

Автоматизация выбора рациональных режимов токарной обработки тонкостенных деталей

© П.А. Еремейкин¹, А.Д. Жаргалова²

¹Институт машиноведения им. А.А. Благонравова РАН,
Москва, 101990, Россия

²МГТУ им. Н.Э. Баумана, Москва, 105005, Россия

Рассмотрена проблема назначения рациональных режимов резания при точении тонкостенных заготовок. На точность результата обработки существенное влияние оказывают деформации заготовки, вызванные силами закрепления и резания. Обработка деформированной детали приводит к значительным отклонениям от заданных размеров. В ряде случаев стандартные методы обеспечения заданного допуска (с использованием дополнительной технологической оснастки) оказываются неприменимыми ввиду существующих ограничений. В качестве альтернативного решения разработана методика назначения рациональных режимов закрепления и обработки с учетом жесткости заготовки, которые обеспечивают меньшие деформации в процессе обработки. Для определения значений параметров обработки и закрепления предложен подход, основанный на численном моделировании деформаций. В целях сокращения трудоемкости расчетов и повышения точности обработки разработана интегрированная система автоматизированного выбора технологических параметров. Основное внимание уделено техническим аспектам программной реализации такой системы, а также описанию приемов разрешения проблем, выявленных в процессе разработки.

Ключевые слова: токарная обработка, деформация, тонкостенная деталь, разработка программной системы, численное моделирование, метод конечных элементов

Введение. Обработка тонкостенных заготовок требуется в различных отраслях промышленности, при этом возникает ряд проблем, связанных с определением методов обработки. Указанные проблемы в своей основе имеют следующее противоречие: способ изготовления детали необходимо выбрать таким образом, чтобы, с одной стороны, обеспечить ее высокое качество, а с другой — минимизировать себестоимость. Решение такого рода проблем особенно актуально, например, при производстве изделий аэрокосмической промышленности [1]. В этом случае объемы выпуска составляют единицы изделий, поэтому вариант обработки с применением дополнительной технологической оснастки может существенно повысить себестоимость деталей. Таким образом, требуется определить, возможны ли альтернативные способы обработки. К таким способам относится методика назначения «мягких» режимов резания при использовании стандартной оснастки.

Далее рассмотрен процесс разработки программной системы для решения вопроса о возможности изготовления тонкостенной детали

с применением рациональных режимов обработки. Цель проекта — продемонстрировать принципиальную осуществимость автоматизированного выбора рациональных режимов токарной обработки цилиндрической детали и создать программную основу, которая позволит применять указанный метод для решения более общих технологических задач. В ходе работы необходимо решить следующие задачи.

1. Разработать программный продукт, который обладает функционалом, необходимым для назначения рациональных режимов резания для тонкостенного цилиндра, и имеет расширяемую архитектуру для обобщения подхода на различные типоразмеры деталей.

2. Выявить основные проблемы, связанные с автоматизацией методики назначения «мягких» режимов резания, и найти способы их решения.

3. Определить дальнейшие направления работ по методологической поддержке описанных подходов к проблеме.

Постановка задачи. При обработке тонкостенных деталей существенную роль в достижении требуемой точности играют погрешности, связанные с деформацией заготовки под действием сил резания и закрепления.

Рассмотрим пример закрепления цилиндрической заготовки. Расчетная схема нагружения для случая обработки детали на токарном станке проходным резцом по внешней поверхности при установке в трехкулачковом патроне может быть задана эскизом (рис. 1).

До закрепления заготовка находится в недеформированном состоянии (штриховые линии). После закрепления происходит деформация заготовки под действием сил со стороны кулачков. Кроме сил закрепления в деформацию заготовки вносят свой вклад также силы резания, тепловые процессы и другие факторы.

