

Sp. 6.7
495

ISSN 0136-3549
0320-3544

TALLINNA
POLÜTEHNILISE INSTITUUDI
TOIMETISED

495

ТРУДЫ ТАЛЛИНСКОГО
ПОЛИТЕХНИЧЕСКОГО
ИНСТИТУТА

ТРИ
'80

АВТОМАТИЗАЦИЯ
ТЕХНОЛОГИЧЕСКОГО
ПРОЕКТИРОВАНИЯ
ПРОЦЕССОВ
МЕХАНИЧЕСКОЙ
ОБРАБОТКИ





Ep. 6.7

495

TALLINNA POLÜTEHNILISE INSTITUUDI TOIMETISED

**TPI
'80**

ТРУДЫ ТАЛЛИНСКОГО ПОЛИТЕХНИЧЕСКОГО ИНСТИТУТА

УДК 658.512+621.9+681.3

● АВТОМАТИЗАЦИЯ
ТЕХНОЛОГИЧЕСКОГО
ПРОЕКТИРОВАНИЯ
ПРОЦЕССОВ
МЕХАНИЧЕСКОЙ
ОБРАБОТКИ

Машиностроение ХУШ

Таллин 1980

С о д е р ж а н и е

1.	Ю.А. Пяэсукке. Структура программного обеспечения системы автоматизации проектирования технологических процессов.	3
2.	Н.Х. Викторова, О.М. Эльяс. Разработка базового программного обеспечения поднастраиваемых САПР-ТП.	II
3.	Ю.И. Прууден, Р.А. Кютнер. Монитор системы автоматизации проектирования технологических процессов (требования и реализация).	19
4.	А.П. Калья. Индивидуальные базы данных для технологического проектирования.	29
5.	Ю.В. Папстел . К вопросу о глубине кодирования формы детали.	37
6.	Ю.Э. Рийвес. Применение множества приоритетов для выбора модели станка.	45

Таллинский политехнический институт
Труды ТПИ № 495
АВТОМАТИЗАЦИЯ ТЕХНОЛОГИЧЕСКОГО ПРОЕКТИРОВАНИЯ
ПРОЦЕССОВ МЕХАНИЧЕСКОЙ ОБРАБОТКИ
Машиностроение ХУШ

Редактор Г. Гроссшмидт, Технический редактор В. Ранник.
Сборник утвержден коллегией Трудов ТПИ 30 мая 1980 года.
Подписано к печати 28 ноября 1980 г. Бумага 60x90/16.
Печ. л. 3,5 + 0,25 приложение, Уч.-изд. л. 3,0. Тираж 300.
МВ-09070. Ротапринт ТПИ, Таллин, ул. Коскла, 2/9. Зак. № 629
Цена 45 коп.

© ТПИ, Таллин, 1980



СТРУКТУРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СИСТЕМЫ
АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ ТЕХНОЛОГИЧЕСКИХ
ПРОЦЕССОВ

На современном уровне развития вычислительной техники, программных средств общего назначения и методики проектирования технологических процессов система автоматизации проектирования технологических процессов (САПР-ТП) разрабатывается для ее реализации на ЕС ЭВМ в комплексе с мини-ЭВМ типа СМ.

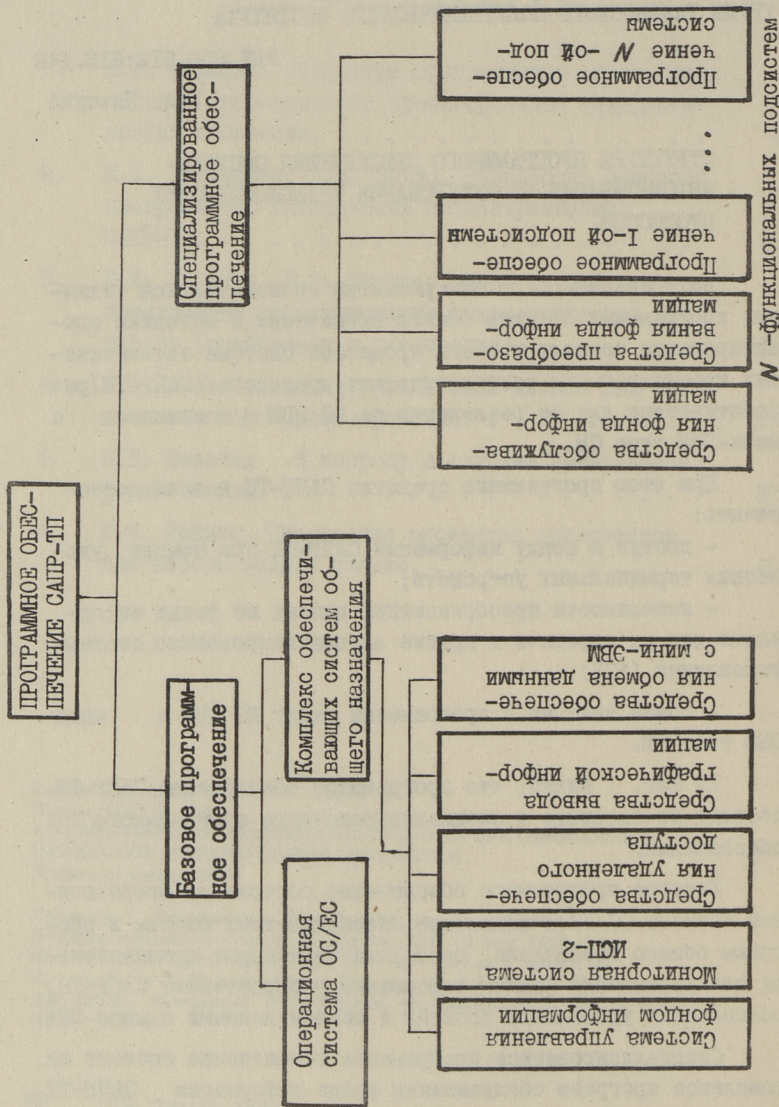
При этом программные средства САПР-ТП должны обеспечить:

- доступ к фонду информации САПР-ТП при помощи удаленных терминальных устройств;
- возможности преобразования данных из фонда информации для их передачи в другие автоматизированные системы управления (АСУ);
- обмен данными и программами между ЕС ЭВМ и мини-ЭВМ типа СМ.

Из фиг. 1 видно, что программное обеспечение САПР-ТП состоит из базового и специализированного программного обеспечения.

Базовое программное обеспечение состоит из операционной системы ОС/ЕС и комплекса обеспечивающих систем и программ общего назначения, последние из которых предназначены для управления фондом информации, подсистемами САПР-ТП, обеспечения удаленного доступа и обмена данными с мини-ЭВМ.

Специализированное программное обеспечение состоит из комплекса программ обслуживания фонда информации САПР-ТП, программ преобразования ее в форматы, доступные стандарт-



N — функциональных подсистем

Фиг. 1. Структура программного обеспечения.

ным средствам управления данными ОС/ЕС, и программного обеспечения конкретных подсистем САПР-ТП.

1. Базовое программное обеспечение. Работа САПР-ТП предъявляет определенные требования к конфигурации ЕС ЭВМ и операционной системе ОС/ЕС. Проведенный анализ показал, что основные требования к операционной системе для управления работой САПР-ТП, которые необходимо соблюсти при ее генерации, следующие:

- обеспечение мультипрограммного режима с переменным числом задач (режим MVT);
- включение пакета машинной графики и макрокоманд для работы с алфавитно-цифровыми дисплеями типа ЕС-7066;
- включение телекоммуникационных методов доступа BTAM и TCAM, а также всех стандартных методов доступа ОС/ЕС;
- обеспечение мультитерминального управления и системной мониторной системы;
- включение возможностей рестарта и свертки-развертывания.

По мере развития и увеличения круга пользователей САПР-ТП предполагается включение в ОС/ЕС возможности разделения времени.

Выбор режима MVT ОС/ЕС с возможностями рестарта и свертки-развертывания связан с удовлетворением требования повышения надежности работы системы путем применения более совершенных средств защиты, распределения и управления основной памятью ЭВМ.

Комплекс обеспечивающих систем и программ общего назначения САПР-ТП включает:

- систему ведения фонда информации, предназначенную для создания и манипулирования информационно-нормативными данными, а также данными выполненных инженерно-технических решений и др., составляющих банк данных САПР-ТП;
- мониторную систему ИСП-2 [1], предназначенную для создания, управления и поддержания работы отдельных функциональных подсистем САПР-ТП;
- программные средства обеспечения удаленного доступа к фонду информации;

- средства вывода графической информации на графопостроители, плоттеры и т.д.;

- средства, обеспечивающие обмен данными и текстами программ между ЕС ЭВМ и мини-ЭВМ.

Система ведения фонда информации допускает цепную организацию данных и удовлетворяет основным требованиям, предъявляемым к структурам данных САПР-ТП. Эти требования следующие:

- реализация т.н. $m:n$ соотношений между наборами записей данных (например, соотношения "режущий инструмент - материал режущей части" на фиг. 2 или "оборудование - цех/предприятие" и т.д.) в фонде информации;

- реализация соотношений между различными элементами данных одного и того же набора (например, описание компоновки приспособлений, описание размерных и геометрических связей между поверхностями и др.);

- описание нескольких параллельных соотношений между одними и теми же наборами записей данных.

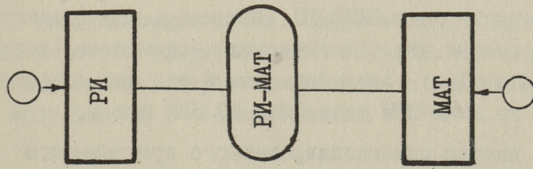
Эти соотношения реализуются адресными указателями и изображены линиями на фиг. 2.

Мониторная система ИСП-2 используется для управления подсистемами САПР-ТП и подробно изложена в [2].

Средства обеспечения удаленного доступа к фонду информации САПР-ТП предназначены для обработки и удовлетворения запросов с ближних и удаленных терминальных устройств. Эти средства включают программы ввода-вывода запросов, чтения данных из фонда информации и их преобразования в формат сообщений соответствующих терминальных устройств. Преобразование информации осуществляется на основе описания структуры информационного фонда и форматов экрана и данных, соответствующих определенному типу запроса.

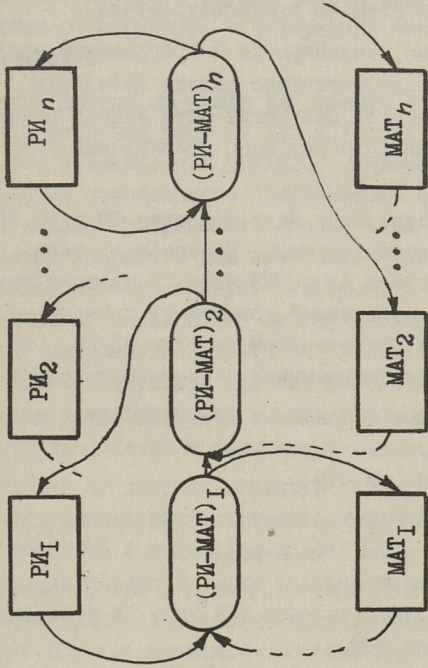
Программы средств удаленного доступа составляются по модульной структуре и используют стандартные средства ОС/ЕС для алфавитно-цифровых дисплеев, а также средства телекоммуникационных методов доступа.

