

Учет инженерных ограничений на различных этапах PLM

Pierre Grignon (Dassault Systèmes), Дмитрий Ушаков (ЛЕДАС)

Введение

В течение длительного времени термин “ограничение” в области CAD ассоциировался исключительно с понятием геометрического отношения между компонентами: совпадение (инцидентность), параллельность и т.д. Недавняя эволюция систем CAD к системам PLM (управление жизненным циклом изделия – ЖЦИ) придала термину “ограничение” более широкое значение, которое включает в себя любое отношение между значениями параметров (независимо от того, являются они геометрическими или нет). В простейшем случае значение одного параметра зависит от значений других и вычисляется по заданной формуле с использованием примитивных математических функций. В более сложных случаях значения нескольких параметров могут входить в произвольные взаимосвязи, выражаемые в виде систем уравнений и неравенств, “черных ящиков” (таким термином традиционно обозначаются функции с неизвестным алгебраическим определением), правил, таблиц и т.п. За подобными выразительными средствами закрепился термин “инженерные ограничения” (ИО), и именно они являются предметом рассмотрения настоящей статьи.

Прежде всего мы рассмотрим специфику использования инженерных ограничений на различных этапах проектирования изделия, планирования и сопровождения его производства, организации продаж и т.д., – то есть всего того, за чем закрепился устойчивый термин PLM. Затем мы поговорим о подсистеме поддержки инженерных ограничений в CATIA V5, которая была создана при непосредственном участии специалистов российской компании ЛЕДАС. Ну а завершится статья рассказом о семействе универсальных математических решателей, которые служат основой для различных подсистем решения ограничений в самых разных областях PLM. Один из таких решателей компания ЛЕДАС предлагает к внедрению всем разработчикам PLM-решений.

На каких этапах PLM необходимо учитывать инженерные ограничения

Инженерные ограничения широко используются на многих этапах управления ЖЦИ. Прежде всего, они играют важную роль на этапе концептуального проектирования изделия, когда отсутствие проработанной геометрии вынуждает проектировщика вести проектирование в терминах абстрактных геометрических, физических и экономических параметров будущего изделия. В этом контексте ограничения обеспечивают скелет, границы и широкие ориентиры, на основе которых в дальнейшем может

быть осуществлено детальное проектирование.

1 ИО при концептуальном проектировании

На этапе концептуального проектирования изделия геометрии, как таковой, еще не существует (или она существует в зачаточном состоянии). В этом случае инженер-проектировщик использует абстрактные параметры, которые выражают геометрические, физические и экономические параметры будущего изделия. Эти параметры связываются друг с другом рядом ограничений, которые можно разделить на три категории.

Во-первых, это универсальные ограничения для всего проекта:

- физические ограничения (на этапе концептуального проектирования они могут выражать общие взаимосвязи параметров с помощью фундаментальных физических законов – механики, термодинамики, электричества. К примеру, в судостроении на этом этапе важно учитывать закон Архимеда);
- геометрические ограничения (когда самой геометрии еще нет, важно заложить возможность взаимосвязи геометрических параметров, например: высота дома равна высоте одного этажа, умноженной на количество этажей, плюс высота чердака).

Ко второй группе относятся ограничения (неравенства), задающие направления и границы, которые не могут быть нарушены, например:

- стоимость должна быть меньше предельной стоимости
- общий вес должен быть меньше предельного веса.

К третьей группе относятся *индикаторы*, т.е. ограничения, которые обеспечивают информацию о состоянии проекта (обычно они задаются равенствами) – вычисление текущей стоимости продукта, его веса, динамических характеристик.

Как правило, ограничения на этапе концептуального проектирования являются достаточно тривиальными, но тем важнее определить правильный способ их удовлетворения. Произвольное задание параметров, соответствующих всем условиям, как правило, не устраивает проектировщика. Ему важно найти:

- оптимальное решение системы ограничений согласно заданной целевой функции (а также возможность оптимизировать найденное решение по другим критериям);
- ключевые параметры проекта;
- интервалы параметров, внутри которых содержатся все возможные решения задачи (чтобы видеть, какие параметры и в каких пределах можно менять во время детального проектирования).