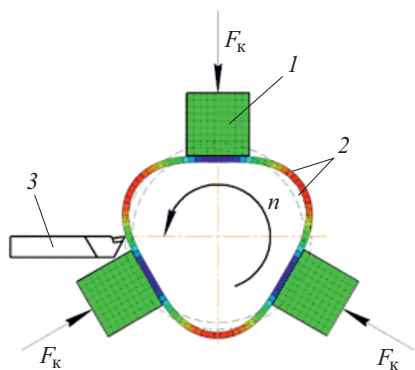


Рис. 1. Схема нагружения тонкостенной детали при токарной обработке:
1 — кулачок патрона; 2 — цилиндрическая заготовка; 3 — токарный резец

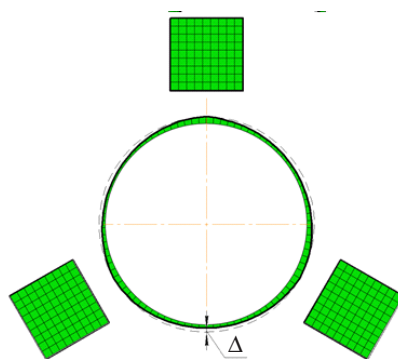


Рис. 2. Результат обработки тонкостенной детали

Обработка деформированной заготовки приводит к неравномерному снятию припуска. Таким образом, результирующий профиль заготовки (в зависимости от условий обработки) может не удовлетворять заданным ограничениям. На рис. 2 изображена форма готовой детали. Буквой Δ обозначена величина огранки.

В настоящее время основные методы решения предполагают применение специального технологического оборудования. Например, в случае закрепления заготовки в разжимной оправке требуется хранить определенную номенклатуру оснастки под разные диаметры. Аналогичный недостаток характерен для сырых кулачков, растачиваемых для каждого размера. При использовании технологического заполнителя (например, сплава Вуда) увеличивается время подготовки к обработке и требуется соблюдение специальных мер хранения (многие легкоплавкие сплавы токсичны) [2].

В связи с изложенными факторами предлагается рассмотреть альтернативный подход — разработку программной системы информационной поддержки, позволяющей с помощью CAE-моделирования прогнозировать результат обработки тонкостенных деталей для заданных условий, а также обладающей возможностью доступно и гибко регулировать необходимые параметры процесса, что является актуальной научно-практической задачей.

К достоинствам предлагаемого подхода следует отнести сокращение продолжительности технологической подготовки производства и уменьшение затрат, связанных с закупкой и хранением дополнительного оборудования и оснастки.

Указанный подход позволяет выработать методику назначения рациональных параметров технологического процесса в рамках единого информационного пространства [3]. Методика реализована в форме программной информационной поддержки технологических процессов обработки тонкостенных деталей.

Методы решения проблемы и принятые допущения. В основе решения поставленной в работе задачи назначения рациональных режимов резания лежит метод CAE-моделирования. В частности, использовали метод конечных элементов (МКЭ) как наиболее применимый для программного определения деформаций, методы математического моделирования, системного анализа, 3D-моделирование [4].

Спроектированная программная система через графический интерфейс пользователя запрашивает необходимые данные для построения МКЭ-модели. Введенные значения параметров используются для автоматического построения модели с последующим расчетом в популярной CAE-системе Abaqus. Результаты расчета отображаются в интерфейсе в виде диаграмм узловых перемещений.

Программная система предполагает расширение функционала с помощью загрузки пользовательских расчетных случаев. В целях

демонстрации рассмотрена обработка тонкостенной цилиндрической детали (см. рис. 1 и 2). При этом приняты следующие допущения: воздействие резца на деталь учитывается в виде точечной трехкомпонентной силы, нагрев детали в рассмотрение не принимается. Считается, что сила резания в процессе обработки постоянна, глубина резания не изменяется, материал заготовки однородный.

Разработка автоматизированной системы. В соответствии с современными принципами разработки программных систем для реализации проекта была выбрана модульная архитектура, основанная на объектно-ориентированном подходе. В качестве языка программирования используется Java, определенный по ряду критериев (таких, как распространенность, скорость разработки, гибкость и безопасность с точки зрения обработки исключений).

Полноценная реализация МКЭ — чрезвычайно сложная с технической точки зрения задача, для решения которой требуется огромная исследовательская и проектная работа коллектива профессионалов. Однако благодаря доступности программных интерфейсов в таких программных средствах, как ANSYS, Abaqus, уже готовые решения можно использовать в другом программном обеспечении. В данном проекте предпочтение отдано системе Abaqus, так как она обладает более широкими возможностями интеграции с внешними системами. Например, для описания моделей в Abaqus применяется язык программирования общего назначения Python, в отличие от ANSYS, где используется специализированный язык APDL. В функциональном плане ANSYS и Abaqus практически эквивалентны (по крайней мере в рамках рассматриваемой задачи).