Средства вывода графической информации представляют собой стандартные модули программ, использующие стандарт-



данные
режущих инструментов

данные
материалов реж. части



Фиг. 2. Описание п.п.п соотношений.

ные средства машинной графики ОС-ЕС и предназначенные для вывода эскизов и схем на графопостроители.

Как известно, технические и программные средства мини-ЭВМ типа СМ позволяют успешно решать целый ряд подзадач САПР-ТП. Кроме этого, они могут быть использованы для подготовки и создания программного обеспечения САПР-ТП на ЕС ЭВМ.

Также известно, что мини-ЭВМ типа СМ и ЕС ЭВМ являются программно несовместимыми. Однако в составе дисковой операционной системы мини-ЭВМ типа СМ имеется транслятор с языка ФОРТРАН, реализующий логический уровень соответствующего транслятора операционной системы ОС/ЕС. Это допускает осуществить обмен программами на уровне исходных модулей.

Обмен данными и текстами программы между мини-ЭВМ и ЕС ЭВМ можно осуществить через перфоносители, магнитные ленты и непосредственно от ЭВМ.

В настоящее время разработаны программы передачи текстов программ и данных на перфолентах с ЕС ЭВМ в мини-ЭВМ. Также разработаны программы ввода и преобразования текстов программ на перфолентах мини-ЭВМ типа СМ в библиотеки исходных модулей ЕС ЭВМ.

2. Специализированное программное обеспечение. Это программное обеспечение разрабатывается для решения задач, связанных с функционированием конкретных подсистем САПР-ТП (например, подсистемы проектирования технологических процессов, подсистемы автоматизации подготовки управляющих программ для станков с ЧПУ и т.д.).

При разработке специализированного программного обеспечения САПР-ТП придерживаются также принципов модульного и структурного проектирования и конструирования программ, при котором обязательно выделяются модули, используемые в нескольких подсистемах САПР-ТП (например, программы распечатки конкретных технологических документов, расчета норм времени и т.д.), разрабатываемые для совместного использования на мини-ЭВМ типа СМ и ЕС ЭВМ и т.д.

Первой частью специализированного программного обеспечения, представленной на фиг. 1, являются средства обслуживания фондами информации, которые включают программы за-

грузки и изменения соответствующих структур данных в банке данных САПР-ТП. Состав и логика работы каждой из этих программ зависит от структуры и содержания соответствующего фонда информации.

Передача данных из фонда информации соответствующей подсистемы в мини-ЭВМ типа СМ или АСУ другого уровня или типа, требует создания для каждого формата и содержания выходных данных специального программного обеспечения. При разработке такого программного обеспечения для мини-ЭВМ предполагается сохранить для нее логическую структуру данных. Для решения этого следует разработать систему манипулирования данными, являющуюся подмножеством системы управления фондом информации САПР-ТП на ЕС ЭВМ. В результате этого логические структуры данных, используемые на мини-ЭВМ, являются подструктурами данных информационного фонда на ЕС ЭВМ.

Совмещение логических структур данных ведет к более эффективному использованию возможностей мини-ЭВМ типа СМ при проектировании и решении подзадач САПР-ТП.

Программы выбора и преобразования данных из информационных фондов САПР-ТП предназначены для передачи информации в другие АСУ или системы обработки информации. Эти программы осуществляют формирование файлов, которые могут быть обработаны обычными средствами управления данными операционной системы ОС/ЕС.

Функциональные подсистемы управляются мониторной системой ИСП-2 и предназначены для решения разнообразных задач проектирования технологических процессов. Интеграция и дифференциация этих задач на подсистемы во многом зависит от конфигурации конкретной вычислительной установки, постановки задачи и т.д.

На кафедре технологии машиностроения разрабатывается САПР-ТП по изложенной схеме. Принятая схема построения САПР-ТП является открытой как в базовой, так и в специализированной части и строится целиком с соблюдением принципов модульности и эволюционности.

Л и т е р а т у р а

1 Мельников И.А., Мартин К.О., Прууден Э.В., Прууден Ю.И., Пукк Р.А., Саар Х.Я., Тамм Б.Г. Метасистема для создания информационно-связанных специализированных систем программирования. - Кибернетика, 1974, № 6, с. 69-79.

2. Прууден Ю.И., Квэтнер Р.А. Монитор системы автоматизации проектирования технологических процессов (требования и реализация). См. наст. сб., с. 19.

J. Paasuke

Software Structure of the Computer Aided Manufacturing Process Planning Systems

Summary

Software structure of the computer aided manufacturing process planning (CAPP) systems is represented for the ES-computer and for the mini-computer type SM in this paper.

The main requirements for the basic and specialized software for CAPP-systems are described.

The interface between the mini-computer is also discussed.

Н.Х. Викторова, О.М. Эльяс

РАЗРАБОТКА БАЗОВОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
ПОДНАСТРАИВАЕМЫХ САПР-ТП

Большое разнообразие и изменчивость условий производства требует определенных динамических свойств и легкой адаптируемости САПР-ТП. Затраты, необходимые для ввода изменений и запуска системы в значительной мере влияют на эффективность системы проектирования.

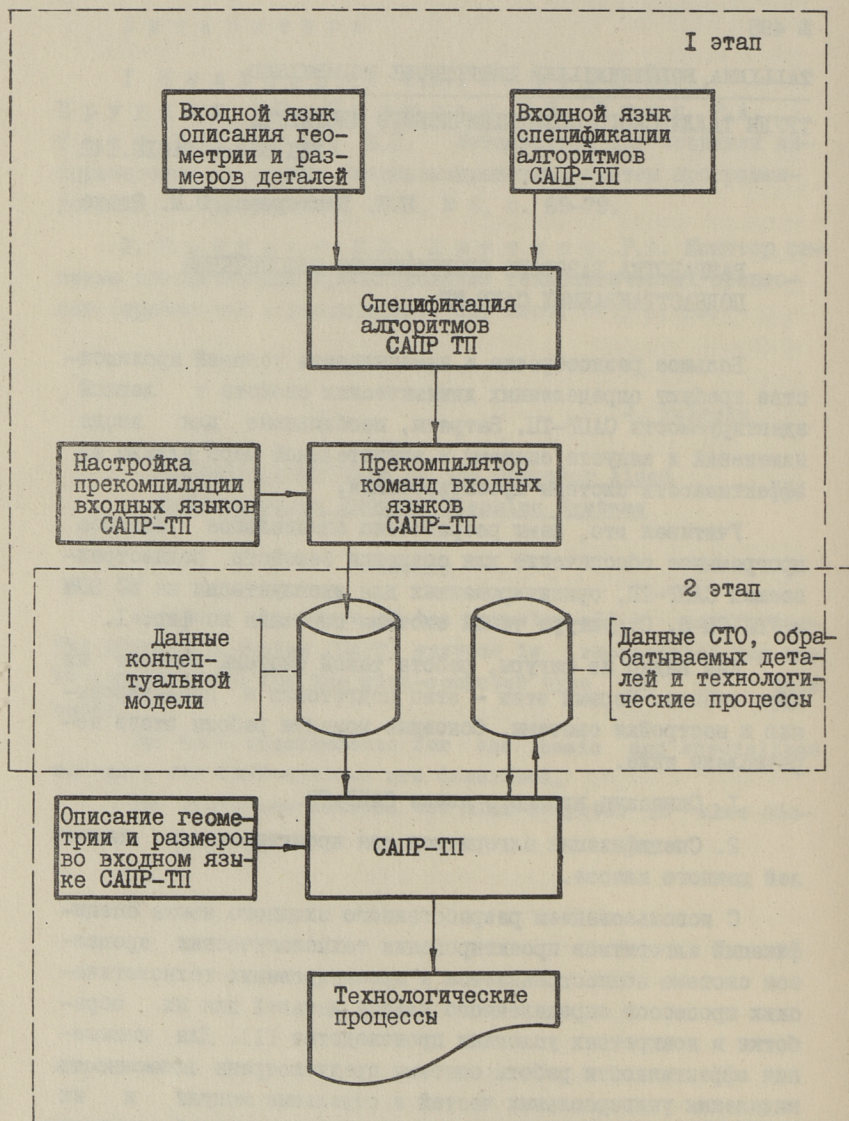
Учитывая это, нами разработано специальное базовое программное обеспечение для создания семейств поднастраиваемых САПР-ТП, ориентированных для эксплуатации на ЕС ЭВМ и УБК СМ-3. Структура такой системы показана на фиг. 1.

Как видно из фигуры, работа такой системы состоит из двух этапов. Первый этап — этап подготовки к проектированию и настройки системы. Основные моменты работы этапа перечислены ниже.

1. Описание входного языка САПР-ТП.

2. Спецификация алгоритмов для проектирования деталей данного класса.

С использованием разработанного входного языка спецификаций алгоритмов проектирования технологических процессов система поднастраивается к проектированию технологических процессов определенного класса деталей для их обработки в конкретных условиях производства [1]. Для повышения эффективности работы системы предусмотрена возможность выделения универсальных частей в отдельные модули и их многократного использования. Связь между системой и универсальными модулями устанавливается при спецификации алгоритмов, используя средства входного языка. Структура входного языка спецификации алгоритмов приведена на фиг. 2.



Фиг. 1. Структура поднастраиваемой САПР-ТП.

Например, спецификация записи оборудования показана ниже:

'ЗАП1' 'ОП' '1032'

'ЗАП2' 'ОБ1' '16K20Ф3' 'ОП' 'ЕСЛИ' 'D.OБ < = L.OБ &
D.OБ < = 200. ! D.OБ > L.OБ' 'ПР' 'PRIOR'

'ЗАП3' 'ОБ2' '1Б732Ф3' 'ОП' 'ЕСЛИ' 'D.OБ < = L.OБ!
D.OБ > L.OБ & D.OБ ≤ 320.' 'ПР' 'PRIOR'.

СТРУКТУРА ВХОДНОГО ЯЗЫКА СПЕЦИФИКАЦИИ АЛГОРИТМОВ

$$\begin{array}{l} \text{ЗАПИСЬ_НОМ} _ \text{'элемент'} \left[_ \text{'вариант'} \left[_ \text{'n1'} \left[_ \text{'n2'} \left[_ \text{'n3'} \left[_ \text{'n4'} \right] \right] \right] \right] \right] \\ \left[\text{ЭХ} _ \text{'эл-вх'} \left[_ \text{'вариант'} \right] \left[_ \text{'ЕСЛИ} _ \text{'выраж 1'} \left[_ \text{'ПР} _ \text{'выраж 2'} \right. \right. \\ \left. \left. _ \text{'ИСК} _ \text{'выраж 1'} \left[_ \text{'ВЫЧ} _ \text{'перем'} _ \text{'=} _ \text{'выраж 2'} \right] \right] \right] \end{array}$$

где: ном - порядковый номер команды;
'элемент' - наименование элемента модели;
вариант - порядковый номер альтернативного варианта элемента;
n1, n2, n3, n4 - значения параметров элемента;
'эл-вх' - наименование элемента, предка к данному элементу;
'выраж 1' - логическое выражение условия допустимости;
'выраж 2' - арифметическое выражение расчета приоритета
или параметров.

Фиг. 2. Структура входного языка спецификации алгоритмов.

3. Компиляция команд входного языка спецификации алгоритмов.