2 ИО при детальном проектировании

При проектировании детали важно увязать её геометрические и физические свойства. Последние обычно выражаются в виде инженерных ограничений. В качестве примеров можно указать такие

типовые задачи:

- ✓ Спроектировать подшипник, удовлетворяющий как заданным динамическим нагрузкам, так и промышленным стандартам. Варьируемые параметры – тип подшипника (шариковый или роликовый), диаметр шариков/роликов (который может принимать только конечный ряд стандартных значений) и сорт стали (из заданного списка). При этом требуется обеспечить заданные динамические характеристики изделия.
- ✓ Спроектировать оптимальную упаковку. Варьируемые параметры – геометрические размеры пластиковой бутылки. Требуется обеспечить заданный объем.
- ✓ Спроектировать оптимальный рисунок протектора автомобильной шины. Варьируемые параметры – углы и расстояния заданной геометрии. Необходимо оптимизировать площадь сцепления колеса с дорогой.
- ✓ Спроектировать турбину двигателя. Варьируемые параметры – некоторые характерные расстояния. При этом необходимо сохранить физические свойства лопасти, зависящие от её площади и центра масс.

Несмотря на значительную семантическую разницу этих задач, все они могут быть решены с использованием одних и тех же методов спецификации (задания) и учета инженерных ограничений. Мы выделяем следующие специфические требования в этой области *PLM*:

- учет ИО, заданных в виде системы уравнений и неравенств;
- учет ИО, заданных в виде таблиц;
- учет “черных ящиков” (расчет объемов и центра масс для сложной геометрии);
- изменение набора определяемых параметров, легкий переход от прямой задачи к обратной;
- нахождение решения, ближайшего к текущим значениям параметров;
- предъявление пользователю нескольких решений с возможностью выбора наиболее подходящего для него.

Заметим здесь, что реализация функций, включающих геометрические вычисления, может быть выполнена, как минимум, двумя разными способами. Во-первых, интеграцией инженерных ограничений в геометрический решатель (например, решатель *LGS*, представленный в предыдущих номерах журнала, позволяет интегрировать в него пользовательские ИО и в перспективе решать оптимизационные задачи). Во-вторых, использованием специализированных алгоритмов, которые рассматривают геометрический решатель как “черный ящик”. Изменение значений параметров во время работы таких алгоритмов вызывает пересчет геометрических ограничений с помощью решателя.

3 ИО при совместном проектировании

Совместное проектирование (*collaborative design*) еще в большей мере, чем другие подобласти *PLM*, невозможно реализовать без инженерных ограничений. Они используются здесь для того, чтобы уменьшить информационные потоки между участниками процесса проектирования.

При совместном проектировании изделия инженеры согласовывают не геометрию (как правило, разные части проектируются разными участниками процесса проектирования), а параметры. Геометрические параметры должны быть согласованы между собой, а также соответствовать прочим параметрам (физическими и экономическими). Главная специфика здесь состоит в распределенности модели и невозможности загрузки всех данных на один компьютер, вследствие чего возникает значительный информационный поток между инженерами или между разными организациями. Следовательно, вычисления (нахождение совместных значений параметров) тоже должны быть распределенными и оптимизированными с точки зрения наименьшего числа итераций между участниками.

С технической точки зрения мы выделяем следующие важные требования к процессу удовлетворения инженерных ограничений при совместном проектировании:

- учет иерархии ограничений (задача совместного проектирования, как правило, является *переопределённой*, поэтому важно учитывать приоритет каждого ограничения);
- работа с “черными ящиками” (многие вычисления зависят от геометрических данных, которые в силу своего большого объема или содержащихся в них ноу-хау не могут быть загружены на вычислительный сервер, поэтому их приходится представлять в виде “черных ящиков” – функций с неизвестной семантикой, которые вычисляют выходные значения по заданным входным);
- диагностика несовместности и развитые методы её разрешения.

