

О. С. ЮЛЬМЕТОВА, А. Н. СИСЮКОВ, Р. Р. ЮЛЬМЕТОВА

## СОЗДАНИЕ БАЗЫ ДАННЫХ НЕПАРАМЕТРИЧЕСКИХ КРИТЕРИЕВ ОЦЕНКИ МИКРОГЕОМЕТРИИ ФУНКЦИОНАЛЬНЫХ ПОВЕРХНОСТЕЙ

Обосновывается необходимость создания базы данных непараметрических критериев оценки микрогеометрии функциональных поверхностей. Рассматриваются этапы ее проектирования с выбором инструментальных средств разработки и описанием структурно-логической схемы построения базы данных.

*Ключевые слова:* функциональные свойства, непараметрические критерии оценки, проектирование базы данных, система управления базой данных.

Эффективность эксплуатации изделий приборостроения существенным образом зависит от качества изготовления входящих в их состав деталей, которые имеют функциональные поверхности, подвергаемые в процессе эксплуатации механическим, электрическим, трибологическим и другим видам нагрузки. Физические характеристики функциональных поверхностей, получаемых, как правило, посредством механической обработки, в значительной степени определяются формой и величиной микронеровностей (шероховатостью) последних. Требования к качеству функциональных поверхностей, а также возможность эффективного управления их механической обработкой в значительной степени зависят от технических методов контроля и измерения шероховатости.

Микрогеометрия функциональных поверхностей деталей, подвергнутых механической обработке, определяет эксплуатационные свойства изделия и измеряется с использованием метода контактно-щупового контроля. Этот метод основан на контактном движении датчика-преобразователя по шероховатой поверхности, в результате которого получают профиль поверхности. Обработка этого профиля с целью выявления степени влияния шероховатости на то или иное функциональное свойство может проводиться различными способами.

Традиционно отечественные и зарубежные исследования шероховатости проводятся с использованием параметрического описания профиля. В частности, Ю. Г. Шнейдером была выявлена взаимосвязь эксплуатационных характеристик изделий, таких как износостойкость, усталостная прочность, коррозионная стойкость, с геометрическими параметрами их микро-рельефа — средним арифметическим отклонением профиля, средним шагом неровностей по вершинам, радиусом закругления неровностей и т.д. [1].

В дальнейшем исследования влияния микрогеометрии на функциональные и эксплуатационные свойства велись на базе предложенных В. А. Валетовым более информативных непараметрических критериев оценки, а именно функций плотностей распределения ординат и тангенсов углов наклона профилей [2].

На сегодняшний день актуальна задача создания единой информационной модели, позволяющей в упорядоченном виде хранить данные (координаты профиля), с помощью которых можно построить графические изображения непараметрических кривых, описывающих оптимальную микрогеометрию для конкретных функциональных свойств изделия. При этом поиск оптимальной микрогеометрии подразумевает экспериментальное определение наилучшей микрогеометрии для конкретного функционального свойства. В рамках экспериментального поиска необходимо:

- изготовить серию образцов;
- зафиксировать виды и режимы их обработки;

- снять профили поверхностей образцов;
- построить необходимые критерии оценки микрогеометрии;
- подвергнуть образцы одинаковому функциональному воздействию;
- осуществить контроль величины функциональных характеристик;
- выявить образец, максимально отвечающий предъявленному требованию.

Микрорельеф наилучшего образца принимается в качестве оптимального для данного функционального свойства.

База данных, содержащая непараметрические критерии оценки микрогеометрии функциональных поверхностей, позволит:

- накапливать информацию по виду непараметрических кривых для конкретных функциональных свойств;
- получать сведения о виде оборудования и инструменте, режимах резания и материале, необходимых для получения оптимальной микрогеометрии, описывающей требуемое функциональное свойство.

В основу разработки базы данных (БД) положена структурно-логическая схема, представленная на рис. 1.



Рис. 1

Из схемы видно, что условно в БД выделены три информационных блока: конструкторский, технологический и метрологический. Каждый блок несет специфическую информацию, например, в конструкторском содержатся характеристики, определяемые конструктором: марка материала, форма детали или узла, а также ее наименование; в технологическом — перечень двух последних операций, так как именно на этом этапе формируется рабочий микрорельеф поверхности детали или узла; наименование и марка станка и инструмента, а также

режимы обработки — эти данные необходимы для поиска и технологического обеспечения оптимальной микрогеометрии. Немаловажным является блок, содержащий дополнительную информацию об обработке, например сведения о дисперсности притирочных паст, которая во многом определяет свойства полученного в процессе доводки микрорельефа.