Для повышения эффективности САПР-ТМ предусматривается предварительная компиляция команд входного языка спецификации алгоритмов. Сканирование записей входного языка осуществляется средствами ИСП-2 [1]. Для обработки логических и арифметических выражений разработаны специальные программные средства - прекомпилятор выражений. Работа прекомпилятора выражений основана на принципе сканирования, описанного в [2]. Этап сканирования разбивает выражение на элементы, после чего осуществляется их синтаксический анализ. В случае обнаружения каких-либо ошибок выдается соответствующая диагностическая печать.

При компилировании выражений использован известный принцип преобразования выражения к виду инверсной польской записи. Параллельно формированию польской записи происходит процесс заполнения таблицы переменных (ТАВ) элементами компилируемого выражения.

Выражения могут содержать:

- а) идентификаторы;
 - составные, т.е. идентификаторы с признаком принадлежности к конкретному массиву данных, например, TYPE.ОБ, где: TYPE — идентификатор;
 - ОБ — признак принадлежности к массиву данных обоработования.
 - несоставные, т.е. без признака принадлежности (рабочие);
- б) литеральные константы, например, 'Т15К8', 'К62', 'СТАНОК';
- в) вещественные и целые числа со знаком или без;
- г) стандартные функции или подпрограммы, например, SIN, COS...
- д) любые написанные подпрограммы, используемые для нужд систем автоматизированного проектирования.

Для улучшения читаемости и удобства записи элементы выражения допускается отделять друг от друга любым количеством пробелов.

Требования к синтаксису и список допустимых операций аналогичен общеизвестным алгоритмическим языкам (ФОРТРАН, ПЛ/I).

Для удобства использования данных выражений предусмотрена возможность идентификации их именами, т.е. если какое-то выражение имеет вид:

$$IDT.PI = 100 + TYPE.PI/10. \ \& \ IDT.PI = 200 + TYPE.PI/20, \ (I)$$

то можно правой и левой частям присвоить имена А1 и А2 соответственно, т.е.

$$A1 \sqsubset IDT.PI = 100 + TYPE.PI/10. \ , \ (2)$$

$$A2 \sqsubset IDT.PI = 200 + TYPE.PI/20. \ (3)$$

Используя (2) и (3) запись (I) примет более простой вид: А1 & А2.

В системе предусмотрены следующие возможности диагностики ошибок:

- контроль установленных допустимых длин элементов;
- контроль занятости таблицы переменных и польской записи, с целью предотвращения переполнения служебных массивов;

- проверка принадлежности элементов к спискам допустимых элементов системы (специфицируются при настройке);
- контроль полученной польской записи на способность быть свернутым интерпретатором.

При обнаружении ошибки компиляции ошибочного выражения прекращается. При работе компилятора используется BLOCK DATA, которая допускает производить настройку компилятора, т.е. позволяет изменять длины и структуры информационных массивов без внесения изменений в программную часть.

Приведенный ниже пример иллюстрирует основные этапы работы компилятора.

Исходное выражение:

$IDT.PI = 200 + 10.0 \& MAT.PI = 'T15K6'$

Форма представления переменных приведена в табл. I.

Т а б л и ц а I

№	Код элемента	Признак принадлежности к массиву исходных данных	Идентификатор переменной	Адрес значения идентификатора в массиве данных	Тип переменной или количество литер
1	D	PI	IDT	5	I
2	CI	200			
3	CF	10.0			
4	D	PI	MAT	1	6
5	CA	T15K6			

Польская запись:

$I \ 2 \ 3 \ - \ I \ - \ 7 \ 4 \ 5 \ - \ 7 \ - \ I2,$

- где
- I - операция сложения;
 - 7 - логическое равенство;
 - I2 - логическое И;

I, 2, 3, 4, 5 - номера строк таблицы переменных, используемые для выборки элементов;

D - код в случае, если элемент является идентификатором;

CI - код в случае, если элемент - целое число;

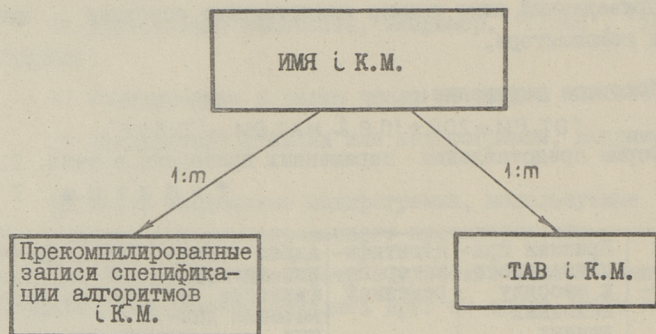
CF - код в случае, если элемент - вещественное число;

CA - код в случае, если элемент - литеральная константа;

F - код в случае, если значение элемента вещественного типа;

- I — код в случае, если значение элемента целого типа;
 PI — признак массива значений элементов режущего инстру-
 мента.

В результате работы первого этапа получается концептуальная модель (К.М.) проектирования технологических процессов деталей данного класса, которая записывается в банк данных (БД). По мере появления деталей иных классов БД пополняется их концептуальными моделями, т.е. БД обладает свойством накопления.



Фиг. 3. Структура концептуальной модели.

Структура i концептуальной модели представлена на фиг. 3.

Второй этап — этап проектирования. Полученная и записанная на первом этапе в БД К.М. позволяет производить многократное проектирование технологических процессов деталей данного класса. Проектирование рабочих технологических процессов состоит из описания геометрии и размеров конкретных деталей и из использования универсального интерпретатора проектирующего с использованием данных К.М. технологические процессы обработки описанных деталей.

Разработанные средства перекрывают значительный диапазон проектирования технологических процессов.

Л и т е р а т у р а

I. K y t t n e r, R., S h t c h e g l o v, N., K i m m e l, A. A complex computer-aided process planning and optimization for machine production. — "Prolomat-79", USA, Michigan, Technical paper MS79-165.

2. Г р и с Д. Конструирование компиляторов для цифровых вычислительных машин. М., Мир, 1975. 544 с.

N. Viktorova, O. Elyas

The Basic Software for Adaptive
Computer-aided Process Planning Systems

Summary

This article deals with the problems of software elaboration for adaptive computer-aided process planning systems. The main stages of the system functions are described. The structure of the system is given. The special language for describing the technological conceptual models and the special compiler for this language are considered.

МОНИТОР СИСТЕМЫ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ
ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ (ТРЕБОВАНИЯ
И РЕАЛИЗАЦИЯ)

Основными функциями монитора системы автоматизации проектирования технологических процессов (САПР-ТП) являются:

- 1) генерация входных языков проектирования САПР-ТП;
- 2) компоновка подсистем и блоков САПР-ТП на базе исходных программных модулей;
- 3) организация передачи данных между различными составными частями САПР-ТП и между различными системами;
- 4) управление процессами решения задач на базе интерпретации команд входного языка.

Типовая структура системы автоматизации проектирования представлена на фиг. 1.

В статье рассматриваются вопросы применения разработанной в Институте кибернетики АН ЭССР интегрированной системы программирования ИСП-2 [1], [2] в качестве монитора САПР-ТП.

Состав ИСП-2

ИСП-2 состоит из трех основных частей:

- 1) совокупности подсистем;
- 2) системной части ИСП-2;
- 3) системы программирования с процедурной ориентацией (ИСПФОР).

Каждая отдельная реализация САПР-ТП является подсистемой некоторой вышестоящей интегрированной системы подготовки производства.

Все подсистемы ИСП-2 должны иметь одинаковую структуру и каждая из них состоит из:

- входного языка с проблемной ориентацией (или нескольких таких языков);

- средств реализации входных языков подсистем.

Входной язык каждой подсистемы является специализированным языком с расширенной структурой, который применяется для описания геометрии и размеров обрабатываемых деталей И/ИЛИ для описания элементов технологического процесса.

Средством реализации входного языка каждой подсистемы являются специальная управляющая информация для системной части ИСП-2 и множество семантических программ, реализующих отдельные базисные операции - элементарные части проектных процедур - при выполнении команд входного языка САПР-ТП. С целью повышения эффективности работы САПР-ТП программы отдельных проектных процедур, имеющих близкое функциональное назначение и выполняемых последовательно, могут быть объединены в блоки САПР-ТП (фиг. 1).

Особое значение при реализации САПР-ТП имеют две специальные вспомогательные подсистемы ИСП-2:

- подсистема генерирования команд (ШК-2);
- подсистема определения структур данных (ПОСД-2).

Эти подсистемы применяются разработчиками подсистемы при создании, расширении или изменении САПР-ТП.

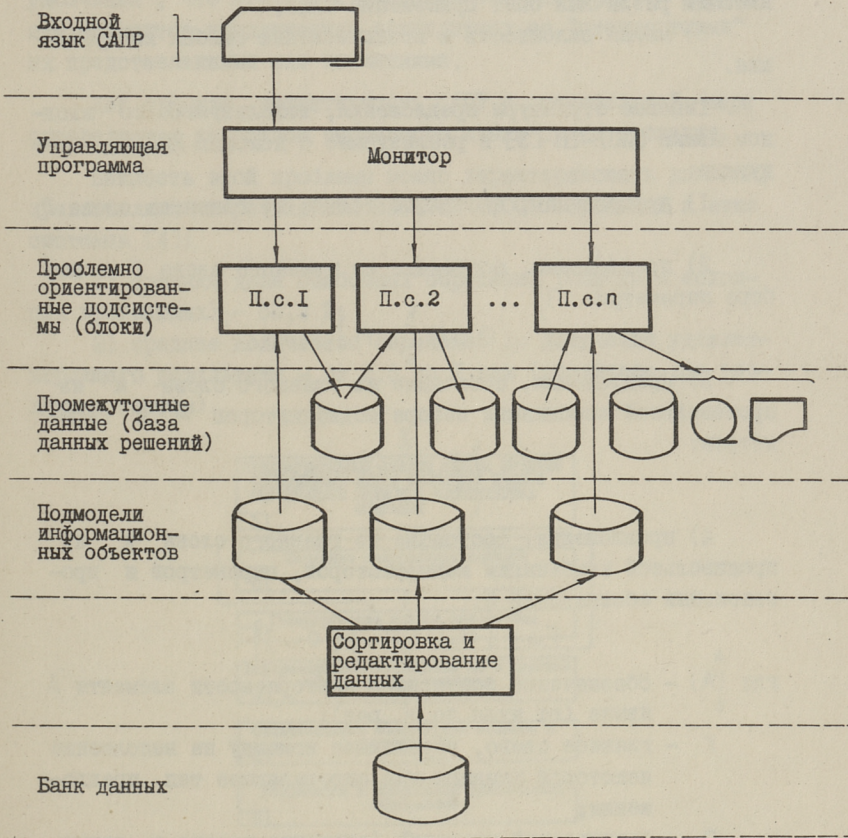
ШК-2 предназначена для генерирования входных языков САПР-ТП.

ПОСД-2 предназначена для формирования таблиц описания структур данных для тех САПР-ТП, которые могут быть совместно использованы при решении комплексных проблем интегрированных систем проектирования.

Системная часть ИСП-2 предназначена для управления работой как системы в целом, так и отдельных ее подсистем различных САПР-ТП.

Система программирования с процедурной ориентацией (ИСПФОР) предназначена для создания подсистемы программ. Язык ИСПФОР является расширением языка ФОРТРАН IV.

Более подробное описание средств ИСП-2 можно найти в [1], [2].