4 ИО при планировании и сопровождении производства

Ограничения планирования (такие как порядок выполнения работ, конфликт за общие ресурсы) сами по себе являются инженерными ограничениями. Кроме того, специфика различных производств такова, что при планировании необходимо учитывать самые разнообразные ограничения, которые не поддаются никакой классификации. Помимо собственно ресурсно-календарного планирования, в отдельную область выделяют планирование рабочих смен (операторов, экипажей и т.п.) и средств производства (станков, автомобилей, самолетов и т.п.), которые имеют свою специфику.

Характерными чертами этой области *PLM* являются:

- эффективный учет целого ряда стандартных ограничений (на длительность работы, на порядок работ, на конфликт за ресурсы и рабочее место, на время отдыха работника и т.п.);
- учет произвольных ограничений в дополнение к стандартным;
- развитые средства оптимизации (по времени, по стоимости).

5 ИО в системах PDM/DMS

Конфигурационное проектирование состоит в выборе подходящей конфигурации разрабатываемого изделия, компоненты которого описаны в таблицах баз данных (БД). Подобная функциональность используется также для технического обслуживания и послепродажного сервиса. Поэтому важно учитывать инженерные ограничения при выполнении запросов к БД. Здесь потребуются следующие свойства решателя ограничений:

- таблицу реляционной БД можно рассматривать как ограничение, заданное явным образом перечислением всех возможных сочетаний значений параметров, – решатель должен уметь удовлетворять таким ограничениям;
- возможность смешивать в рамках одной модели алгебраические выражения с табличными ограничениями, что необходимо для удобства спецификации (задания) и для эффективности процесса удовлетворения таким смешанным ограничениям;
- решатель должен оптимизировать запросы к БД, чтобы ускорить процесс проектирования.

Возможности работы с инженерными ограничениями в системе CATIA V5

В системе CATIA V5 средства работы с инженерными ограничениями предлагаются вместе с группой продуктов **Knowledgeware**. Уже в базовую версию системы включена возможность создания пользовательских параметров и соотнесения их (как друг с другом, так и с геометрическими параметрами) посредством *формул*. Формула связывает произвольное количество параметров и определяет простое отношение между её единственным выходным параметром (левой частью) и входными параметрами (правой частью). Например, параметр *mass.1* может быть вычислен с помощью простой формулы $mass.1 = density.1 \cdot l \cdot h \cdot L$, что интерпретируется так:

