

МЕХАТРОНИКА

УДК 681.3

ПРОЕКТИРОВАНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ ОПЕРАТОРСКИХ СТАНЦИЙ
РАСПРЕДЕЛЕННЫХ СИСТЕМ УПРАВЛЕНИЯ

Т.В. Плугина, доцент, к.т.н., Д.А. Маркозов, доцент, к.т.н., ХНАДУ

Аннотация. Рассмотрена задача проектирования интеллектуальных операторских станций распределенных систем управления (ИОСПС). Выделены вопросы поддержки принятия решений оператором распределенных систем управления, основные этапы проектирования ИОСПС, проведен анализ инструментальных средств для их реализации.

Ключевые слова: проектирование, технологический процесс, распределенная система управления, операторская станция, интеллектуальный интерфейс.

ПРОЕКТУВАННЯ ІНТЕЛЛЕКТУАЛЬНИХ ОПЕРАТОРСЬКИХ СТАНЦІЙ
РОЗПОДІЛЕНИХ СИСТЕМ УПРАВЛІННЯ

Т.В. Плуґіна, доцент, к.т.н., Д.О. Маркозов, доцент, к.т.н., ХНАДУ

Анотація. Розглянуто задачу проектування інтелектуальних операторських станцій розподілених систем управління (ІОСПС). Виділено питання підтримки прийняття рішень оператора розподілених систем управління, основні етапи проектування ІОСПС, проведено аналіз інструментальних засобів для їх реалізації.

Ключові слова: проектування, технологічний процес, розподілена система управління, операторська станція, інтелектуальний інтерфейс.

DESIGNING OF INTELLECTUAL OPERATOR STATIONS OF DISTRIBUTED
CONTROL SYSTEMST. Pluhina, Associate Professor, Candidate of Technical Science,
D. Markozov, Associate Professor, Candidate of Technical Science, KhNAHU

Abstract. The design problem of intellectual operator stations of distributed control systems (IOSDCS) has been considered. The issues of support in making decisions by the operator of distributed control systems, the main design phases of IOSDCS are specified. The analysis of tools for their implementation is carried out.

Key words: design, technological process, distributed control system, operator station, intellectual interface.

Введение

Количественный и качественный состав оборудования, на котором реализуются распределенные системы управления, большой объем информации, сложность решаемых задач, короткое время на принятие решений приводят к несоответствию возможностей человека-оператора требованиям эффективно управлять объектом. Актуальной становится задача проектирования интеллектуальной

операторской станции распределенной системы (ИОСПС) для помощи в оперативном управлении, контроле и прогнозировании развития ситуации на объекте. Такие станции позволяют снизить нагрузку на оператора, повысить эффективность его действий и увеличить надежность функционирования объекта. Возможность реализации ИОСПС появилась благодаря впечатляющим успехам теории и практики искусственного интеллекта (ИИ).

Стремительное развитие инфраструктуры вычислений и систем телекоммуникаций не только открывает новые способы разработки распределенного программного обеспечения (ПО) с качественно новыми свойствами, но и влияет на науку о распределенных вычислениях в аспектах метафор вычислений, архитектур и технологий реализации вычислительных систем, методов и средств разработки приложений. Сегодня распределенная информационно-вычислительная система нового поколения – это сообщество многочисленных автономных разумных программ, которые, равноправно сотрудничая с пользователями и друг с другом, решают большую сложную и достаточно амбициозную задачу в таких областях, как производство, энергетика, торговля, социальная сфера, образование, наука или оборона [1].

Анализ публикаций

Распределённая система управления (в переводе с английского Distributed Control System, сокращенно DCS) – это комплекс технических и программных решений для построения АСУ ТП, характерной чертой которой является децентрализованная обработка данных и наличие распределенных систем ввода и вывода информации, повышенная отказоустойчивость, стандартная и единая структура базы данных.

Развитие распределенных систем управления обусловлено увеличением площадей производства, количества датчиков, модернизации и усложнения стандартных алгоритмов управления сложными объектами [2]. Контроллеры, модули ввода и вывода данных, датчики, исполнительные механизмы разнесены в пространстве. Каждый контроллер работает со своими датчиками и исполнительными механизмами, решает свою локальную задачу, не зависит от других контроллеров. Для выполнения общей задачи и достижения заданных показателей качества контроллеры взаимодействуют с другими контурами и устройствами.

Возможности объектно-ориентированного программирования позволили реализовать данную схему управления сложными объектами [3].