Наиболее разумный способ построения модульной архитектуры приложения — использование одной из существующих программных платформ. Для языка программирования Java такими платформами являются NetBeans Platform и Eclipse RCP. Основное различие платформ состоит в том, что пользовательский интерфейс NetBeans основан на библиотеке Swing [5], которая является стандартной для языка Java и поставляется в комплекте разработчика JDK (Java Development Kit). Для Eclipse роль библиотеки графического интерфейса играет SWT, поведение которой может изменяться в зависимости от операционной системы. Кроме того, в Eclipse модульная система реализована в соответствии со стандартом OSGi (Open Service Gateway initiative), в то время как в NetBeans применяется специфическая модульная система, хотя в документации NetBeans также указана частичная поддержка OSGi. После учета всех особенностей каждой из платформ был сделан выбор в пользу NetBeans. На это решение, кроме указанных факторов, повлияло наличие большого числа доступных ресурсов для освоения платформы, таких как проекты с откры-

тым исходным кодом и обучающие официальные материалы, а также подробная документация.

Как показали предыдущие работы по данной теме [1, 5, 6], в составе системы следует выделить такие основные компоненты:

- модуль расчета режимов резания (калькулятор режимов резания);
- загрузчик моделей;
- графический редактор параметров;
- модуль визуализации результатов;
- модуль генерации отчетов.

Каждый компонент представлен в виде отдельного модуля в терминологии платформы NetBeans. Коммуникация между основными частями обеспечивается вспомогательными составляющими, которые также представляют собой модули NetBeans, но в них, как правило, описываются интерфейсы межкомпонентного взаимодействия.

Калькулятор режимов резания. Для первичного определения режимов резания применяется соответствующий модуль. Детально разработка калькулятора режимов резания рассмотрена в [7]. Основная особенность данного модуля заключается в том, что его работа организована по принципу экспертных систем. Как показал анализ литературы [8], описание традиционных методов определения силы резания не структурировано, алгоритм расчета может изменяться в зависимости от исходных данных, используется большое количество табличных коэффициентов. Для решения такого рода задач хорошо зарекомендовали себя экспертные системы.

Java-библиотека Drools предлагает разработчику весь необходимый функционал для работы со знаниями. Например, в составе библиотеки имеется машина вывода, работающая по алгоритму Rete [9]. Также предусмотрен механизм представления знаний в виде продукционных правил на предметно-ориентированном языке DRL (Drools Rule Language). Такое представление может быть записано явно в текстовой файл или в виде Excel-таблицы, которая компилируется в DRL во время работы программы. Ниже приведена типичная структура описания продукционного правила на языке DRL:

```
rule "name"  
    attributes  
    when  
        LHS  
    then  
        RHS  
end
```

Описание правила начинается с его имени, далее следует перечисление специальных атрибутов, модифицирующих работу правил. После ключевого слова «when» указываются условия, при которых срабатывает правило (антецедент), а после слова «then» — голова (консеквент) правила, т. е. действия, которые будут выполнены после срабатывания.

В текстовом формате были сохранены те знания, которые изначально представлены в виде формул. Табличные данные, такие как выбираемые коэффициенты в формулах, сохранены в Excel-файл.

На рис. 3 показан фрагмент Excel-файла с таблицей для определения глубины резания. Первые три столбца относятся к левой части продукционного правила, задающей условие выполнения его правой части, к которой относится только последний столбец. Столбцы LOCK-ON-ACTIVE и PRIORITY определяют значения специальных атрибутов правила.

RuleTable Depth of cut					
CONDITION	CONDITION	CONDITION	LOCK-ON-ACTIVE	PRIORITY	ACTION
step : TurningStepBean()					
kind in \$param	type = \$param	diameter > \$1 , diameter <= \$2			modify(step) {setT(\$param)}
Вид обработки	Тип точения	Диаметр			Глубина резания
(EXTERNAL_TURNING, GROOVING, PARTING)	ROUGH	6, 10	true	-1	0.85
		10, 18	true	-2	0.85
		18, 30	true	-3	1.05
		30, 50	true	-4	1.05
		50, 80	true	-5	1.25
		80, 100	true	-6	1.35
	SEMI_ROUGH	6, 10	true	-7	0.65
		10, 18	true	-8	0.7
		18, 30	true	-9	0.8
		30, 50	true	-10	0.85
		50, 80	true	-11	1.1
		80, 100	true	-12	1.15

Рис. 3. Структура таблицы в Excel

На рис. 4 представлен интерфейс калькулятора режимов резания: слева изображено окно для ввода информации пользователем, справа — окно результатов. После ввода данных пользователь нажимает на кнопку «Рассчитать». При этом сохраненные значения параметров интерпретируются библиотекой Dgoal как совокупность фактов, к которым применяются описанные ранее продукционные правила. В процессе применения правил вычисляются значения выходных данных, которые отображаются в окне результатов.