$$mass.1 \leftarrow density.1 \cdot l \cdot h \cdot L$$

Когда параметр в правой части формулы (*density.1, l,*

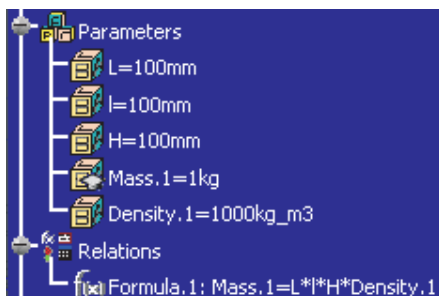


Рис. 1. Пример использования формулы в CATIA V5 Part Design

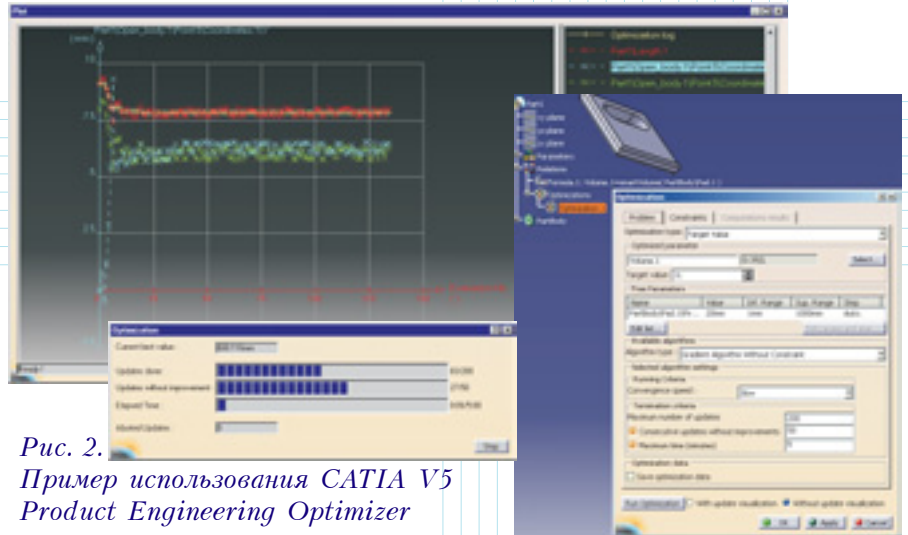


Рис. 2. Пример использования CATIA V5 Product Engineering Optimizer

L или *h*) меняет свое значение, параметр *mass.1* пересчитывается (см. пример отображения формулы в дереве свойств CATIA на рис. 1).

Параметры формулы могут быть самых разных типов: пользовательские параметры, параметры-свойства (*features*) или внутренние параметры свойств (такие как длины и углы). Вот пример формулы, использующей геометрическое свойство (*Pad.3*) в качестве параметра:

$$mass.2 = density_material.2 \cdot smartVolume(Pad.3)$$

Эта формула пересчитывается каждый раз, когда модифицируется геометрия призмы *Pad.3*, – к примеру, когда меняется её высота или форма основания.

Продукт **CATIA V5 Knowledge Advisor** повышает выразительный уровень формул. С его помощью параметры могут связываться посредством *правил* – простейших программ, содержащих условные выражения и операторы присваивания. Правила могут менять значения сразу для нескольких параметров, поэтому они уже не ассоциируются с параметрами, а являются разновидностью такой категории CATIA V5, как ограничение.

Другим видом ограничений являются проверки (*checks*) – логические условия, которые автоматически сигнализируют пользователю в случае любого изменения параметров, ведущего к их нарушению.

Важно понимать, что формулы, правила и проверки имеют направленную природу, – они вычисляют значения выходных параметров по значениям входных. При этом в графе ограничений не допускается наличие циклов, т.е. такой цепочки из параметров и ограничений, по которой можно ходить бесконечно. Система **Knowledgeware** автоматически проверяет отсутствие циклов и сообщает об этом пользователю.

Несколько другую функциональность, связанную с решением инженерных ограничений, предлагает еще один **Knowledgeware**-продукт – **Product Engineering Optimizer (PEO)**. Он предлагает алгоритмы оптимизации, а также методы удовлетворения ограничениям

и исследования пространства проектирования. Поэтому *PEO*, в отличие от формул и ограничений *Knowledge Advisor*, не включается в цикл автоматического обновления параметров. Пользователь *PEO* участвует в процессе решения задачи оптимизации непосредственно: помимо указания критерия оптимизации и списка параметров оптимизации он имеет возможность выбрать подходящий алгоритм, специфицировать критерий останова (такой как максимальное время счета и др.) и явным образом запустить расчет (см. пример использования *PEO* на рис. 2). Если найденное решение его удовлетворяет, пользователь может принять полученные значения параметров и синхронизировать их с моделью *CATIA*.

Релиз *V5R8* помимо описанных выше возможностей *Knowledge Advisor* обогатился спецификацией системы уравнений (*Set of Equations*). Начиная с релиза *V5R10* аналогичные возможности (но с более широкой функциональностью) появились и в продукте *PEO* – под названием *удовлетворение ограничений* (*Constraint Satisfaction*). Эти разработки создавались при непосредственном участии специалистов российской компании ЛЕДАС на основе универсального математического решателя *NemoNext* (собственный продукт ЛЕДАС). Ниже мы разберем эти новые возможности, а затем рассмотрим лежащий в их основе математический решатель.