Последний блок содержит метрологическую информацию, а именно: данные о длине пройденной трассы в процессе определения микрогеометрии контактно-щуповым методом, радиус использованного щупа, направление движение щупа, информацию о произведенной фильтрации. При этом все блоки объединяет конкретное функциональное свойство.

Процесс создания базы данных включает ряд последовательных этапов:

- выбор системы управления базой данных СУБД (реляционной — РСУБД, объектной — ОСУБД, объектно-реляционной — ОРСУБД, транзакционной встраиваемой);
- определение вида разрабатываемого прикладного обеспечения (web-ориентированное, GUI-ориентированное (graphical user interface), консольное, сервис-ориентированное);
- анализ существующих сред разработки прикладного программного обеспечения (ПО) и языков программирования с выбором средств разработки;
- проектирование структуры базы данных и ее объектов (типы, таблицы, виды, индексы, хранимые функции, процедуры, запросы и т.д.);
- разработка приложения (построение интерфейса и организация взаимодействия с СУБД).

На первом этапе проектирования следует определить тип СУБД.

Из реляционных СУБД наиболее распространенными вариантами будут MS SQL Server 2005/2008 и MySQL Server 5. Но реляционная СУБД, дополненная средствами объектной СУБД, является более предпочтительной по соотношению производительность—функциональность. Среди наиболее известных ОРСУБД можно выделить Oracle Database и PostgreSQL. Также вызывают интерес встраиваемые СУБД, например, Oracle Berkeley DB или MySQL Embedded.

В дальнейшем предполагается использовать создаваемое ПО в распределенной среде с единой БД, поэтому от использования варианта встраивания придется отказаться. Распределенная РСУБД MS SQL Server 2005 и ОРСУБД Oracle являются платными продуктами и тоже исключаются. В силу свободного распространения PostgreSQL остановим свой выбор на ней.

На втором этапе необходимо определить вид разрабатываемого ПО. Выбор GUI-приложения обусловлен возможностью выполнения большей части логики ПО на рабочей станции пользователя, что разгружает серверную составляющую приложения в клиент-серверной архитектуре.

Следующие этапы проектирования БД предполагают определение языка доступа к БД и среды программирования.

Выбор языка программирования предлагается осуществить из следующего списка: C++, Delphi (Pascal), C#, Python, Java. Так как их функциональность примерно одинакова, то критерием выбора будет кроссплатформенность и снижение трудозатрат.

Из предложенного перечня языков наиболее гибким средством является C++. Однако этот язык в большей степени удобен в системном программировании. Написание ПО с графическим интерфейсом, даже с использованием GUI-библиотек (Qt) и алгоритмических библиотек (boost), будет с точки зрения времени разработки нерациональным решением. Недостатком языка Delphi (Pascal) является отсутствие его кроссплатформенности.

В силу гибкости, безопасности языка C# и производительности SQL Server 2005 интересна связка языка C# (среда программирования .NET) с SQL Server. Отказ от нее обусловлен ограничением платформы .NET семейством операционных систем (ОС) Windows.

Объектно-ориентированный язык Python идеально подойдет в качестве надстройки над главным приложением при построении баз знаний. Наличие C++ и Java-вариаций Python

делает его сравнительно гибким. Но простота декомпиляции байт-кода исключает возможность его использования для создания базового приложения.

Таким образом, в качестве средства разработки ПО выбираем бесплатную среду NetBeans IDE 6.5 от Sun и язык программирования Java SE [3].

Выбранные компоненты, участвующие в процессе создания и функционирования приложения, дают следующие преимущества:

- бесплатны для разработчика;
- кроссплатформенны (не ограничены ОС);
- функциональны и удобны в использовании.

Такой выбор программных средств позволяет разработчику создавать коммерческие приложения без начальных инвестиций в средства разработки и сопутствующее ПО.

На заключительном этапе проектирования определяется структура БД, формируется графический интерфейс (рис. 2), создается логика приложения, посредством которой организуется взаимодействие пользователя с СУБД.