The image shows a software interface for calculating cutting regimes, divided into two windows:

- CalculatorInput Window (Входные параметры):**
 - Диаметр поверхности D, мм: 40
 - Тип точения: Черновое
 - Вид обработки: Внешнее точение
 - Материал заготовки: Сталь
 - Вид заготовки: Поковка
 - Материал инструмента: T15K10
 - Период стойкости резца, мм: 20.0
 - Шероховатость поверхности Ra: 0.63
 - Радиус при вершине резца, мм: 1
 - Кнопка: Рассчитать
- CalculatorOutput Window (Результат):**
 - Глубина резания, мм: 1.050
 - Подача, мм/об: 0.450
 - Скорость резания V, м/мин: 163.969
 - Сила резания P, Н: 545.720
 - Момент резания M, Нм: 10.914
 - Потребляемая мощность N, кВт: 1.462
 - Частота вращения n, об/мин: 1304.821
 - Кнопка: Вставить

Рис. 4. Интерфейс калькулятора режимов резания

Загрузчик моделей и редактор параметров. Каждый расчетный случай имеет свою специфику, поэтому разработанная система позволяет создавать произвольные модели. Модель состоит из трех частей: Python-сценарий, список параметров, база данных. Сценарий на языке Python необходим для описания расчетного задания внутри Abaqus. Предполагается, что некоторый набор параметризованных типовых сценариев обеспечит достаточную гибкость в реальных условиях без необходимости написания новых сценариев. Список параметров указывает, какие величины требуется ввести или выбрать пользователю, для того чтобы Python-сценарий отработал корректно. Сбор данных осуществляет разработанная система, пользователь не имеет представления о внутреннем описании задания с помощью Python, от него требуется только ввод конкретных значений. База данных обеспечивает систему информацией о константных значениях. Например, в базе данных модели должны быть сохранены значения модуля упругости материалов, коэффициентов Пуассона и т. п. При выборе материала по названию загружаются соответствующие физические константы и происходит их установка в качестве аргументов Python-сценария. Необходимость ввода таких констант с клавиатуры отпадает.

Система поддерживает два возможных формата хранения моделей: Excel-таблицы и база данных SQLite. В обоих случаях Python-сценарий хранится непосредственно в виде исполняемого кода, записанного в ячейку Excel-таблицы или базы данных. Список параметров хранится в отдельной таблице (или Excel-листе) с указанием отношения типа «родитель — потомок». Каждая строка соответствует определенному параметру или группе параметров и содержит как минимум название, комментарий и значение по умолчанию.

Интерфейс редактора параметров представлен на рис. 5, на котором приведен список параметров после загрузки файла с описанием расчетного случая обработки тонкостенного цилиндра. Значение параметра можно вводить в соответствующее поле с клавиатуры либо выбрать из выпадающего списка.

Название	Комментарий	Значение
Модель		Тонкостенный цилиндр
Деталь		
• Внутренний радиус	Внутренний ра...	0.028
• Внешний радиус	Внешний радиус	0.03
• Длина	Длина	0.06
• Допуск	Допуск	0.001
Материал	Материал	
• Название	Название	Сталь
• Модуль упругости	Модуль упругос...	2.1E11
• Коэффициент Пуассона	Коэффициент ...	0.29
Кулачки	Кулачки	
• Сила	Сила закреплен...	100.0
• Количество	Количество	3.0
Размеры	Размеры	
• Длина	Длина	0.015
• Высота	Высота	0.015
• Ширина	Ширина	0.015

Рис. 5. Интерфейс редактора параметров

Модуль визуализации результатов. Визуализация результатов сводится к построению полярных диаграмм перемещений. Результатом работы Python-сценария является текстовый файл, в который записаны значения узловых перемещений. Этот файл обрабатывается созданной системой и отображается в окне визуализации результатов. Вид этого окна для рассматриваемого примера приведен на рис. 6.

