РАЗДЕЛ III. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Амбалова З.А., Гахова Н.Н.

Имитационное моделирование задачи распределения обязанностей участников комплексных проектов

Белгородский государственный национальный исследовательский университет
(Россия, Белгород)

doi:10.18411/spc-12-03-2018-03

idsp: 000001:spc-12-03-2018-03

Метод имитационного моделирования сочетает в себе особенности экспериментального подхода в условиях использования вычислительной техники при исследовании реальной системы по ее имитационной модели [1,2].

В качестве комплексного проекта будем понимать проект, ограниченный временными рамками, направленный на разработку, создание и внедрение крупных прикладных информационных систем [1]. При этом необходимо распределить и выполнить M различных работ между N сотрудниками. Эффективность выполнения проекта зависит от команды участников и уровня их компетенции, взаимодействия и распределения обязанностей между ними. Поэтому задача распределения обязанностей участников комплексных проектов в современных условиях является актуальной.

Для моделирования рассматриваемого процесса, целесообразно представить его как стохастическую систему, в которой переменные на входе и выходе являются случайными величинами, функциями, процессами или последовательностями. При этом необходимо проводить серию прогонов и статистическую обработку полученных результатов, чтобы иметь объективную характеристику изучаемого объекта [2]. Такую задачу можно отнести к задачам, решаемым методами математического программирования, как задачу о назначениях. Для ее решения необходимо устранить дисбаланс, добавив недостающие работы (если $n > m$) или сотрудников (если $n < m$). В этом случае требуется минимизировать ее целевую функцию (1):

$$Z = \sum_{i=1}^n \sum_{j=1}^n c_{ij} * x_{ij} \quad (1)$$

т.е. необходимо распределить и выполнить n различных работ i между n сотрудниками j , так чтобы общие затраты Z на выполнение всех работ были минимальными. Исходными данными является таблица $C = \{c_{ij}\}$, с равным количеством строк и столбцов, элементами которой являются показатели эффективности назначений:

работа \ сотрудник	1	2	...	j	...	n
1	c_{11}	c_{12}	...	c_{1j}	...	c_{1n}
...	...	...	...	...	...	...
i	c_{i1}	c_{i2}	...	c_{ij}	...	c_{in}
...	...	...	...	...	...	...
n	c_{n1}	c_{n2}	...	c_{nj}	...	c_{nn}

Задача о назначениях реализуется венгерским методом, алгоритм которого выполняется пошагово с получением редуцированной матрицы. Такая система была реализована в среде Borland C++ Builder, результаты представлены на рисунке 1.

Рисунок 1. Реализация венгерского алгоритма в среде Borland C++ Builder

Реализация такой системы позволяет эффективно распределить сотрудников на требуемые работы. При этом план работ составлен так, что все работы выполнены, каждый сотрудник занят, а общие затраты на выполнение всех работ минимальны.

При моделировании важно понимать насколько точно модель описывает реальный объект моделирования. Для этого существуют две проверки: валидация и верификация. Их основная цель – исследование свойств имитационной модели таких как: точность, устойчивость и чувствительность результатов моделирования [1].

Таким образом, используя методы имитационного моделирования и современные информационные технологии, было получено эффективное решение задачи о назначениях.

1. Лычкина Н.Н. Имитационное моделирование экономических процессов // Учебное пособие для слушателей программы eMBI / Лычкина Н.Н., Академия АйТи, Государственный университет управления, 2005
2. Сакалаускас Л. Нелинейная стохастическая оптимизация методом Монте-Карло // Стохастическая оптимизация в информатике: Межвузовский сб. / Под ред. проф. О.Н. Граничина. – СПб.: Изд-во СПб гос. ун-та, 2005.–Вып. 1.–С.190–205.

Преображенская Т.В.

Минимизация точек ввода данных в информационных системах

*Новосибирский государственный технический университет
(Россия, Новосибирск)*

doi:10.18411/spc-12-03-2018-04

idsp: 000001:spc-12-03-2018-04

При разработке баз данных (БД) информационных систем (ИС) аналитики опираются на известные принципы [1]. Их соблюдение позволяет обеспечить: недопустимость неточно введенных данных или дублирование; защиту от ошибок при обновлении; невозможность удаления связанных данных разных таблиц; неискажение данных при работе в многопользовательском режиме и др.

Но, любая ИС в процессе ее эксплуатации претерпевает ряд изменений - растет или уменьшается количество пользователей и рабочих мест, появляются новые запросы или исчезают те, что были раньше. Изменяются интерфейсы (формы экранов и