Фиг. 1. Архитектура САИР-ТП.

Генерация входных языков САИР-ТП

К входным языкам САИР-ТП предъявляются следующие основные требования:

- 1) простота кодирования исходных понятий, применение несложного синтаксиса и мнемонических кодов;
- 2) хорошая читаемость языка, возможность структурирования программ, написанных на входном языке;

- 3) малая трудоемкость описания на входном языке объектов проектирования и проектных процедур;
- 4) адаптируемость к различным группам пользователей, имеющим различный опыт применения САПР;
- 5) малая склонность к возникновению ошибок кодирования.

Типовые структуры предложения, используемые во входном языке САПР-ТП [3] и реализуемые с помощью ИСП-2, следующие:

- 1) предложения, состоящие только из главного слова: Г;
- 2) предложения, состоящие из главного слова Г и набора параметров П:

$$Г \quad П \quad \begin{bmatrix} \text{ } \\ \text{ } \\ 0 \end{bmatrix} \text{ П };$$

- 3) предложения, состоящие из главного слова и из произвольной комбинации набора модификаторов М и параметров:

$$Г \quad \left\{ \begin{bmatrix} \text{ } \\ 0 \end{bmatrix} \begin{bmatrix} k \\ \text{ } \\ \text{ } \end{bmatrix} \text{ М } \begin{bmatrix} \text{ } \\ \text{ } \\ 0 \end{bmatrix} \begin{bmatrix} n \\ \text{ } \\ \text{ } \end{bmatrix} \text{ П } \begin{bmatrix} \text{ } \\ \text{ } \\ \text{ } \end{bmatrix} \right\};$$

- 4) предложения, состоящие из главного слова и из произвольной комбинации модификаторов, параметров и произвольных обозначений О:

$$Г \quad \left\{ \begin{bmatrix} k \\ \text{ } \\ 0 \end{bmatrix} \begin{bmatrix} \text{ } \\ \text{ } \\ \text{ } \end{bmatrix} \text{ М } \begin{bmatrix} \text{ } \\ \text{ } \\ 0 \end{bmatrix} \begin{bmatrix} m \\ \text{ } \\ \text{ } \end{bmatrix} \text{ П } \begin{bmatrix} \text{ } \\ \text{ } \\ 0 \end{bmatrix} \begin{bmatrix} n \\ \text{ } \\ \text{ } \end{bmatrix} \text{ О } \begin{bmatrix} \text{ } \\ \text{ } \\ \text{ } \end{bmatrix} \right\};$$

где (А)₀^к — обозначение допустимой повторяемости элемента А языка (от нуля до к раз);

Г — главное слово, выражающее команду на выполнение некоторой функции или определяющее тип предложения;

П — фактический параметр (значение вместе с наименованием или без наименования параметра);

М — модификатор;

О — произвольное обозначение.

В описаниях объектов проектирования и проектных процедур на входных языках САПР-ТП может перед любым предложением Пр стоять маркер (метка) в виде произвольного обозначения (числа или слова):

$$0) \text{ Пр.}$$

Это позволяет избежать дублирования повторяющихся групп предложений и переходить к помеченным предложениям по командам перехода.

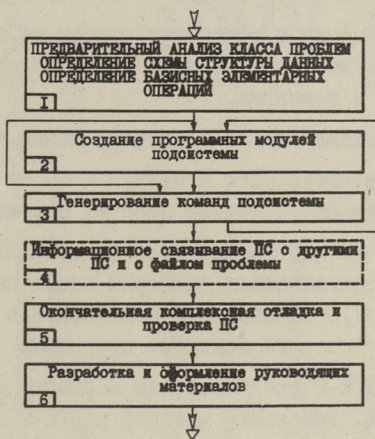
Для повышения читаемости в предложения могут быть включены игнорируемые слова – пояснения. С другой стороны, при записи предложений можно широко использовать "принцип умолчания", что позволяет укорачивать запись путем опускания элементов предложения, совпадающих со "стандартными" их представлениями или значениями.

Как правило, эффективность САПР в значительной мере определяется качеством выбранного языка проектирования.

Важность этой проблемы можно характеризовать данными, представляющими результаты анализа 40000 обращений к этим системам [4]:

1) средняя доля ошибочных обращений (ошибки в исходных программах) – 56,3 %;

2) среднее количество обращений до получения удовлетворяющего результата – от 2,5 до 7 раз (различное для различных систем).



Фиг. 2. Последовательность действий при генерации входных языков САПР-ТП.

Последовательность действия при генерации входных языков и при создании САПР-ТП с использованием ИСП-2 представлена на фиг. 2.

В подсистеме ППК-2 имеются средства для описания состава и структуры команд входного языка, а также средства описания информационных связей между командами входного

языка и семантическими программами, реализующими процесс проектирования технологических процессов.

В качестве примера использования входного языка ШК-2 на фиг. 3 приведено описание команды "ПРОДТ" одной подсистемы САПР-ТП. Отметим, что приведенное на этой фигуре описание определит одновременно не только формат команды и применяемые семантические программы, но и диагностические реакции монитора в случае ошибочной записи команды и стандартные значения в случае использования "принципа умолчания".

В зависимости от необходимости разработчик САПР-ТП сможет предусмотреть следующие специальные мероприятия по диагностике и обработке ошибок в исходных программах на входных языках подсистем [1], [2]:

- диагностику и обработку синтаксических ошибок;
- проверку правильности последовательности команд;
- логическую проверку отдельных параметров команд;
- обработку ошибок, обнаруженных в программных модулях.

Кроме диагностики и обработки ошибок необходимо предусмотреть мероприятия по уменьшению вероятности возможного появления ошибок в командах.

При использовании ИСП-2 последнее можно достигать:

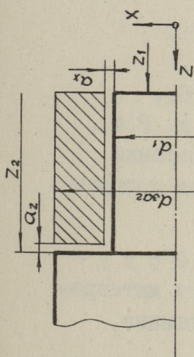
- улучшением читаемости команд;
- применением относительно простых структур команд.

При этом, по нашему мнению, необходимо различать "читаемость" опытного пользователя, который пишет несколько сот команд каждый день, и начинающего пользователя или студента, который пишет свои первые исходные программы.

Средствами ШК-2 предусмотрены варианты для обоих видов пользователей, для этого введены в ШК-2 понятия "полное командное слово" и "ключевое слово команды".

Последнее является также одним средством значительно уменьшения вероятности ошибок при написании команд.

Допустим, что имеем следующее командное слово CIRCLE и ключевое слово команды CR. В данном случае типичные ошибочно написанные слова, как, например, CIRCLE, CIRC1E, CIRCLEL, CIRKLE и т.д., ошибки в которых обычно трудно обнаруживаются, интерпретируются системой правильно.



Описание формата команды ПРОДОЛЬНОЕ ТОЧЕНИЕ

ПРОДТ } ЧЕРН α_z z₁ z₂ α_z } [ПРИТ α_x α_z] [γ γ₁ t]
 ЧИСТ α_z z₁ z₂

ДОБ α_z КД } ПРОДТ } ПРОДОЛЬНОЕ ТОЧЕНИЕ'
 МОД НАЧАЛО } ЧЕРН

ПЕРЕМ } ВЕЩ } КОЭФ/КС } 4 } ЛОК } РА } ГРЕБ

МОД } ЧИСТ

ПЕРЕМ } ВЕЩ } КОЭФ/КС } 2 } ЛОК } РА } ГРЕБ

ИНАЧЕ

ОШИБКА } ЛОК } ИДИ

ИЗВ } НЕПРАВИЛЬНО ЗАДАНА КОМАНДА } ПРОДТ'

КОНЕЦ } БЛОКА } МОД

ПЕРЕМ } ИДИНТ } ПРИТ } ВЕЩ } КОЭФ/КС } 2 } ЛОК } РА } СТАНЦ } 0.5

ПЕРЕМ } ИДИНТ } ПР } ВЕЩ } ЛОК } Т } СТАНЦ } 1 } 0

ИДИ } РТР

КОНЕЦ } КД

- добавление новой команды ПРОДТ
- модификация - черновое точение
- описание параметров для чернового точения
- модификация - чистовое точение
- описание параметров для чистового точения
- ошибка при задании модификации команды

- описание параметров припуска, стандартное значение 0.5
- описание параметра, задающего глубину резания
- вызов семантической программы РТР

Фиг. 3. Пример использования ПКК-2 для спецификации команд входного языка САПР-ПТ.

Компоновка САПР-ТП из отдельных модулей осуществляется специальной системной программой ИСП-2. Передача данных из команд входного языка программным модулям подсистемы (семантическим программам) осуществляется через системное общее поле данных (соответствующее непомеченной области COMMON в языках ФОРТРАН и ИСПФОР). Для передачи данных между модулями САПР-ТП доступны все средства, имеющиеся в языке ФОРТРАН.

Эффективность применения ИСП-2 для разработки САПР-ТП определяется в значительной мере наличием универсальных программных модулей проектирования отдельных элементов технологических процессов.

На кафедре технологии машиностроения сгенерированы в настоящее время три системы автоматизации проектирования технологических процессов. Полученный опыт показывает, что ИСП-2 дает в руки разработчиков удобные средства для разработки САПР-ТП.

Л и т е р а т у р а

1. Мельников И.А., Мартин К.О., Пруден Э.В., Пруден Ю.И., Пукк Р.А., Саар Х.Я., Тамм Б.Г. Метасистема для создания информационно-связанных специализированных систем программирования. - Кибернетика, 1974, № 6, с. 69-73.

2. Мартин К.О., Пруден Э.В., Пруден Ю.И., Тамм Б.Г. Реализация общей части интегрированной системы программирования. - Программирование, 1975, № 5, с. 3-12.

3. К о с н а н, D. Fertigungsprozessgestaltung und Informationsverarbeitung. Berlin, VEB Verlag Technik, 1977, S. 266.

4. К о л о с, Ir.J. The influence of the programming language on the productivity and reliability of part programming. - NC/CAM. The new industrial revolution. ASME/CASA, 1977, p. 465-482.

Monitor for the Computer Aided
Manufacturing Process Planning Systems

Summary

An example of using the Integrated Programming System ISP-2 for the generation of the computer aided manufacturing process planning systems (CAPP) is described. The whole CAPP system is proposed as a set of problem-oriented sub-systems (POS). Each POS needs its own problem-oriented language (POL). The process of definition of the POL and the generation of the new POS using the facilities of ISP-2 are described.

ИНДИВИДУАЛЬНЫЕ БАЗЫ ДАННЫХ ДЛЯ ТЕХНОЛОГИЧЕСКОГО ПРОЕКТИРОВАНИЯ

1. Введение

Разработанная для ЭВМ ЕС инструментальная система программирования ПРИЗ позволяет создавать пакеты прикладных программ, снабженные проблемно-ориентированными входными языками высокого уровня. При этом обеспечивается программная совместимость пакетов, являющаяся необходимой предпосылкой создания интегрированной системы решения задач из разных областей инженерного проектирования [1].

Рассматриваемая здесь система управления базами данных (СУБД) DAVU является также пакетом прикладных программ, созданных в системе ПРИЗ ЕС. Система DAVU предназначена для создания и обслуживания индивидуальных баз данных, допускающих сложные запросы и интегрированную обработку данных совместно с другими пакетами программ инженерного проектирования (в том числе технологического проектирования).