В данном случае отображены перемещения, вызванные только силами закрепления. Синими сплошными линиями обозначены поля допусков, заданные пользователем в редакторе параметров. Красные маркеры отображают профиль заготовки после закрепления, а черные сплошные линии — номинальный внешний диаметр детали.

Для задания положения профиля деформированной заготовки используется контрольная плоскость (рис. 7). Положение контрольной плоскости передается в сценарий Python, который обеспечивает запись в обменный файл значений узловых перемещений для требуемых точек.

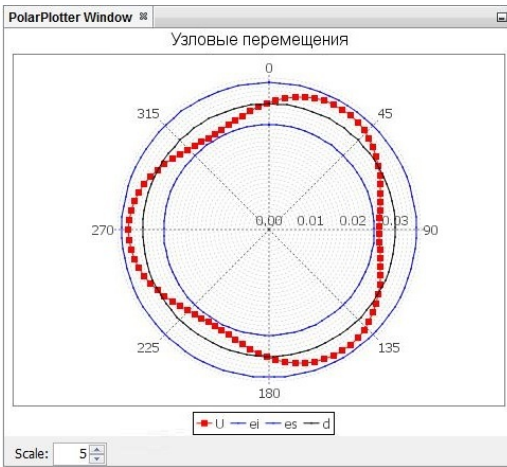


Рис. 6. Окно визуализации

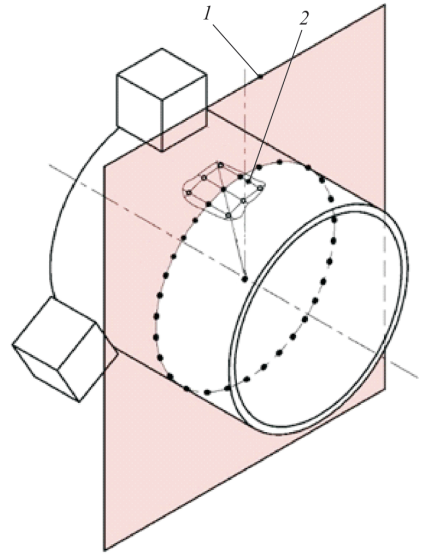


Рис. 7. Схема задания положения профиля:
1 — контрольная плоскость; 2 — сетка

Модуль генерации отчетов. Имеется возможность сохранить отчет о результатах работы программы в формате Microsoft Word. Отчет включает таблицу исходных данных, графическое цветное изображение полей перемещений и их графики.

Построение МКЭ-модели. Как было указано ранее, расчетное задание формируется сценарием языка Python. С учетом принятых допущений разработан скрипт для определения значений перемещений. В таблице приведены все параметры модели.

Параметры модели

Обозначение и название	Схема
<i>Деталь</i>	
<p>L — длина детали</p> <p>$d_{\text{внутр}}$ — внутренний диаметр</p> <p>$d_{\text{внешн}}$ — внешний диаметр</p> <p>E_d — модуль упругости материала детали</p> <p>μ_d — коэффициент Пуассона материала детали</p>	

Обозначение и название	Схема
<i>Сила резания</i>	
<p>F_o — осевая компонента</p> <p>F_T — тангенциальная компонента</p> <p>F_P — радиальная компонента</p> <p>l — осевое положение точки приложения</p> <p>α — угловое положение точки приложения</p>	
<i>Кулачки</i>	
<p>Δ — радиальное смещение</p> <p>N — количество кулачков</p> <p>w — ширина кулачка</p> <p>h — высота кулачка</p> <p>t — толщина кулачка</p> <p>E_k — модуль упругости материала кулачков</p> <p>μ_k — коэффициент Пуассона материала кулачков</p>	

На рис. 8 показаны результаты расчета в виде полей напряжений и перемещений. Следует отметить, что на диаграммах виден результат приложения точечной силы резания. Как было принято в допущениях, модуль упругости кулачков намного больше модуля упругости материала детали, поэтому кулачки не подвержены деформациям.

Интерфейс программы. С точки зрения пользователя программный продукт в целом выглядит, как показано на рис. 9. Графический интерфейс программы типичен для разработок на платформе NetBeans и позволяет закреплять окна внутри программы практически в любых комбинациях.

На рис. 9 скрыты окна калькулятора режимов резания. В верхнем левом углу закреплено окно редактора параметров, под ним расположено окно визуализации результатов. Верхний правый угол занимает окно предварительного просмотра заготовки, под которым находится окно текстового вывода служебных сообщений.

