждое из которых связано с определенным ОК или узлом. Каждое СвОФК описывает свойство соответствующего экземпляра компонента. Например, свойство «Максимальная скорость» объекта «Автомобиль».

Множество отношений структурирует, классифицирует и детализирует ОФК. Отношения могут применяться следующих типов: род – вид, часть – целое, объект – свойство, интерпретация.

Множество интерпретаций (АС) образует функциональные зависимости между свойствами множества СвОФК. Каждая зависимость может связывать свойства, принадлежащие различным ОФК. Множество интерпретаций образует систему уравнений, определяющих корректность сформированной конфигурации.

Интерпретации необходимы при синтезе конфигурации для доопределения значений некоторых свойств на этапе наполнения структуры, а также при проверке конфигурации на соответствие требованиям. Использование интерпретаций позволяет обуславливать применение одних компонентов в конфигурации значениями свойств других.

Интерпретации делятся на два вида:

Функции, позволяющие найти значения вычисляемых свойств (например, $(AM. Полная_масса = ДВГ. Масса + КП. Масса + ТБ. Масса)$).

Интерпретации (неравенства) проверки соответствия созданной конфигурации заданным требованиям. Данные интерпретации могут храниться в объектах крупных законченных компонентов (узлов) либо в объекте изделия.

Каждая такая интерпретация может быть реализована в виде n -сторонней функциональной зависимости, т.е. в виде одной и той же зависимости, но выраженной через разные аргументы (рис.2).