Система DAVU применима также для построения небольших баз технологических данных, которые не требуют обеспечения секретности данных (назовем эти базы индивидуальными базами данных).

2. Характеристика пакета программ СУБД

В СУБД DAVU реализована множественная модель базы данных, концепции которой близки к языку SETL. Существенно, что элементом множества может быть объект любого типа, описанный на входном языке системы ПРИЗ-УТОПИСТ [2].

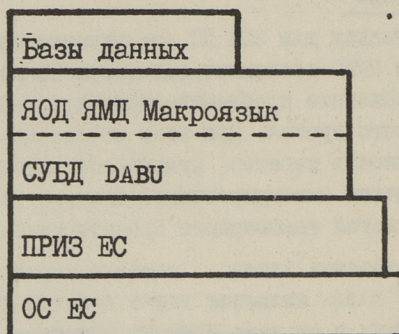
Как в большинстве системах управления базами данных, так и здесь для общения с системой используются два языка

(оба языка имеют диалоговые режимы): язык описания данных ЯОД и язык манипулирования данными ЯМД [3].

Кроме того, система DAVU имеет макроязык запросов, на котором создаются содержательные и краткие запросы, близкие к запросам на естественном языке [4].

Система DAVU может управлять большим числом баз данных, при этом настраивание системы на конкретные базы данных осуществляется указанием в тексте запроса соответствующих имен файлов и имен схем баз данных.

Архитектура системы приведена на фиг. 1.



Фиг. 1.

3. Описание технологических данных

Для описания технологических данных в системе DAVU используется язык описания данных. ЯОД включает все средства описания данных из входного языка системы ПРИЗ-УТОПИСТ, к которым добавлено понятие множества — SET.

Ниже приведен фрагмент описания технологической базы данных (схема базы данных) на ЯОД.

```
LET' SCHEMA1:(MAT:(TYPE:STRING';  
             HB, SIGMA.REAL');  
           DETAIL:(MATERIAL:MAT;  
                  L,D:REAL');
```



```

MACHINETOOL:(CLASS,TYPE:STRING';
              YEAR:INTEGER';
              MAXDIAM,MAXLENGTH:REAL');
MATERIALS:SET OF MAT;
DETAILS: SET OF DETAIL;
MACHINETOOLS:SET OF MACHINETOOL);
INSERT' SCHEMA1;

```

Данный пример содержит описания объектов (записей) материала, детали и станка и их множеств (файлов).

Напомним, что все ключевые слова языка УТОПИСТ отмечаются символом "'".

Каждое описание начинается со слова LET'. В конце описания данная структура вводится директивой INSERT' в библиотеку описаний.

Кроме директивы INSERT' для работы с библиотекой описаний имеются директивы DELETE' и EXEC' HELP.

Директива DELETE' служит для удаления из библиотеки описаний ненужных схем или их частей.

Например, удаление из объекта "станок" компоненты "год выпуска" описывается так:

```
DELETE' SCHEMA1. MACHINETOOL.YEAR;
```

Директивой EXEC' HELP (при работе в диалоговом режиме) выполняется программа-информатор HELP, которая позволяет ознакомиться с библиотекой описаний путем вывода на экран ее содержания.

4. Манипулирование технологическими данными

Все запросы к базе данных строятся на языке манипулирования данными - ЯМД. ЯМД является расширением языка УТОПИСТ понятиями обработки данных. Эти понятия следующие: подмножество, минимум, максимум, сумма элементов, вывод множества и другие. Существенно, что все понятия языка манипулирования данными строятся также из конструкции языка УТОПИСТ и хранятся в библиотеке описаний (MODLIB, см. фиг. 2). Описание и ввод некоторых понятий приведены ниже:

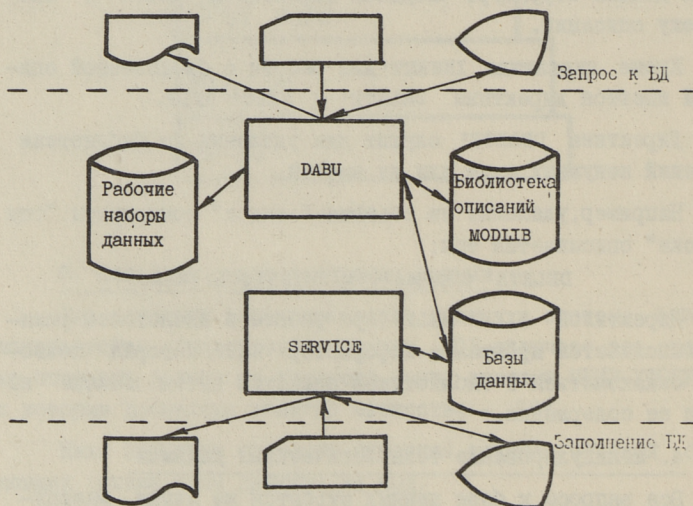
```

LET DBMS:(SET:(OF:UNDEFINED';
           ID:STRING';
           KEY:UNDEFINED';
           NR:INTEGER';
MODULE'SETNEX IN' ID, NR OUT' OF;
MODULE'SETKEY IN' ID, KEY OUT' OF);

SUBSET:(OF, IS: SET;
        COND:LOGICAL';
        N: INTEGER';
MODULE' SUBSEM 2 IN' OF.ID OUT' IS.ID,
        N ARG' OF.NEXT RES'OF.OF,COND);

INSERT DBMS;

```



Фиг. 2.

В этом примере описаны (на языке УТОПИСТ) понятия множества (понятие использовалось уже выше) и подмножества. Запросы на ЯМД к базе данных строятся из копии этих понятий (предложений), указывая каждому предложению свойственные

характеристики-параметры. Ниже следует пример описания запроса к базе данных. Данный запрос состоит в том, что требуется выделить из числа всех станков цеха "12" те станки, на которых можно обрабатывать турбинные валы максимальной длиной. Строки комментариев в тексте запроса начинаются символом "*" .

```
* THE PROBLEM:
* PRINT ALL MACHINETOOLS IN SHOP 'TWELVE'
* WHICH CAN BE USED FOR CUTTING THE
* SHAFTS OF TURBINES.
  PROBLEM' QUEST;
  FROM' DBMS;
* DECLARATIONS
  LET' SHAFT:DETAIL;
  LENGTH:REAL';
  SHAFTS:SET OF SHAFT, SHAFT.L = LENGTH;
  LATHES, OUTSET: MACHINETOOLS;
* MANIPULATIONS
  INPUT1: TAKE SHAFTS, FROM 'TURBINES';
  INPUT2: TAKE LATHES, FROM 'TWELVE';
  MAXIMUM: MAX LENGTH, OF SHAFTS;
  WANTED: LOGICAL';
  ASSIGN' WANTED:= LATHES.MACHINETOOL.MAXLENGTH GE'
  MAXIMUM.RES;

  OUT: SUBSET OF LATHES, IS OUTSET, COND=WANTED;
  'OUTPUT: DISPLAY OUTSET, PICT 'MACHINETOOL';
ACTIONS'
  ON' QUEST COMPUTE OUTPUT. PRINT;
END'; +++
```

Данный текст начинается директивой FROM', которой открываются понятия баз данных из области DBMS файла MODLIB для дальнейшего использования. В следующих двух предложени-

ях определяются понятие "вал", получающее структуру объекта "деталь" и величина "длина". Далее определяются множества "валы" (длина которых нас интересует), множество "токарные станки" и множество "найденные токарные станки".

В следующих двух предложениях открываются соответственно файлы турбинных валов и токарных станков двенадцатого цеха. Затем определяются максимальная длина валов и величина "подходящие", которой присваивается значение условия выбора станков. Теперь определяется, что из множества токарных станков образуется подмножество найденных токарных станков, используя при этом описанное выше условие выбора.

В конце запроса указывается, что требуется выдать множество найденных станков по формату "станок". Запрос заканчивается оператором постановки задачи, который указывает, что по тексту данной задачи следует вычислить множество подходящих станков и выдать его на АПЛ.

5. Запросы на макроязыке

Для повышения естественности синтаксиса языка манипулирования данными и для сокращения текстов запросов разработан макроязык запросов. Этот язык является менее универсальным чем ЯМД, но он более удобен для технологов, имеющих малый опыт программирования [4].

Макроопределения данного языка создаются только для типовых запросов. Все макроопределения хранятся в отдельной макробιβотеке, куда они предварительно записываются программами обслуживания макробιβотеки. Запросы на макроязыке переводятся макрогенератором системы ПРИЗ ЕС (используя макробιβотеку) в текст на ЯМД и обрабатываются дальше средствами этого языка.

Пример запроса на макроязыке приведен ниже:

НАЙТИ ВСЕ ДЕТАЛИ ИЗ МАТЕРИАЛА '40XМ'.

6. Заключение

Описанная выше система управления базами данных DABU предлагает пользователям ряд инструментов для ведения и обслуживания баз данных. Система позволяет хранить на

внешних устройствах ЕС ЭВМ как технологические данные, так и их описания в виде схем баз данных. Для описания и обработки технологических данных пользователь получает диалоговые языки описания данных и манипулирования данными, а также комплекс диалоговых программ для обслуживания баз данных.

Кроме того, созданные в системе DABU базы данных применимы совместно с другими созданными в системе ПРИЗ ЕС пакетами прикладных программ, предлагая им сервис обработки данных.

Л и т е р а т у р а

1. Т н у г у Э. Программы и системы программирования. Вып. I. Система программирования ПРИЗ. Общее описание, ТПИ Таллин, 1977.

2. Л е в и н Д.Я. Система СЕТЛ. Вып. I. Программирование в системе СЕТЛ.АН СССР, ВЦ СО, Новосибирск, 1978.

3. К а л ь я А.П. Языки системы управления базой данных DABU. Пакеты прикладных программ САПР. Материалы Всесоюзн. научно-техн. совещания. Таллин, 1978, с. 40-46.

4. К а л ь я А.П., М а ц к и н М.Б. Интеллектуальный диалог с базами данных. Интерактивные системы. Материалы советско-финского симпозиума. ВЦ АН СССР. М., 1979, с. 124-136.

A. Kalja

The Individual Data Bases for the
Technological Design

Summary

A dialogue with a technological data base, which has high level query language is considered in this paper.

The dialogue is implemented in a data base management system called DABU, which runs on EC-computer.

The examples of the query descriptions and operation of the whole DABU system have been taken into consideration.

К ВОПРОСУ О ГЛУБИНЕ КОДИРОВАНИЯ ФОРМЫ ДЕТАЛИ

Под кодированием понимают обычно процесс обозначения исходного множества объектов или сообщений набором символов заданного алфавита на основе совокупности определенных правил [1].

Так как большинство методов кодирования основывается на предварительной классификации объектов и их свойств, то целесообразно дать и определение классификации согласно [1]. Классификация, или система классификации, это совокупность правил и результатов распределения заданного множества элементов на подмножества (классы) в соответствии с установленными признаками сходства или различия.

Код характеризуется алфавитом, совокупностью правил образования кодовых комбинаций и емкостью. Теоретические аспекты способов и правил кодирования всесторонне рассмотрены в [1, 6]. Однако иногда возникает вопрос о целесообразной глубине кодирования свойств объекта в зависимости от класса задач.