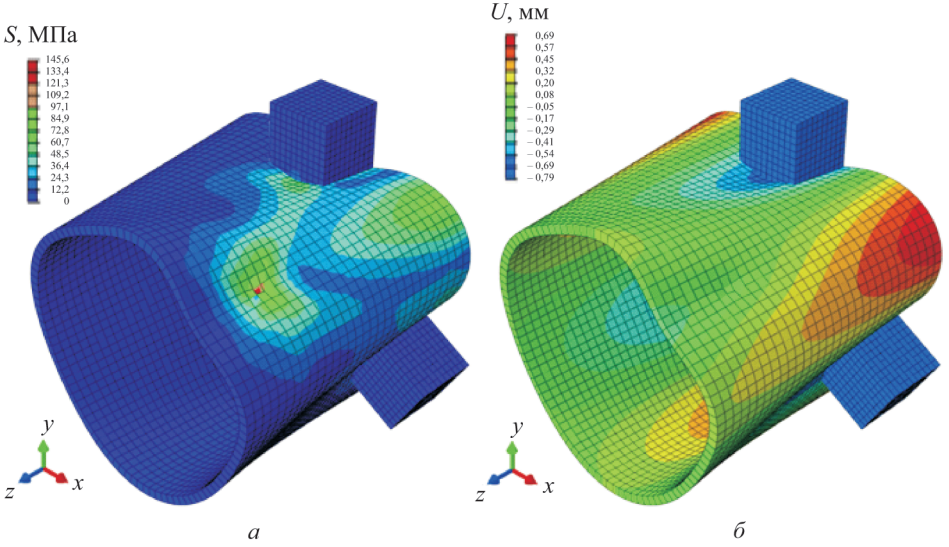


Рис. 8. Результаты моделирования:
 а — поле напряжений S ; б — поле перемещений U

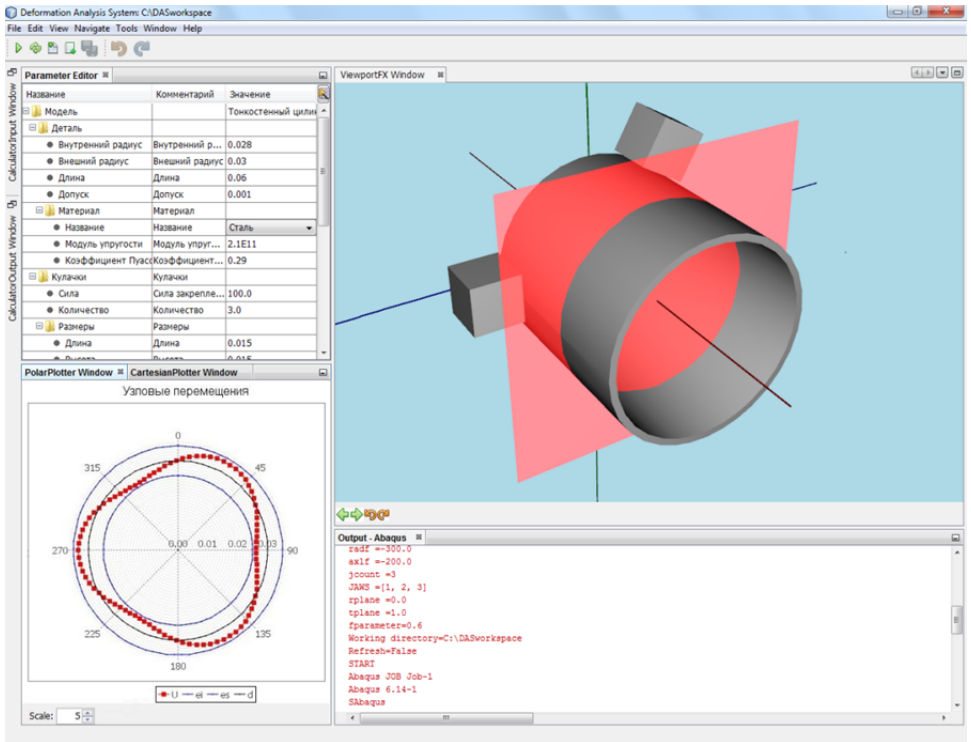


Рис. 9. Пользовательский интерфейс программы

Выводы. Результатом работы стала интегрированная система поддержки принятия решения, позволяющая спрогнозировать деформации, возникающие в процессе обработки тонкостенных деталей, и тем самым определить пригодность выбранных условий обработки для точения заготовки. Кроме того, составлена модель токарной обработки тонкостенного цилиндра. Программные модули, входящие в состав системы, способствуют повышению эффективности первичного расчета режимов резания, моделирования заданного расчетного случая и анализа его результатов. Таким образом, программный продукт обладает функционалом, необходимым для назначения рациональных режимов резания для тонкостенного цилиндра.