Структура DCS в наиболее классическом виде состоит из трех уровней. Нижний уровень

ввода/вывода содержит в себе датчики, исполнительные механизмы. Средний уровень состоит из контроллеров. Их задача – обработать полученные данные, выдать управляющее воздействие, передать данные на верхний уровень. На верхнем уровне расположены серверы баз данных и операторские станции, задача которых – предоставить человеко-машинный интерфейс оператору и осуществлять обмен с сервером и программируемыми логическими контроллерами (ПЛК).

На операторском уровне интерфейс должен предоставлять и визуализировать все процессы, протекающие в объекте. Именно на этом уровне реализуется ИОСРС [4].

Цель и постановка задачи

Целью работы является повышение эффективности функционирования распределенных систем управления технологическими процессами за счет проектирования и разработки интеллектуальных операторских станций поддержки принятия управленческих решений.

Для создания ИОСРС необходимо интегрировать алгоритмические методы управления сложными объектами и методы искусственного интеллекта для задач, характеризующихся неопределенностью исходной информации. К таким задачам ИОСРС можно отнести: оценку ситуации; прогноз поведения объекта в штатном режиме; прогноз развития аварийных ситуаций; синтез и оценку возможных действий и выбор наилучших. Отличительной особенностью интеллектуальных систем является способность к планированию поведения, адаптации и обучению.

Этапы проектирования ИОСРС

Процесс проектирования ИОСРС и его основные этапы представлены на рис. 1.

Элементарной ячейкой инфраструктуры ИОСРС должен быть автономный, разумный и коммуникабельный программный объект – интеллектуальный агент.

Существует определенная аналогия между понятиями автономии в программировании с использованием агентов и инкапсуляции в объектно-ориентированном программировании.

нии [3]. Объект инкапсулирует (заключает в виртуальную оболочку) некое состояние и контролирует его так, что доступ к нему возможен только через методы, предоставляемые объектом.

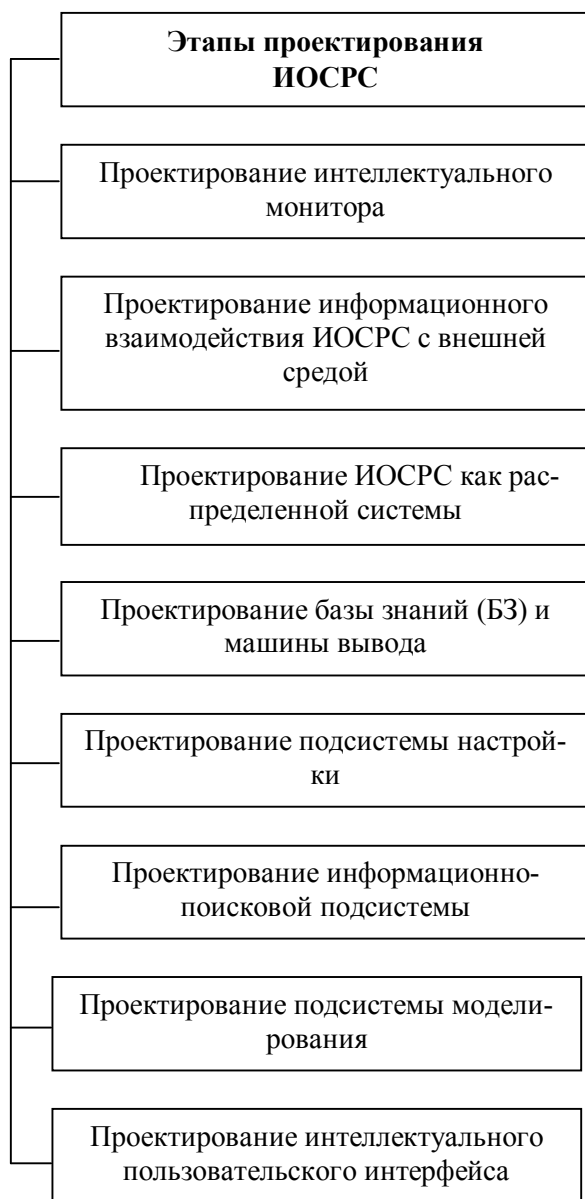


Рис. 1. Процесс проектирования ИОСРС

В отличие от этого, агент инкапсулирует не только состояние, но и поведение. Объект не инкапсулирует поведение, т.к. не контролирует выполнение своих методов. Например, если объект x вызывает метод m объекта y , то объект y не знает, выполнен ли уже метод m или только выполняется. В этом смысле объект y не автономен, т.к. не контролирует свои собственные действия. Напротив, агент полностью управляет своими действиями в любой момент. Поэтому агенты не столько вызывают действия других агентов, сколько

просят других агентов выполнить желаемые действия, и те сами решают, выполнять ли запрошенные действия.