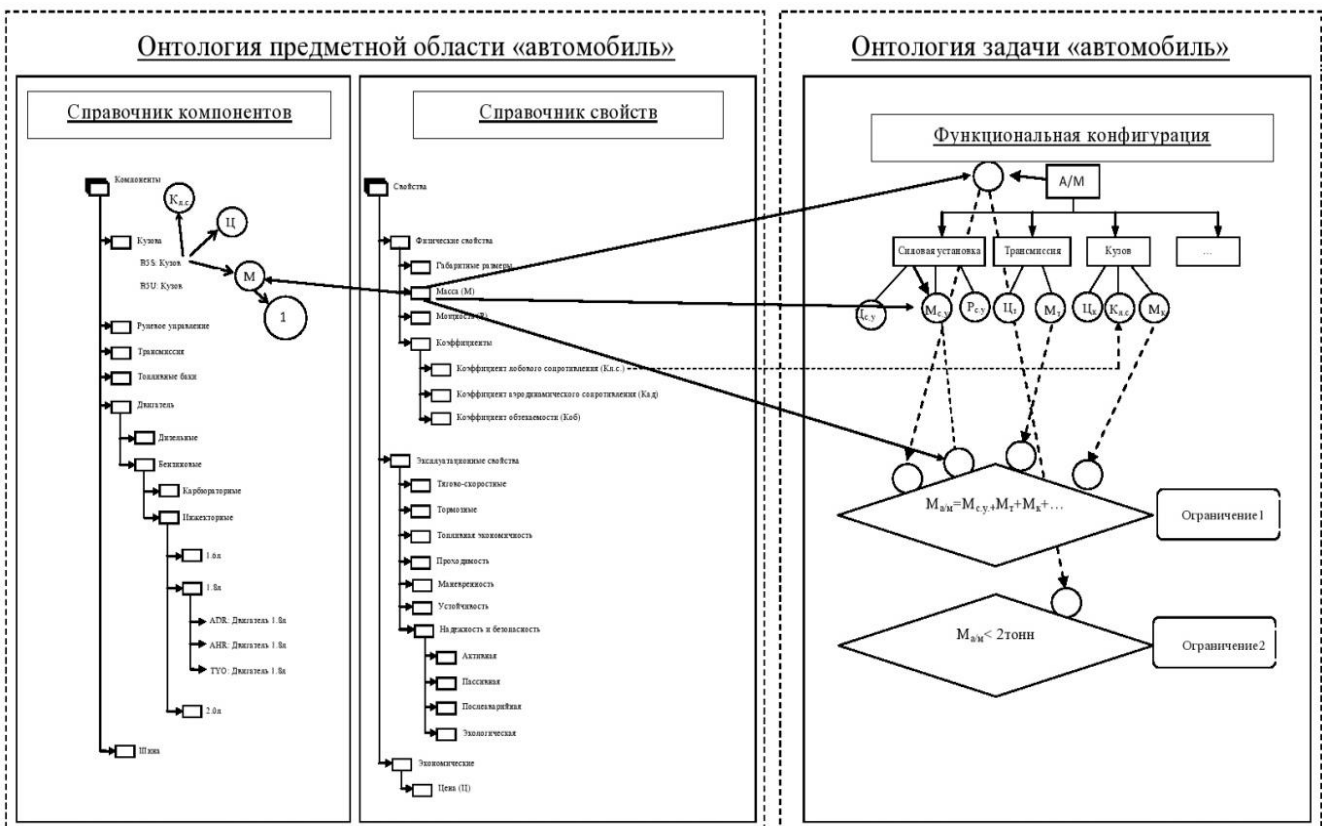


рис. 2. Прикладная онтология для синтеза конфигурации автомобиля

Например, автомобиль имеет свойство «запас хода», а топливный бак – «емкость». Для выбора топливного бака функциональная зависимость должна обеспечивать расчет минимальной емкости бака.

Такая зависимость может быть преобразована в следующее выражение во множестве интерпретаций:
 $((AM. Запас_хода * AM. Расход_топлива) / 100 - ТБ. Объем) \leq 0$.

Особенность данного подхода заключается в оперировании, при решении спектра задач управления конфигурацией, широким многообразием реальных параметров объектов с применением методов программирования в ограничениях. Описание природы каждого параметра является частью функциональной конфигурации. Каждый такой параметр отражает какое-либо свойство объекта и может содержать сложную систему значений, таких как набор дискретных значений, диапазоны, исключающие диапазоны и логические выражения. Например, параметр «Диапазон рабочих температур» имеет значение «от -20 до +80».

Заключение

Цель работы — разработка методов и алгоритмов синтеза конфигурации как задачи дискретного программирования основе решения задачи удовлетворения ограничений («Constraint Satisfaction Problem» или CSP).

Предложенные методы и алгоритмы были программно реализованы в виде ПО решателя задач синтеза конфигурации CMSolver [5, 6].

Решатель предназначен для решения относительно узкого класса задач структурного синтеза, в частности для задачи синтеза конфигурации изделия. В этом решателе, возможно, оптимизировать процесс поиска решений за счет применения методов быстрого доступа к данным.

В результате исследования были предложены технические решения подсистемы синтеза конфигурации и программная реализация макета CMSolver и проведено его тестирование.

Литература

1. Бениаминов Е.М., Болдина Д.М. Система представления знаний Ontolingua - принципы и перспективы // НТИ, сер. 2, 1999, №10 - М.: ВИНТИ.
2. Евгеньев Г.Б. Интеллектуальные системы проектирования М.: Издательство МГТУ им. Н.Э. Баумана, 2009, 256 с.
3. Рассел С., Норвиг П. Искусственный интеллект: современный подход. – Киев: Издательский дом “Вильямс”, 2006. 1408 с.
4. Карпенко А. П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учебное пособие для вузов / А. П. Карпенко. М: Изд-во МГТУ им. Н. Э. Баумана, 2014. 446 с.
5. Овсянников М.В., Буханов С.А. Управление конфигурацией методом программирования в ограничениях// Известия высших учебных заведений. Машиностроение. 2012. № 14. С. 70-75.
6. Vochkarev S.V., Petrochenkov A.B., Ovsyannikov M.V., Bukhanov S.A. Structural synthesis of complex electrotechnical equipment on the basis of the constraint satisfaction method // Russian Electrical Engineering. 2015. Т. 86. № 6. С. 362-366.