Благодаря применению объектно-ориентированного языка программирования Java и модульной программной платформы NetBeans достигнуто условие расширяемости архитектуры. Новые модули NetBeans могут быть добавлены к полностью готовой системе без необходимости перекомпиляции программы целиком [10]. Новые типоразмеры деталей могут быть добавлены в систему благодаря загрузке всех атрибутов, специфичных для заданного расчетного случая, из внешнего файла модели. Однако такое решение имеет и недостатки. Например, чтобы составить файл модели, разработчик должен обладать знаниями в области программирования сценариев Abaqus.

При работе над проектом выявлены проблемы, связанные с автоматизацией методики назначения рациональных режимов резания. К таким проблемам относятся, например, вопросы выбора технических средств реализации, способа проектирования расширяемой архитектуры, обеспечения гибкости при предварительном расчете режимов резания, проблемы взаимодействия с внешней CAE-системой и др. Для каждой из них предложены соответствующие решения. Например, для обеспечения гибкости при предварительном расчете режимов резания рассмотрена возможность использования подсистемы работы со знаниями.

Следующие работы в рамках темы планируется посвятить экспериментальному подтверждению применимости системы для решения практических задач, а также усовершенствованию используемых МКЭ-моделей обработки и их распространению на более общие технологические задачи. Отдельного внимания заслуживает усовершенствование программного кода и функциональных возможностей системы.

ЛИТЕРАТУРА

- [1] Campbell F.C. *Manufacturing Technology for Aerospace Structural Materials*. Elsevier Ltd, 2006, 603 p.
- [2] Гаврюшин С.С., Жаргалова А.Д., Лазаренко Г.П., Семисалов В.И. Метод определения условий механической обработки тонкостенных деталей. *Известия высших учебных заведений. Машиностроение*, 2015, № 11, с. 53–60.

- [3] Евгеньев Г.Б., ред. *Основы автоматизации технологических процессов и производств*. Москва, Изд-во МГТУ им. Н.Э. Баумана, 2015, 441 с.
- [4] Гаврюшин С.С., Барышников О.О., Борискин О.Ф. *Численный анализ элементов конструкций машин и приборов*. 2-е изд. Москва, Изд-во МГТУ им. Н.Э. Баумана, 2014, 480 с.
- [5] Böck H. *The Define Guide to NetBeans Platform 7*. Apress, 2012, 558 p.
- [6] Еремейкин П.А. Система анализа деформаций тонкостенных деталей при токарной обработке. [Электрон. ресурс]. *Молодежный научно-технический вестник*, 2016, № 5. URL: <http://sntbul.bmstu.ru/doc/840485.html> (дата обращения 02.11.2016).
- [7] Еремейкин П.А. Разработка программного инструмента для расчета режимов резания токарной обработки «Калькулятор». [Электрон. ресурс]. *Молодежный научно-технический вестник*, 2015, № 7. URL: <http://sntbul.bmstu.ru/doc/791980.html> (дата обращения 02.11.2016).
- [8] Дальский А.М., Косилова А.Г., Мещеряков Р.К., Суслов А.Г. *Справочник технолога-машиностроителя*. Т. 2. Москва, Машиностроение, 2003.
- [9] Drools Documentation. *JBossDeveloper*, 2016. URL: <http://docs.jboss.org/drools/release/6.4.0.Final/drools-docs/html/>
- [10] Boudreau T., Tulach J., Wielenga G. *Rich Client Programming: Plugging into the NetBeans Platform*. Prentice Hall, 2007, 640 p.