Агент – это вычислительная система, которая внедрена в некоторую среду и способна к автономному поведению в ней согласно своим целям. Система должна быть способна действовать без прямого вмешательства людей (или других агентов) и контролировать свои собственные действия и внутреннее состояние.

Рассмотрим архитектуру простого агента (рис. 2), взаимодействующего с информационной окружающей средой (ИОС), в качестве которой может выступать Интернет, корпоративная сеть (Интранет) или виртуальная частная сеть (VPN).

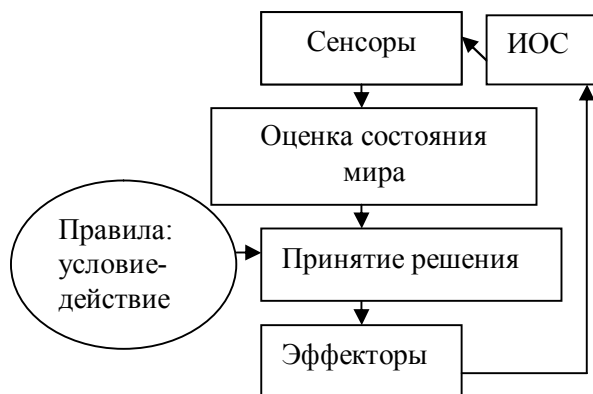


Рис. 2. Архитектура простого агента

Агент воспринимает окружающую среду через свои сенсоры, а затем на основе восприятия выбирает и выполняет действие через свои эффекторы. Сенсорный ввод, помимо прочего, может представлять собой также прием сообщений из сети, а действие – передачу сообщений в сеть.

В плане реализации агента значимыми являются следующие его свойства: уникальная индивидуальность, активность (способность к упреждающему действию), живучесть, автономия и общительность.

Для создания ИОСРС необходимо подобрать соответствующие инструментальные средства, которые должны поддерживать жизненный цикл приложения, используя:

- объектно-ориентированный подход, графическую среду и структурированный естественный язык;

- объектные библиотеки и функциональные модули, которые содействуют сокращению времени разработки приложения;
- машину вывода со средствами для сокращения перебора, реакции на непредвиденные события, богатый набор способов возбуждения правил;
- множественный доступ к централизованной базе знаний и групповую работу с приложением;
- одновременную связь с множеством баз данных и других программных систем;
- взаимодействие с внешним окружением в реальном времени;
- интерактивный редактор, который помогает разработчикам создавать правила, процедуры и модели, используя структурированный естественный язык;
- механизмы поддержки для тестирования и верификации приложения, включая динамическое моделирование для проверки работоспособности по различным сценариям;
- одновременную разработку приложения несколькими группами разработчиков;
- переход на новые платформы без репрограммирования.

Описанными свойствами в разной степени обладают несколько инструментальных интеллектуальных систем реального времени: G2 фирмы «Gensym» (США), RT Works фирмы «Talarian» (США), COMDALE/C («Comdale», Канада), COGSYS («SC», США), ILOG Rules («ILOG», Франция). Однако система G2 обладает явными преимуществами (табл. 1).

Таблица 1 Сравнительный анализ

Свойства ЭС РВ	A	B	C	G2
Работа в реальном времени, внутренний планировщик, параллельные процессы рассуждения	–	+	+	+
Структурированный естественно-языковой интерфейс	–	–	–	+
Общие правила, уравнения и динамические модели	–	–	–	+
Прямой, обратный вывод, сканирование, фокусирование, использование метазнаний	–	–	–	+
Интеграция подсистемы моделирования с динамическими моделями	–	–	–	+
Структурирование БЗ	–	–	–	+
Библиотеки знаний – ASCII-файлы, портируемые на любые аппаратные платформы	–	–	–	+