Возможности *Set of Equations* и *Constraint Satisfaction*

1 Система уравнений и неравенств

Функциональность *Knowledge Advisor Set of Equations* допускает решение уравнений и неравенств в рамках *CATIA V5*. Пользователь специфицирует уравнения и параметры, которые он рассматривает как неизвестные (они также называются *выходными параметрами* для *Set of Equations*, в то время как оставшиеся параметры называются

входными и рассматриваются в процессе решения уравнений как константы).

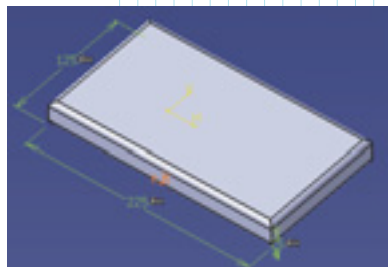
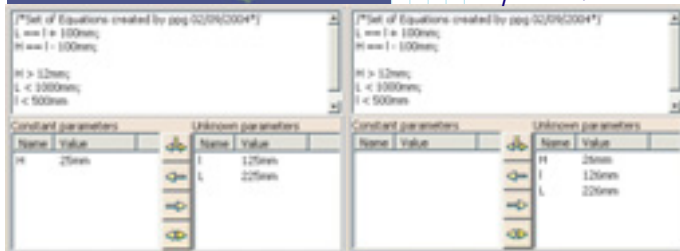


Рис. 3. Деталь с ассоциированным свойством *Set of Equations*



Далее вычисления запускаются либо по команде пользователя, либо автоматически – при изменении значения одного из входных параметров.

2 Поиск ближайшего решения

С инженерной точки зрения несколько особенностей *Set of Equations* и *Constraint Satisfaction* отличают их от традиционных математических решателей. Во-первых, для того чтобы гарантировать минимальное изменение начального проекта, *Set of Equations* обеспечивает поиск решения, которое является максимально близким к первоначальным значениям его выходных параметров. Такое поведение гарантирует, что геометрия не придет в несовместимое состояние после решения.

3 Поиск нескольких решений

Constraint Satisfaction предлагает возможность поиска нескольких решений для одной задачи. Эта особенность полезна для инженеров, которым необходимо иметь альтернативные решения задачи или обеспечить различные стартовые позиции (на этапе предварительного проектирования), т.е. предусмотреть различные направления для последующего детального проектирования.

В конце концов, поиск нескольких решений полезен, даже когда задача вообще не имеет решений (что достаточно часто бывает в проектировании из-за переопределённой природы механических задач). В этом случае задачу необходимо ослабить (путем удаления одного или нескольких ограничений), и затем сравнить несколько её решений для того, чтобы найти компромиссный вариант (см. рис. 4÷7 с примерами поиска нескольких решений).

4 Переход от прямой задачи к обратным

В любое время уравнения могут редактироваться,

а входные/выходные параметры могут быть изменены (их значения могут быть модифицированы, параметры могут переключаться между входными и выходными), – см. рис. 8.



Рис. 4. Пример недоопределенной задачи *Constraint Satisfaction*

Line	F1(x)	x	y
1	25	2.67298	4.22054
2	24.99951708	-4.999810408	-0.029145224
3	24.999389523	4.999879819	-0.029426314
4	24.999762402	-0.000794899	-4.999976177
5	24.999214972	-0.323998027	4.989413017
6	24.999486586	-0.007368779	4.999942227
7	24.999814827	4.327020616	2.505369089
8	24.999888620	2.148514112	4.505279859
9	24.999844420	3.252767364	3.797303211
10	24.999997156	2.439577425	4.364454342
11	24.999789961	2.304045760	4.437472591
12	24.99999322	2.961794051	4.013654721
13	24.99964909	2.846172395	4.130033967
14	24.999666077	3.134235788	3.806272986
15	24.99999919	3.439452222	3.423664681
16	24.999579602	3.877624745	3.156335653
17	24.999624217	3.425602889	3.642091134
18	24.999999274	3.739481915	3.510077294
19	24.99999242	4.121440285	2.830852231
20	24.99999434	4.024796415	2.983054022