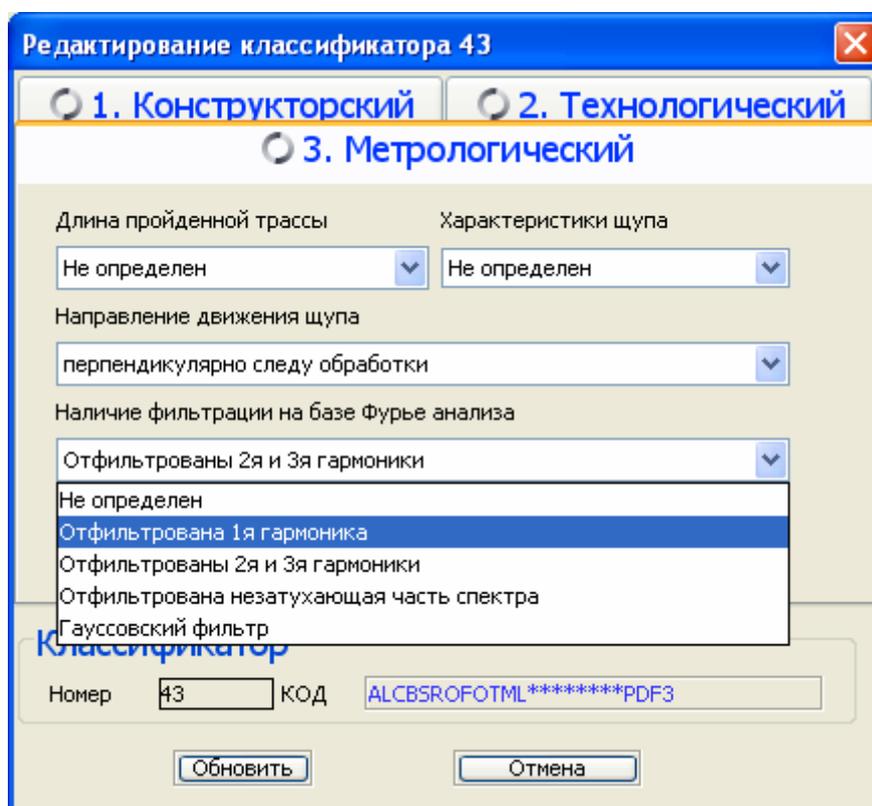


Рис. 2

Разработанная БД состоит из модифицируемых справочников классификатора, основной таблицы учета, схемы настроек, таблиц модулей (конструкторский, технологический, метрологический). Для разгрузки основной логики приложения часть кода перенесена на СУБД и реализована в виде хранимых процедур.

Спроектированное приложение позволяет вести учет проводимых экспериментов, регистрировать новые классификаторы, осуществлять поиск в базе данных по частичным критериям. В дальнейшем предполагается расширить функционал большей интерактивностью, возможностью совмещения измеренного профиля с эталонным для конкретного функционального свойства.

Созданная БД позволит накапливать и хранить сведения по виду непараметрических кривых, соответствующих определенному уровню функционального свойства, способу их

технологического обеспечения и метрологического контроля, что играет важную роль в процессе оптимизации микрогеометрии для функциональных свойств при решении проблемы повышения качества изделий.

Работа выполнена при поддержке РФФИ, проект № 10-08-00158а.

#### СПИСОК ЛИТЕРАТУРЫ

1. Шнейдер Ю. Г. Эксплуатационные свойства деталей с регулярным микрорельефом. СПб: СПбГУИТМО (ТУ), 2001. 264 с.
2. Мусалимов В. М., Валетов В. А. Динамика фрикционного взаимодействия. СПб: ПИМаш, 2006. 168 с.
3. Хорстманн К. С., Корнелл Г. Java 2. Т. 2. Тонкости программирования. М.: Вильямс, 2007.

#### *Сведения об авторах*

- Ольга Сергеевна Юльметова** — Санкт-Петербургский государственный университет информационных технологий, механики и оптики, кафедра технологии приборостроения; ассистент; E-mail: [www.ralli@rambler.ru](mailto:www.ralli@rambler.ru)
- Артем Николаевич Сисюков** — канд. техн. наук; Санкт-Петербургский государственный университет информационных технологий, механики и оптики, кафедра технологии приборостроения; ассистент; E-mail: [i@an.spb.ru](mailto:i@an.spb.ru)
- Рина Ринатовна Юльметова** — студентка; Санкт-Петербургский государственный университет информационных технологий, механики и оптики, кафедра технологии приборостроения; E-mail: [rina-yulmetova@yandex.ru](mailto:rina-yulmetova@yandex.ru)

Рекомендована кафедрой  
технологии приборостроения

Поступила в редакцию  
14.12.09 г.