Окончание табл. 1

Свойства ЭС РВ	A	B	C	G2
Развитый редактор	–	–	–	+
Средства инспекции базы знаний	+	+	–	+
Средства управления доступом с помощью механизма авторизации пользователя	–	–	–	+
Средства трассировки и отладки БЗ	+	–	+	+
Интерфейс оператора включает графики, диаграммы, шкалы, кнопки, редактор многослойных пиктограмм	–	+	–	+
Исполняется на ряде универсальных ЭВМ, рабочие станции DEC, HP, SUN, IBM	+	+	–	+
Кооперация ЭС РВ по сетевому протоколу TCP/IP или DECnet с другими приложениями	–	–	–	+
Удаленные окна, интерактивная многопользовательская работа	–	–	–	+
Интерфейсы с источниками данных обеспечивают эффективную связь с внешними системами и СУБД	–	+	+	+

В табл. 1 система G2 сравнивается со следующими группами инструментальных средств для создания экспертных систем (ЭС) реального времени:

- группа А – статические оболочки ЭС;
- группа В – супервизорные системы управления;
- группа С – оболочки ЭС реального времени.

С помощью G2 интеллектуальная система может работать: на платформах Windows, Unix, VMS; через Internet; с базами данных ORACLE, Sybase, Informix, DBC; с клиентами на Microsoft Visual C++ и Visual Basic; с ведущими сетевыми технологиями.

Приведенные определения понятия «агент» позволяют далее ввести расширенное понятие «мультиагентная система» (МАС) как сообщество взаимодействующих и целенаправленных агентов, где множественное взаимодействие между ними может иметь форму сотрудничества, соперничества или того и другого. В существующей литературе синонимом понятия МАС является более раннее понятие «распределенный искусственный интеллект» (distributed artificial intelligence, DAI [1]).

Управление технологическим процессом – естественная область применения МАС, так как регуляторы процессов представляют собой автономные реактивные системы. МАС эффективны в распределенных системах, где агенты (программы или роботы) размещены в разных местах, получают данные из разных источников и должны координировать свои действия для оптимизации глобального критерия.

Типичные примеры – управление сложными технологическими процессами, обнаружение атак в компьютерных сетях, имитация социальных явлений, робототехника (коллективная работа роботов), дистанционное образование (педагогические агенты) и компьютерные игры (виртуальные игроки). Типичный пример такой системы – ARCHON (Дженнингс, 1995). Это платформа для создания МАС. На базе ARCHON создан ряд приложений, включая управление передачей электроэнергии (Испания) и управление ускорителем элементарных частиц. Каждый агент в ARCHON содержит модуль коммуникаций (МК), модуль планирования и координации (МПК), модуль управления информацией агента (МУИ) и интеллектуальную систему (ИС). Здесь МК управляет обменом сообщениями с агентами, а МПК – действиями агента; МУИ поддерживает в агенте модель мира, а ИС – знания агента о предметной области. При этом ИС упаковывается в «агентную оболочку» (МК, МПК и МУИ) и превращается в агент.

Выводы

Развитие интеллектуальных систем поддержки принятия решений в распределенных системах управления свидетельствует об их большой перспективности для применения в управлении сложными объектами.

Разработка методов проектирования интеллектуальных систем в информационно-управляющих системах на сложных объектах

обеспечит решение задач автоматизации процесса генерации решений.

Достижения искусственного интеллекта и использование современных инструментальных средств создания интеллектуальных приложений позволят проектировать сложные иерархические интеллектуальные операторские станции распределенных систем управления, повысить эффективность действий оператора и увеличить надежность функционирования объекта.

Теория МАС находится на стыке новейших достижений ИИ, теории рационального поведения и деятельности, теории организации, теории малых групп, семиотики, теории коммуникаций, управления и экономики, что однозначно показывает чрезвычайную важность концепции МАС при проектировании ИОСПС.

Литература

1. Шахмаметов Р.Г. Распределенные системы искусственного интеллекта: учеб. пособие / Р.Г. Шахмаметов. – Новосибирск: НГТУ, 2007. – 36 с.
2. Люгер Д.Ф. Искусственный интеллект: стратегии и методы решения сложных проблем / Д.Ф. Люгер. – М.: Издательский дом «Вильямс», 2003. – 136 с.
3. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++ / Г. Буч; пер. с англ. – М.: Бином, 1999. – 112 с.
4. Ногин В.Д. Принятие решений в многокритериальной среде: количественный подход / В.Д. Ногин. – М.: ФИЗМАТЛИТ, 2002. – 144 с.

Рецензент: О.Я. Никонов, профессор, д.т.н., ХНАДУ.

Статья поступила в редакцию 10 декабря 2013 г.