Рис. 5. Фрагмент из списка 250 найденных решений для предыдущей задачи

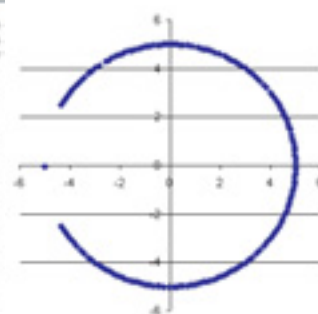


Рис. 6. Геометрическое представление 250 найденных решений

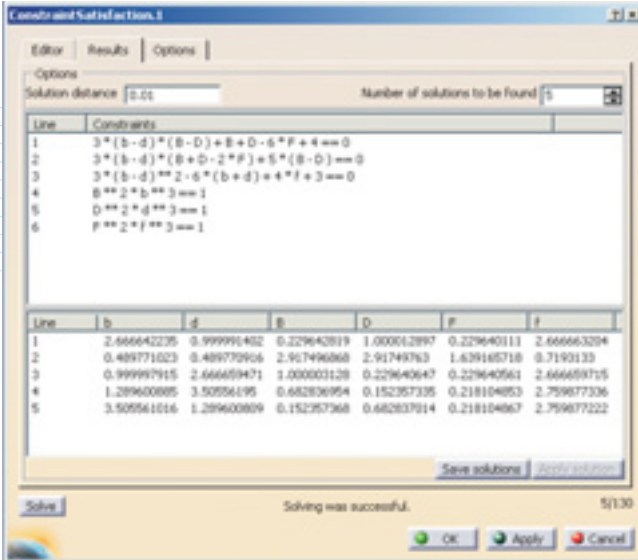


Рис. 7. Табличный вид нескольких решений

Чтобы объяснить необходимость в такой функциональности, рассмотрим задачу проектирования пружины (рис. 9). После того как созданы описывающие пружину математические уравнения, проектировщик может манипулировать *Set of Equations* несколькими способами. Например, с точки зрения инженера пружина может проектироваться для того, чтобы отвечать на заданную нагрузку заданным сжатием (эти значения рассматриваются как входные для *Set of Equations*). В этом случае количество витков и диаметр пружины вычисляются (как выходные значения) в *Set of Equations*. Однако задача может быть обращена, — т.е., зная диаметр и количество витков, можно вычислять максимально допустимую нагрузку (геометрическая точка зрения).

Возможные будущие улучшения в *Set of Equations* и *Constraint Satisfaction*

Constraint Satisfaction будет эволюционировать со следующими релизами *CATIA*. Мы можем анонсировать, по меньшей мере, два возможных расширения его функциональности. Нахождение частичного решения и диагностика переопределенности позволит инженерам решать несовместные задачи, в которых невозможно удовлетворить всем ограничениям одновременно. В этом

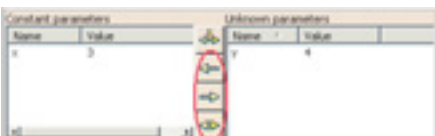


Рис. 8. Возможность редактирования списка параметров

случае пользователи хотят получить некоторую разумную конфигурацию, которая удовлетворяет большинству наложенных ограничений, в то время как остальные нарушаются с минимальными погрешностями. Диагностика несовместности покажет, где расположен источник противоречия, и какие ограничения должны быть удалены из модели для того, чтобы последняя стала совместной.