Одной из проблем является кодирование формы детали. Имеется ряд систем классификации деталей, в пределах которых пытаются решить проблему [2, 7, 8]. Из перечисленных систем классификации наиболее полной и обобщающей является общесоюзный классификатор промышленной и сельскохозяйственной продукции (ОКП), классы 40 и 50 [2]. На фиг. 1 представлен принцип классификации. Все детали делятся на тела вращения - класс 40 и тела невращения - класс 50 (Кл). Внутри этих классов происходит иерархическая классификация: класс делится на десять подклассов (ПК), подкласс на десять групп (Гр), группа на десять подгрупп (ПГ), подгруппа на десять видов (В). Таким образом, код детали получа-

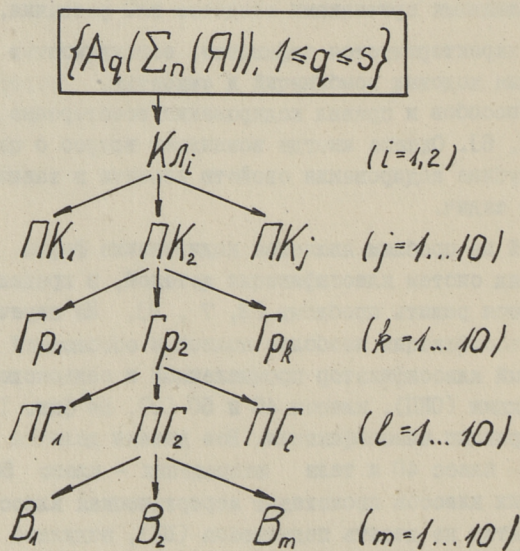
ется в виде набора символов, обозначающих путь слеживания до низшего подмножества. Придерживаясь обозначений на фиг. 1, можно код формы детали обозначить структурной формулой

$$K_l, PK_j, \Gamma_{pk}, \Pi_{\Gamma_l}, B_m. \quad (1)$$

Если мы имеем алфавит Я и совокупность определенных правил Σ_n для представления объекта кодирования в понятном восприятию человека виде, то исходное представление детали может быть выражено в виде $A_q(\Sigma_n(\mathcal{Y}))$. Обозначив буквой К процесс кодирования можем эту процедуру выразить следующим образом:

$$K\{A_q(\Sigma_n(\mathcal{Y}))\} \xrightarrow[q=1,2,\dots,s]{t(K^p\{A_q(\Sigma_n(\mathcal{Y}))\})} \{Y_q^p(D)\}, \quad \bigcup_{t=1}^p 0_t, \quad (2)$$

где $t(K^p\{A_q(\Sigma_n(\mathcal{Y}))\})$ - трудоемкость кодирования заданного множества деталей $\{1,2,\dots,s\}$ на уровне p ;
 $\{Y_q^p(D)\}$ - результат кодирования на уровне p с использованием заданного алфавита D ;
 $\bigcup_{t=1}^p 0_t$ - множество ошибок кодирования на уровне p с учетом ошибок на предыдущих уровнях.



Фиг. 1. Иерархическая система классификаций деталей по ОКП.

Используя выражения (1) и (2) и учитывая, что согласно фиг. 1 $p = 5$ можем весь процесс кодирования формы деталей по ОКП выразить следующим образом:

$$K\{A_q(\Sigma_n(R))\} \xrightarrow[q=1,2,\dots,s]{t(K^1\{A_q(\Sigma_n(R))\})} \{K_{li}\}, O_1 \xrightarrow[i=1,2; j=1\dots 10]{t(K^2\{A_q(\Sigma_n(R))\})} \{K_{li}PK_j\}, \sum_{t=1}^2 O_t \rightarrow (3)$$

$$\xrightarrow[k=1\dots 10]{t(K^3\{A_q(\Sigma_n(R))\})} \{K_{li}PK_j\Gamma_{pk}\}, \sum_{t=1}^3 O_t \xrightarrow[l=1\dots 10]{t(K^4\{A_q(\Sigma_n(R))\})} \{K_{li}PK_j\Gamma_{pk}\Pi_l\}, \sum_{t=1}^4 O_t \rightarrow$$

$$\xrightarrow[m=1\dots 10]{t(K^5\{A_q(\Sigma_n(R))\})} \{K_{li}PK_j\Gamma_{pk}\Pi_l B_m\}, \sum_{t=1}^5 O_t.$$

Кодирование на первом уровне определяет класс детали. На втором уровне определяется функциональная принадлежность: валы, втулки, диски, болты и т.д. На третьем уровне — конструктивную разновидность: валы гладкие, с уступами и т.д. Четвертый и пятый уровни уточняют конструктивные особенности: наличие лысок, канавок, отверстий и т.д.

Таким образом, кодирование формы детали может охватывать любое количество уровней классификации, начиная со старших разрядов с условием, что незаполненные младшие разряды заменяются нулями и код является по сути дела идентификатором формы детали. Увеличение глубины кодирования сопровождается уточнением кода формы детали, а также накоплением ошибок, что вызывает дополнительные затраты на устранение этих ошибок.

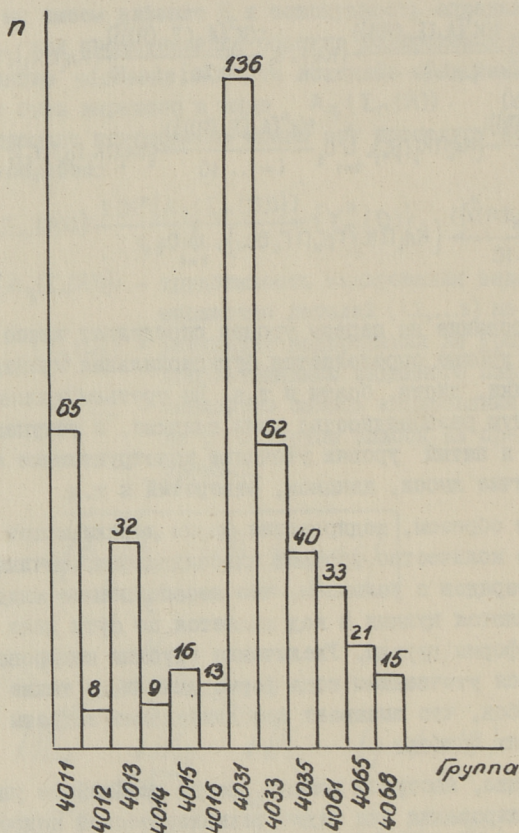
Очевидно, наступит момент, когда дальнейшее увеличение глубины кодирования для задач технологической подготовки производства (ТПП) существенной информации не добавит, а трудоемкость кодирования и устранения ошибок резко увеличивается.

Анализ ряда (3) с учетом класса задач ТПП и результатов практических разработок позволяет высказать следующее.

Во всех задачах ТПП код группы детали используют исключительно для идентификации детали по форме.

Форма детали однозначно определяется тремя уровнями $Y_q^3(D) = K_{li}PK_j\Gamma_{pk}$ при незначительном изменении со-

держания 3 уровня по сравнению с ОКП. Средняя заполненность вида (5 уровень) колеблется от 15 до 80 разновидностей, а это значит, что III и V в (I) часто заменяются нулями, т.е. не несут дополнительной информации.



Фиг. 2. Распределение деталей по отдельным группам. Общее количество деталей класса 40 - 1692. Приведены наиболее объемистые группы.

Получаемая информация на 4 и 5 уровнях незначительная. Например, мы имеем код 40III4, т.е. деталь с наружной поверхностью гладкой, без наружной резьбы (40II00); без центрального отверстия (40IIIO); с пазами, лысками, гранями на наружной поверхности с дополнительными отверстиями (40III4) [2]. Тогда получаемая информация дает возможность группиро-

вать детали в разных целях. Как показывает практика, в условиях серийного производства среднего машиностроения полученные группировки даже на третьем уровне недостаточно велики для решения комплекса задач ТПП, что сопровождается объединением соседних группировок (фиг. 2), а это значит, что информация 4 и 5 уровней кодирования теряется.

Из фиг. 2 видно также, что в основном классификаторе нецелесообразно ввести разделения по наличию наружной резьбы: группы 40I2, 40I4, 40I6 и др. Вместо такого разделения можно внести более четкое определение групп, например, втулок, колец, крышек и т.д. Указанные изменения нами введены и как показывает практика, вполне оправдано.

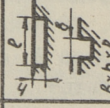

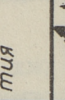
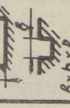


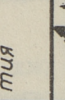
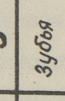
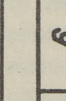
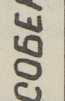
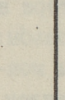
Для кодирования формы детали до 3 уровня необходимо две таблицы и трудоемкость кодирования без вспомогательного времени $t(K^3(A_q(\Sigma_n(N)))) \approx 1$ мин. Для кодирования на 4 и 5 уровнях требуется 240 страниц таблиц для классов 40 и 50, что резко увеличивает трудоемкость кодирования.

Как показывает практика, при кодировании до 3 уровня ошибка составляет 10–15 %^X. При кодировании на 4 и 5 уровнях ошибка может достигнуть до 20–25 %.

В ходе ТПП необходимо решать ряд задач, касающихся унификации металлообрабатывающих инструментов и их размеров, схем компоновок приспособлений и т.д. Для решения подобных задач необходимо кодировать и конструктивные особенности детали с их размерными характеристиками. Здесь мы подразумеваем под конструктивной особенностью (КО) конструктивные элементы детали, которые не обрабатываются во время операции обработки основной поверхности детали.

Нами была предложена система кодирования конструктивных особенностей [4]. КО классифицируются на десять классов формы КФ и десять видов В (фиг. 3). Таким образом, структурная формула кода КО будет $KF_i V_j$. В классификаторе указываются и характерные размеры КО. Кодированию подлежит также класс или степень точности конструктивной особенности.

^X Данные получены при первой классификации всех деталей основного производства на ц/о "Таллэкс". Можно ожидать, что кодирование вновь разрабатываемых деталей сопровождается несколько меньшим количеством ошибок, ввиду отсутствия психологического фактора, наложенного большим количеством деталей.

КЛАССИФИКАТОР КОНСТРУКТИВНЫХ ОСОБЕННОСТЕЙ										
Класс формы	Вид									
	0	1	2	3	4	5	6	7	8	9
		Шпанды канавки	Канавки общего назначен.	Плос- кости	Отвер- тия	Зубья	Шлицы	Резьба внутрен.	Резьба наружн.	
0										
1										
2										

Фиг. 3. Фрагмент классификатора конструктивных особенностей.

Придерживаясь вышеизложенной методики, процесс кодирования множества конструктивных особенностей $\{1, 2, \dots, r\}$ для данной детали представляется в следующем виде:

$$K\{B_x(\Sigma_n(A))\} \xrightarrow[x=1, 2, \dots, r]{t(K\{B_x(\Sigma_n(A))\})} \{K\Phi_i B_j\}, 0_x. \quad (4)$$

Вся информация о конструктивной особенности в кодированном виде будет

$\langle \text{Нр. детали} \rangle \langle \text{Кл}_i \text{ПК}_j \text{Гр}_k \text{П}_l \text{В}_m \rangle \langle \text{КФ}_i \text{В}_j \rangle \langle \text{T} \rangle \langle \text{I.разм} \rangle \langle \text{2.разм} \rangle \langle \text{3.разм} \rangle$

где T — код класса или степени точности.

Размеры записывают в нормальном виде согласно классификатору.