Статья поступила в редакцию 21.11.2016

Ссылку на эту статью просим оформлять следующим образом:

Еремейкин П.А., Жаргалова А.Д. Автоматизация выбора рациональных режимов токарной обработки тонкостенных деталей. *Инженерный журнал: наука и инновации*, 2017, вып. 2. <http://dx.doi.org/10.18698/2308-6033-2017-02-1587>

Еремейкин Петр Александрович — аспирант Института машиноведения им. А.А. Благонравова РАН. Автор двух публикаций. Область научных интересов — разработка программного обеспечения. e-mail: eremeykin@gmail.com

Жаргалова Аягма Дашибалбаровна — старший преподаватель кафедры «Компьютерные системы автоматизации производства» МГТУ им. Н.Э. Баумана. e-mail: azhargalova@bmstu.ru

Automation of assignment of rational thin-walled parts turning mode

© P.A. Eremeykin¹, A.D. Zhargalova²,

¹Mechanical Engineering Research Institute of the Russian Academy of Sciences,
101990 Moscow, Russia

²Bauman Moscow State Technical University, 105005 Moscow, Russia

This article deals with the problem of rational turning mode assignment for thin-walled workpieces. In this instance, the deformations of the workpiece, caused by cutting and clamping forces, strongly affect the accuracy of the result. Deformed workpiece processing leads to significant deviations from a predetermined size. Traditional thin-walled parts processing methods (using machining attachments) are inappropriate in some cases. For example, it is not profitable to use special machining attachments for small batch processing. So, the method of rational turning mode assignment is proposed as an alternative. One can assign the rational turning mode based on accounting rigidity of workpiece. This "soft" mode is required to provide small processing deformations. CAE modeling method is appropriate to resolve the parameters of rational mode. Depending on modeling results, one can choose apply either traditional method or process the workpiece with standard machining attachments (e.g. three-jaw chuck). To use the method on practice, it is required to reduce its applicability. To this end the software system was developed. This paper considers technical aspects of implementing the system and related problems resolution.

Keywords: turning, deformation, thin-walled part, modelling, software development

REFERENCES

- [1] Campbell F.C. *Manufacturing Technology for Aerospace Structural Materials*. Elsevier Ltd, 2006, 603 p.
- [2] Gavryushin S.S., Zhargalova A.D., Lazarenko G.P., Semisalov V.I. Metod opredeleniya usloviy mekhanicheskoy obrabotki tonkostennykh detaley. *Izvestiya vysshih uchebnykh zavedeniy. Mashinostroenie*, 2015 № 11, pp. 53-60
- [3] Böck H. *The Define Guide to NetBeans Platform 7*. Apress, 2012, 558 p.
- [4] Eremeykin P.A. Sistema analiza deformatsiy tonkostennykh detaley pri tokarnoy obrabotke [Electronic resource]. *Molodezhnyy nauchno-tekhnicheskyy vestnik*, 2016, № 5 URL: <http://sntbul.bmstu.ru/doc/840485.html> (accessed 02 November 2016)
- [5] Eremeykin P.A. Razrabotka programmogo instrumenta dlya rascheta rezhimov rezaniya tokarnoy obrabotki kalkulyator [Electronic resource]. *Molodezhnyy nauchno-tekhnicheskyy vestnik*, 2015, № 7. URL: <http://sntbul.bmstu.ru/doc/791980.html> (accessed 02 November 2016)
- [6] Kosilova A.G., Meshcheryakov R.K. *Spravochnik tekhnologa mashinostroyitelya. Vol. 2*. Moscow, Mashinostroenie, 1986, 418 p.
- [7] Drools Documentation. *JBossDeveloper*, 2016. URL: <http://docs.jboss.org/drools/release/6.4.0.Final/drools-docs/html/>
- [8] Evgenev G.B., Gavryushin S.S., Groshev A.V., Ovsyannikov M.V., Shilnikov P.S. *Osnovy avtomatizatsii tekhnologicheskikh processov i proizvodstv*. Moscow, BMSTU, 2015, 441 p.
- [9] Gavryushin S.S., Baryshnikova O.O., Boriskin O.F. *Chislennyy analiz elementov konstruktsiy mashin i priborov*. Moscow, BMSTU, 2014, 480 p.

- [10] Boudreau T., Tulach J., Wielenga G. *Rich Client Programming: Plugging into the NetBeans Platform* Prentice Hall, 2007, 640 p.

Eremeykin P.A. — graduate student, Mechanical Engineering Research Institute of the Russian Academy of Sciences. e-mail: eremeykin@gmail.com

Zhargalova A.D. — senior lecturer, «Computer Systems of Manufacture Automation» Bauman Moscow State Technical University. e-mail: azhargalova@bmstu.ru