Еще одно возможное направление развития *Constraint Satisfaction* состоит в расчете *интервальной оценки* пространства всех решений задачи. Часто такая оценка для пользователя более важна, чем точное решение. Дело в том, что она показывает, где могут быть все решения задачи, как влияет на эти интервалы изменение значений входных параметров, а также что произойдет в случае искусственного сокращения интервалов для некоторых выходных параметров.

Универсальный решатель ограничений

1 Методы удовлетворения ИО

Из математики известно много различных методов разрешения инженерных ограничений — от дискретных алгоритмов из области исследования операций до вычислительных методов локальной и глобальной оптимизации для непрерывных областей. На практике, однако, пользователь задает смешанные задачи, которые включают в себя как различные типы параметров (дискретные и непрерывные), так и различные виды ограничений (алгебраические выражения, таблицы, “черные ящики”). Поэтому так важно иметь универсальные методы решения таких задач.

Такие методы разрабатываются в рамках *программирования в ограничениях (Constraint Programming — CP)*. Это сравнительно новая область

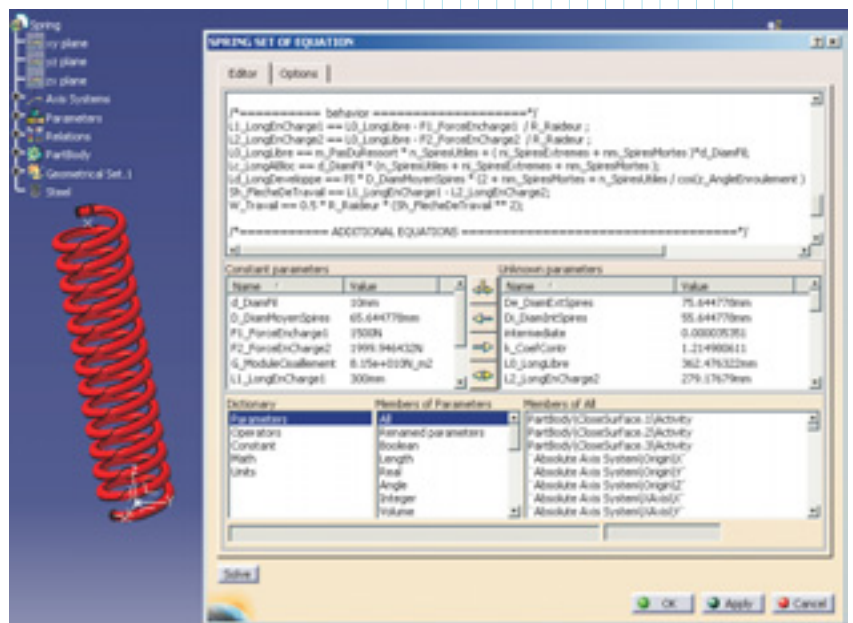


Рис. 9. Использование *Set of Equations* для проектирования пружины

исследований, возникшая на стыке исследования операций, вычислительных методов и искусственного интеллекта. В рамках последней дисциплины первые *CP*-алгоритмы применялись для распознавания образов и оптимизации поиска с возвратами при решении дискретных задач.

Один из *CP*-подходов был разработан в Российском научно-исследовательском институте искусственного интеллекта (**РосНИИ ИИ**) и получил название “метод недоопределенных моделей”. Его основная идея состоит в рассмотрении значения каждого параметра модели как только частично известного. Когда мы начинаем решать задачу, мы знаем только то, что потенциальные значения лежат внутри областей определения параметров (например, углы между двумя прямыми на плоскости могут принимать значения от 0 до 2π). Рассматривая по очереди все ограничения, наложенные на параметры, можно вывести больше информации об их потенциальных значениях, – эта информация используется для уточнения недоопределенных значений. В конце концов, решатель либо докажет, что задача не имеет решений в заданных областях, либо получит максимально уточненные недоопределенные значения, которые могут быть использованы для уменьшения времени работы алгоритмов поиска точного решения внутри заданных областей.