Понятие "конструктивные особенности" использовано и в [5], где оно введено при кодировании данных детали в целях машинного проектирования. На наш взгляд предложенная система кодирования КО лишена обзорности, что затрудняет процесс кодирования и является дополнительным источником ошибок.

В заключение можно сделать выводы:

1. Для однозначной идентификации формы детали достаточно провести кодирование до третьего уровня классификации.
2. В классификаторе ОКП для более точного определения детали необходимо ввести уточнение некоторых групп по форме детали.
3. Для обеспечения информацией большого класса задач ТПП целесообразно кодировать отдельно конструктивные особенности детали.

Л и т е р а т у р а

1. Венчковский Л.Б. Методы кодирования технико-экономической информации. М., Сов.радио, 1978. 120 с.
2. Иллюстрированный определитель деталей общемашиностроительного применения. Руководящий технический материал. Классы 40 и 50 общесоюзного классификатора промышленной и сельскохозяйственной продукции. М., 1972. 339 с.

3. Классификация и кодирование информации. Основные понятия. Термины и определения. ГОСТ 17369-71.

4. Когер Т.А., Папстел Ю.В., Трейер П.И., Щеглов Н.Н. Система информации для технологического анализа машиностроительного производства. Таллин, 1974. 26 с.

5. Митрофанов С.П. и др. Автоматизация технологической подготовки серийного производства. М., Машиностроение, 1974. 356 с.

6. Проскуров В.С. Информация в АСПР. М., Экономика, 1975. 183 с.

7. Шерер Г.А. Классификация деталей. - Стандарты и качество. 1968, № 12, с. 82-85.

8. O p i t z, H. Werkstücksystematik und Teilefamilienfertigung. - Industrie-Anzeiger, 1965, Nr. XII, S. 457-468.

J. Papstel

About the Depth of Coding the Form
of Machine Parts

Summary

In this paper the universal classification system of production and the depth of the machine part's form code are analysed. The two last positions of the six-position code give the minimum extra information and therefore the four-position code is suggested.

A new classification system of the constructive peculiarities is also described.

ПРИМЕНЕНИЕ МНОЖЕСТВА ПРИОРИТЕТОВ ДЛЯ ВЫБОРА МОДЕЛИ СТАНКА

Эффективность обработки деталей на станках определяется совокупностью следующих основных показателей.

1. Производительность.
2. Технологическая себестоимость обработки.
3. Эксплуатационная и технологическая надежность оборудования.
4. Достижимое качество обработки деталей (точность размеров, точность взаимного расположения поверхностей, качество поверхностей).
5. Затраты на подготовку производства.

Детальный расчет представленной совокупности показателей требует полной разработки технологических процессов (структура операции, траектории движения режущих инструментов, режимы обработки, элементы нормы времени), что нельзя считать целесообразным для решения задачи выбора оборудования (распределения обработки деталей между различными станками).

Для оценки исходной совокупности показателей вводятся приоритеты μ_j , $j = 1, 2, \dots, l$. При разработке совокупности приоритетов для выбора оборудования необходимо исходить из следующих предположений.

1. Исходным является описанная совокупность показателей, которая заменяется на новую совокупность, позволяющую с требуемой достоверностью решать поставленную задачу без детальной разработки технологического процесса.

2. Совокупность приоритетов базируется только на геометрических и размерных характеристиках деталей и данных оборудования.

Задачу выбора наиболее целесообразных моделей станков из допустимых альтернативных решений целесообразно решить учитывая взаимодействие всего комплекса описанных показателей.

Такие задачи принятия решения, где одновременно необходимо учитывать многие факторы, называют многокритериальными. Общие положения решения таких задач приведены в литературе, например, [3, 4, 5, 6].

Поставленную задачу решают следующим образом. На основании разработанных функций приоритетов μ_j [8] получают для каждой детали i , $i = 1, 2, \dots, n$ множество оценок $\mu = \{\mu_j(OB_k)\}$ эффективности обработки i -той детали на k -ом ($k = 1, 2, \dots, m$) станке.

В качестве примера в таблице I представлены оценки функции приоритетов [8] для данной детали (фиг. I), на основе которых производится выбор наиболее целесообразной модели станка.

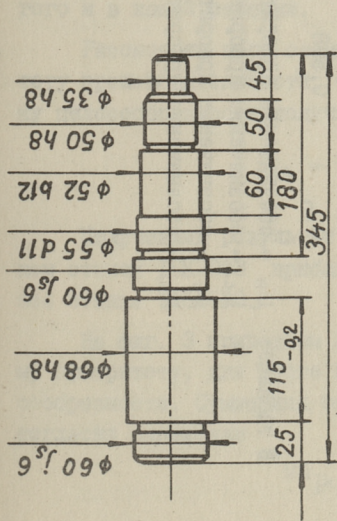
Следующей проблемой является упорядочение полученных оценок. Учитывая характеры разработанных приоритетов, можно полагать, что оборудование ОБ, имеющее оценку приоритета μ (ОБ), предпочтительнее оборудования ОБ' с оценкой μ (ОБ)' по приоритету μ_j при $\mu_j(OB) < \mu_j(OB)'$, $j = 1, 2, \dots, l$. Предполагается, что полученные оценки $\mu_j(OB)$ вполне характеризуют качество оборудования (по одному ее показателю: производительности, точности и т.д.), так что все оборудование, для которого $\mu_j(OB)$ равны, следует считать эквивалентными.

При выполнении данных условий любой разработанный приоритет позволяет построить на исходном множестве оборудования OB_k граф превосходства $G_j(OB, U_j)$ по каждому приоритету. Предположим, что

$$\Theta(OB, OB') \in U_j \Leftrightarrow \mu_j(OB) < \mu_j(OB)',$$

где символ $\Theta(OB, OB')$ используется для обозначения ориентировочной дуги, идущей из оборудования с оценкой приоритета $\mu_j(OB)$ к оборудованию с оценкой $\mu_j(OB)'$, а U_j — множество дуг графа G_j ; при этом ориентированная дуга, имеющая более низкую оценку приоритета, направлена к оборудо-

ОБРАБАТЫВАЕМАЯ ДЕТАЛЬ (№ 719003),



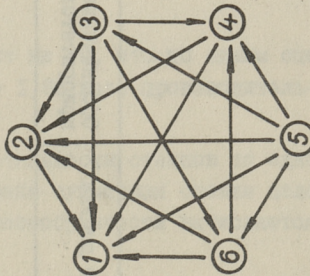
ИСХОДНОЕ МНОЖЕСТВО СТАНКОВ

№	Модель	Технические характеристики	
		DS	DA
1	1A618Ф3	DS = 180	DA = 320
2	1BЭ18Ф3	DS = 180	DA = 320
3	1BК20Ф3	DS = 200	DA = 400
4	1M63Ф3	DS = 350	DA = 630
5	1BК30Ф3	DS = 350	DA = 630
6	1M63Ф306	DS = 300	DA = 630

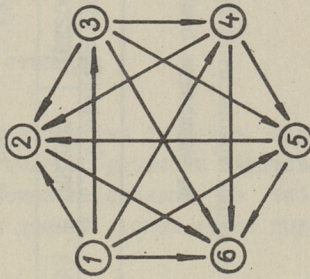
Технические характеристики	
L = 680	L = 680
L = 680	L = 680
L = 930	L = 930
L = 650	L = 650
L = 1200	L = 1200
L = 1400	L = 1400

ОЦЕНКА ЦЕЛЕСОБРАЗНОСТИ ПО ОДНОМУ ПРИОРИТЕТУ

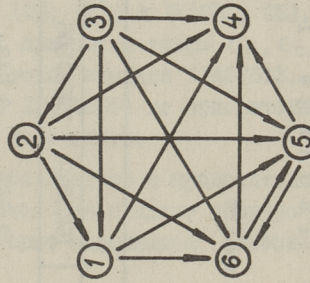
Оценка целесообразности по производительности



Оценка целесообразности по себестоимости



Оценка целесообразности по точности



Фиг. 1. Графические решения выбора наилучшей модели станка с использованием одного приоритета.

Т а б л и ц а I

Оценки функции приоритетов при обработке детали
на множестве станков

№ детали	Обозначение приоритетов	Множество станков					1М63Ф3	1М63Ф3	1М63Ф3	1М63Ф306
		1А616Ф3	1Б616Ф3	1К20Ф3	1К20Ф3	1К20Ф3				
	μ_n	6,368	4,470	3,421	3,710	2,561	3,642			
	μ_c	0,070	0,157	0,102	0,119	0,146	0,179			
719.003	μ_t	0,774	0,724	0,694	0,893	0,844	0,844			
	μ_{nn}	0,598	0,598	0,598	0,598	0,598	0,598			

В таблице: μ_n - обозначение приоритета производительности;

μ_c - обозначение приоритета себестоимости;

μ_t - обозначение приоритета точности;

μ_{nn} - обозначение приоритета, учитывающего затраты на подготовку производства.

ванию с более высокой оценкой, а равенство оценок приоритетов для двух оборудований влечет за собой наличие двух противоположно направленных дуг θ (ОБ, ОБ') и θ (ОБ', ОБ).

В качестве примера на фиг. 1 приведены графы G_n, G_c, G_T упорядочения исходного множества станков (IA6I6Ф3, I6B16Ф3, I6K20Ф3, IM63Ф3, I6K30Ф3, IM63Ф306) по приоритетам производительности, себестоимости и точности.

Для получения сравнительных результатов о предпочтении того или другого станка определяются для каждого оборудования оценки целесообразности, которые выражаются относительными величинами в виде

$$S_{jk_1k_2} = \frac{\mu_j(ОБ_{k_2})}{\mu_j(ОБ_{k_1})}, \quad \text{при } \mu_j(ОБ_{k_2}) > \mu_j(ОБ_{k_1}),$$

в противном случае, если $\mu_j(ОБ_{k_2}) \leq \mu_j(ОБ_{k_1})$,

$$\text{то } S_{jk_1k_2} = 0,$$

k_1, k_2 - индексы сравнительной пары станков;

$$k_1, k_2 = 1, 2, \dots, m.$$

Примером на фиг. 2 приведена распечатка оценок в виде таблицы ($m \cdot l \times m$) для анализа целесообразности изготовления детали № 719003 на данном множестве станков. Из таблицы видно, по скольким приоритетам данный станок лучше другого и в какой степени.

Рассмотрим пару станков IA6I6Ф3 и I6K20Ф3. По приоритету производительности, согласно данным в таблице I, оценку целесообразности получаем

$$S_{из} = \frac{6,368}{3,421} = 1,86.$$

Полученный результат указывает на то, что по нашим оценкам станок I6K20Ф3 приблизительно 1,86 раза производительнее станка IA6I6Ф3.