Наиболее традиционный способ представления недоопределенного значения состоит в использовании интервала, который задает минимально и максимально возможные потенциальные значения для каждого параметра. Перебирая все ограничения, решатель уточняет эти интервалы путем увеличения нижней или уменьшения верхней границы.

Такие интервальные решения также полезны для инженеров, которые могут видеть настоящие области всех решений задачи. Более того, редукция интервалов – достаточно быстрый алгоритм, поэтому на практике инженеры могут мгновенно увидеть, как изменение той или иной границы для одного параметра влияет на интервальные оценки других параметров.

2 Семейство решателей *Nemo*

На основе аппарата недоопределенных моделей было создано несколько программных продуктов. В их числе – математический решатель **UniCalc** с простым интерфейсом, позволяющим пользователю сформулировать любую задачу с помощью алгебраических выражений, а также мощная расширяемая среда для программирования в ограничениях **NeMo+**, которая включает в себя множество специальных методов для редукции интервалов, поиска точного решения и оптимизации. В 1998 году на Шестой национальной конференции по искусственному интеллекту *NeMo+* была названа лучшей российской интеллектуальной системой года.

В том же году этот продукт приобрела известная французская компания **Dassault Systèmes**.

Позднее *Dassault* с помощью российских специалистов создала свой собственный математический решатель **NemoNext**, который представляет собой полный функциональный аналог *NeMo+*, но превосходит последний в производительности и качестве.

Представленные в этой статье продукты *CATIA V5 Knowledge Set of Equations* и *Constraint Satisfaction* были созданы на основе решателя *NemoNext* с участием специалистов ЗАО “ЛЕДАС” – российской софтверной компании, специализирующейся на разработке и внедрении интеллектуальных решателей для *PLM*-продуктов. ЛЕДАС также предлагает свой собственный математический решатель под названием **LeMo**, который включает в себя классические алгоритмы из области программирования в ограничениях, различные методы решения задач на непрерывных и дискретных областях, методы эвристического поиска и оптимизации в ограничениях. Сам по себе или в комбинации с другими интеллектуальными решателями от ЛЕДАС (такими как описанные в предыдущих статьях геометрический решатель и планировщик) он привносит мощные инженерные возможности в любое решение из области *PLM*.

О группе компаний **Dassault Systèmes**

Мировой лидер в области решений *PLM* – группа *Dassault Systèmes* поставляет свои продукты более чем 70 тысячам заказчиков в 80 странах мира. Пионер на рынке программного обеспечения для *3D*, *Dassault Systèmes* с 1981 разрабатывает и обслуживает *PLM*-приложения, которые поддерживают промышленные процессы и обеспечивают объемное видение всего жизненного цикла изделия – от концепции до утилизации. Портфель предложений включает в себя интегрированные *PLM*-решения для разработки изделия (*CATIA*®, *ENOVIA*®, *DELMIA*™, *SMARTEAM*®), систему проектирования изделий (*SolidWorks*) и *3D*-компоненты (ядро *ACIS*®) от *Spatial Corp*. Акции *Dassault Systèmes* котируются на биржах *Nasdaq* (*DASTY*) и *Euronext Paris* (#13065, *DSY.PA*).

За более подробной информацией обращайтесь на сайт <http://www.3ds.com>

ЗАО “ЛЕДАС”

ЗАО “ЛЕДАС” является частной софтверной компанией, основанной в 1999 году. Компания расположена в Новосибирске. Специалистами ЛЕДАС разработана собственная уникальная технология, основанная на программировании в ограничениях. Компания применяет её для решения различных задач из области *PLM*, включая *CAD/CAM*, – главным образом, в партнерстве с *Dassault Systèmes S.A.* (Франция).

За более подробной информацией обращайтесь на сайт <http://www.ledas.com>