На фиг. 3 приведены последовательности станков по одному приоритету, для этого использована суммарная оценка целесообразности. Суммарная оценка целесообразности вычисляется согласно выражению

$$VS_{jk_1} = \sum_{k_2=1}^m S_{jk_1k_2}.$$

ДЕТАЛЬ** 719003

	1A61603	16Б1603	16K2003C5	1M6303	16K3003	1M630306
1A61603						
ПР	99.000	0.0	0.0	0.0	0.0	0.0
СЕ	99.000	2.243	1.457	1.700	2.086	2.557
ТО	99.000	0.0	0.0	1.154	1.090	1.090
16Б1603						
ПР	1.343	99.000	0.0	0.0	0.0	0.0
СЕ	0.0	99.000	0.0	0.0	0.0	1.140
ТО	1.069	99.000	0.0	1.253	1.106	1.106
16K2003C5						
ПР	1.061	1.386	99.000	1.084	0.0	1.065
СЕ	0.0	1.539	99.000	1.167	1.431	1.750
ТО	1.115	1.043	99.000	1.287	1.216	1.216
1M6303						
ПР	1.716	1.270	0.0	99.000	0.0	0.0
СЕ	0.0	1.319	0.0	99.000	1.227	1.504
ТО	0.0	0.0	0.0	99.000	0.0	0.0
16K3003						
ПР	2.407	1.051	1.336	1.449	99.000	1.422
СЕ	0.0	1.075	0.0	0.0	99.000	1.226
ТО	0.0	0.0	0.0	1.050	99.000	0.0
1M630306						
ПР	1.740	1.301	0.0	1.019	0.0	99.000
СЕ	0.0	0.0	0.0	0.0	0.0	99.000
ТО	0.0	0.0	0.0	1.050	0.0	99.000

Фиг. 2. Оценки целесообразности использования разных моделей станков для обработки данной детали при наличии совокупности приоритетов.

ПР - приоритет производительности, СЕ - приоритет себестоимости, ТО - приоритет точности.

Как показывает данный пример, при использовании одного приоритета можно получить неоднозначные результаты, т.е. по разному признаку разные станки являются наиболее эффективными.

Рассмотрим две произвольные оценки $\mu(ОБ)$, $\mu(ОБ)' \in \mu$. Применительно к рассматриваемой паре станков $ОБ$, $ОБ'$ множество имеющихся приоритетов может быть разбито на два "противоположных класса". К первому классу $C(\mu(ОБ), \mu(ОБ)')$ отнесем все те приоритеты μ_j , для которых $\mu_j(ОБ) < \mu_j(ОБ)'$, т.е. приоритеты, согласно которым в графе G_j должна иметь место дуга $\theta(ОБ, ОВ')$, ведущая из $ОБ$ в $ОБ'$,

$$C(\mu(ОБ), \mu(ОБ)') = \{\mu_j | \theta(ОБ, ОВ') \in U_j\}.$$

Остальные приоритеты, согласно которым, напротив, имеет место отношение $\theta(ОБ', ОБ)$, образуют второй класс $D(\mu(ОБ), \mu(ОБ)')$

$$D(\mu(ОБ), \mu(ОБ)') = \{\mu_j | \theta(ОБ, ОВ') \notin U_j\}.$$

Прежде чем приступить к окончательному решению задачи выбора станка, необходимо ввести еще показатель относительно сравнительной важности различных приоритетов. Существует много методик для определения коэффициентов важности приоритетов в задачах принятия решения [1, 2, 7].

В случае выбора наилучшей модели станка предпочтение того или другого признака зависит от конкретных условий производства. Исходя из этого, только человек, приняв решения, исходящие из конкретных требований, может дать качественную оценку отношения превосходства, равенства или меньшей предпочтительности между рассматриваемыми приоритетами μ_j .

Допустим, что меру относительной важности приоритетов можно выразить с помощью коэффициентов C_j , $j = 1, 2, \dots, l$, которые могут быть любыми действительными положительными числами. Сумму всех C_j обозначим через C . В случаях, где в программе выпуска деталей все приоритеты играют одинаковую роль, можно принять $C_j = 1$, при этом $C = l$.

Другие конкретные условия производства можно описать

ДЕТАЛЬ** 719003 ПРИОРИТЕТ**ПР

ПОСЛЕДОВАТЕЛЬНОСТЬ СТАНКОВ

ПО ОДНОМУ ПРИОРИТЕТУ

16K3003	8.544
10K200305	5.396
1M630506	4.069
1M6305	2.994
16B1603	1.343
1A61603	0.0

ДЕТАЛЬ** 719003 ПРИОРИТЕТ**ОБ

ПОСЛЕДОВАТЕЛЬНОСТЬ СТАНКОВ

ПО ОДНОМУ ПРИОРИТЕТУ

1A61603	10.043
16K200305	5.892
1M6303	4.050
16K3003	2.301
16B1603	1.140
1M630306	0.0

ДЕТАЛЬ** 719003 ПРИОРИТЕТ**ТО

ПОСЛЕДОВАТЕЛЬНОСТЬ СТАНКОВ

ПО ОДНОМУ ПРИОРИТЕТУ

16K200305	5.877
16B1603	4.634
1A61603	3.335
1M630306	1.033
16K3003	1.050
1M6303	0.0

Фиг. 3. Преимущественная последовательность станков по одному приоритету согласно суммарной оценке целесообразности.

при помощи системы уравнений, откуда нетрудно вычислить коэффициенты важности. Например, при показателе себестоимости по экспертным оценкам в три раза выше, чем остальные, получим при использовании трех приоритетов следующую систему уравнений:

$$\begin{cases} C_1 + C_2 + C_3 = 3, \\ C_2 = C_3 = \frac{1}{3} C_1, \end{cases}$$

решение которой дает

$$\begin{aligned} C_1 &= 1,8; \\ C_2 = C_3 &= 0,6. \end{aligned}$$

В случае, если $C_j = 1$, из таблицы 2 получается "матрица согласий" $C | C_{k_1, k_2}$ в порядке $(m \times m)$.

Для данного примера "матрица согласий" имеет следующий вид:

$$C = \begin{vmatrix} X & 0,33 & 0,33 & 0,67 & 0,67 & 0,67 \\ 0,67 & X & 0 & 0,33 & 0,33 & 0,67 \\ 0,67 & I & X & I & 0,67 & I \\ 0,33 & 0,67 & 0 & X & 0,33 & 0,33 \\ 0,33 & 0,67 & 0,33 & 0,67 & X & 0,67 \\ 0,33 & 0,33 & 0 & 0,67 & 0 & X \end{vmatrix}.$$

О большем или меньшем "согласии" по различным критериям в пользу отношения "превосходства" $\mu(0B)$ над $\mu(0B)'$ и соответственно о больших или меньших основаниях для проведения в графе G_j дуги $\theta(0B, 0B)'$ можно судить по значению индекса согласия $C(0B, 0B)'$, взяв в качестве такого индекса выражение

$$C(0B, 0B)' = \frac{1}{C} \sum_{\mu_j \in C(\mu(0B), \mu(0B)')} C_j,$$

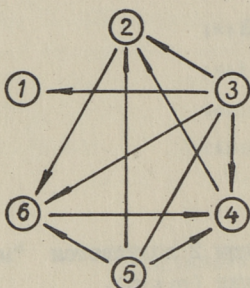
где C — сумма всех C_j .

При одинаковых важностях приоритетов согласно приведенной матрицы C видно, что станок 16K20ФЗ лучше по всем показателям приоритетов, чем станки 16Б16ФЗ, 1М63Ф6, 1М63Ф306 и по двум приоритетам, чем станки 16K30ФЗ и 1А616ФЗ. Данная матрица согласий является основой при комплексном решении поставленной задачи. Некоторые графические решения, при разных исходных условиях, изображены на фиг.4.

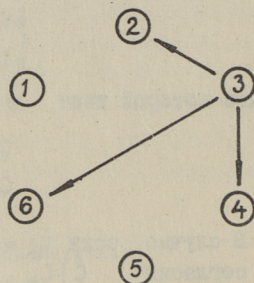
КОМПЛЕКСНАЯ ОЦЕНКА ЦЕЛЕСООБРАЗНОСТИ

Доминирующими являются по крайней мере 2 приоритета и максимальное из возможных отклонений от остальных не превышает 33 %

Доминирующими являются по крайней мере 3 приоритета и максимальное из возможных отклонений от остальных не превышает 10 %



Полученное решение: 16K20Ф3



Полученные решения: 16K20Ф3
1A616Ф3
16K30Ф3

Фиг. 4. Графические решения выбора наилучшей модели станка по комплексной оценке.

Рассмотрим ситуацию, когда коэффициент важности для приоритета себестоимости $C_I = 1,8$, а для приоритетов производительности и точности соответственно $C_2 = C_3 = 0,6$.

Тогда матрицей согласий получаем

$$C = \begin{vmatrix} X & 0,6 & 0,6 & 0,8 & 0,8 & 0,8 \\ 0,4 & X & 0 & 0,2 & 0,2 & 0,8 \\ 0,4 & 1 & X & 1 & 0,6 & 1 \\ 0,2 & 0,8 & 0 & X & 0,6 & 0,6 \\ 0,2 & 0,8 & 0,2 & 0,4 & X & 0,8 \\ 0,2 & 0,2 & 0 & 0,4 & 0 & X \end{vmatrix}$$

Ставя задачу в таком виде, что доминирующими являются, по крайней мере, 2 приоритета и максимальное из возможных отклонений от остальных не превышает 40 %, наилучшим является станок 1A616Ф3 (при исходных условиях $C_I = C_2 = C_3 = 1$, наилучшим оказался станок 16K20Ф3).

Результаты решений показывают, что при помощи данной методики можно решать задачу выбора целесообразной модели станка из множества допустимых решений. При четком сформу-

лировании требований к выбору оборудования можно получить неоднозначно определенный результат. В таких случаях принятие решения производится при помощи человек-машинной процедуры.

Л и т е р а т у р а

1. Брук Б.Н., Бурков В.Н. Методы экспертных оценок в задачах упорядочения объектов. Известия АН СССР. - Техническая кибернетика, № 3, 1972, с. 29-39.

2. Глотов В.А., Павельев В.В. Экспертные методы определения весовых коэффициентов. - Автоматика и телемеханика, № 12, 1976, с. 95-107.

3. Емельянов С.В., Озерной В.М., Гафт М.Г. О построении решающих правил в многокритериальных задачах. Доклады АН СССР, т. 228, № 1, 1976, с. 56-61.

4. Фишберн П. Теория полезности для принятия решений. М., Наука, 1978, с. 352.

5. Подиновский В.В. Многокритериальные задачи с однородными равноценными критериями. Журнал вычислительной математики и математической физики, т. 15, № 2, 1975, с. 330-341.

6. Подиновский В.В. Многокритериальные задачи с упорядоченными по важности однородными критериями. - Автоматика и телемеханика, № 11, 1976, с. 118-127.

7. Подиновский В.В. Коэффициенты важности критериев в задачах принятия решений. Порядковые или ординальные коэффициенты важности. - Автоматика и телемеханика, № 10, 1978, с. 130-141.

8. Ривес Ю.Э. Выбор технологического оборудования. Тезисы докладов Республ. научн.-техн. конф. "Усовершенствование технологической подготовки производства машиностроительных и приборостроительных предприятий с применением средств вычислительной техники". Таллин, 1979, с. 26-29.

The Amount of Priorities Used
for Machine Tool Selection

Summary

A method for selecting the most suitable machine tool for producing a given part is described in the paper. The choice of a machine tool depends on several factors (productivity, technological operating cost; machining accuracy, exploitation reliability, expenditures associated with the preparation of production). On the basis of above-mentioned factors priorities have been worked out.

Two possibilities are shown for solving the problem: using priorities one by one or all together.

On the basis of the data, it is possible to see that according to the various priorities variable machine tools could be most efficient. The final solution can be achieved by using a system of priorities.



Цена 45 коп.