

ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ НА ПРИМЕРЕ СОСТАВЛЕНИЯ РАСПИСАНИЯ ЗАНЯТИЙ ОБРАЗОВАТЕЛЬНОЙ ОРГАНИЗАЦИИ

А.В. Затонский, С.А. Варламова

*Пермский национальный исследовательский политехнический университет,
Березниковский филиал, г. Березники, Пермский край, Россия*

Поставлена задача обучения эффективному использованию информационных технологий при поддержке принятия решений. Систематизированы критерии оценки деятельности экономических и социальных систем, приведена их классификация и области применения частных критериев различной природы. Предложено для интеграции знаний в области информационных технологий с целью поддержки принятия решений использовать актуальную задачу составления расписания занятий. Определены исходные данные, классифицированы факторы модели, построена математическая модель, указаны ее ограничения. Приведены методы получения критериев качества расписания с различных точек зрения и получения интегрального критерия качества. Рассмотрены и классифицированы ограничения возможных решений. Приведены примеры случаев, когда формульные критерии неприменимы. Обоснован выбор алгоритмов оптимизации по интегральному критерию. Показан переход от структурных моделей деятельности лиц, принимающих решения, к информационным моделям систем поддержки принятия решений. Предложена концепция поддержки принятия решений экспертом путем оценки его трудно формализуемых действий по изменению значений параметров системы. Множества объектов системы сопоставлены сущностям информационной модели, приведены ее улучшающие преобразования. Рассмотрены варианты информационной модели с различной степенью нормализации, показаны их сравнительные достоинства и недостатки. Определены направления развития информационной модели в составе автоматизированной системы управления вузом. Сделан вывод о возможности повышения эффективности деятельности сложных социально-экономических систем за счет рационального использования средств математического, информационного и имитационного моделирования. Показано, что эта область деятельности в настоящее время перспективна с точки зрения эффективного приложения усилий специалиста в области информатики.

Ключевые слова: вуз, автоматизация, принятие решений.

Введение

В практической деятельности по разработке информационных систем и при обучении студентов-информатиков наступает момент, когда требуется собрать воедино и эффективно применить компетенции в области программирования, математического и информационного моделирования, теории принятия решений и оптимизации с целью решения задачи управления организацией. Практика показывает, что подобное дается студентам непросто. Да, они все это проходили и вроде бы все умеют, но последовательно пройти все шаги от описания проблем предметной области до построения эффективной информационной модели и проектирования интерфейсов не могут. Возникают множественные вопросы: что связывает критерий оптимальности деятельности и информационную модель? Как превратить записанные словесно ограничивающие условия в алгоритмы? Как перейти от информационного обеспечения к, собственно, принятию решений? Как на практике обеспечить поиск если не оптимальных, то хотя бы *хороших* решений?

Проще, если может быть применен финансовый, «денежный» критерий оценки деятельности. В этом случае задача часто получается однокритериальной, а многокритериальные задачи легко сводятся к однокритериальным, например, путем линейной комбинации критериев. Финансовые критерии так или иначе сводятся либо к вычислению какой-то суммы денег (ЧД, ЧДД, *NPV*) или какого-то «коэффициента при деньгах» (наподобие *IRR*, *MIRR*), либо к определению положения организации на конкурентном рынке в некоторой воображаемой системе координат (результаты *PEST/STEP*, *SWOT* и т. п. методик) [10].

Сложнее, если речь идет об автоматизации социальной системы, не имеющей одного явно выраженного критерия оценки деятельности. В них в качестве основного, мажорирующего, может быть принят имиджевый, экологический, социальный критерий. Последний допускает много интерпретаций (например, отношение горожан к качеству работы городской администрации; эффективность работы инспекции по налогам и сборам не с точки зрения объема собранных штрафов, а в смысле полноты исполнения законов; оценка социальной полезности использования природно-ресурсного потенциала и т. п.). Способ его измерения обычно – непосредственная экспертиза или вычисление по ряду экспертных оценок.

В учебном процессе в качестве подобной системы просто и естественно рассматривать вуз, так как студенты имеют представление о предметной области и способны формулировать свои взгляды на качество управления. Одной из задач, которая решается постоянно и все же никогда не имеет оптимального решения, является задача составления расписания занятий вуза. Каждый год успешно защищаются диссертации на эту тему, предлагаются новые методы и алгоритмы оптимизации расписания [9], разрабатываются новые информационные системы [12]. Тем не менее в каждом вузе есть свои особенности, свои граничные условия и могут быть предложены свои критерии качества расписания [16, 18], что делает эту задачу «бесконечной» в том смысле, что новые ее решения всегда актуальны и иногда практически полезны.

Рассмотрим последовательность действий, некоторые приемы информационного обеспечения, подходы к разработке информационно-управляющей системы при автоматизации процесса составления и оптимизации расписания занятий на примере Березниковского филиала Пермского национально-исследовательского политехнического университета (БФ ПНИПУ) [1].

1. Исходные данные

БФ ПНИПУ включает два учебных корпуса, между которыми несколько остановок общественного транспорта. Занятия проводятся в полуторачасовые интервалы времени (пары) в обоих корпусах. Занятия проводятся как по дневной, так и по очно-заочной (вечерней) форме обучения: по дневной – с 8:30 до 15:40, по вечерней – с 18:30 до 21:30.

Перечень дисциплин, количество аудиторных часов по ним и формы контроля (последние – для составления расписания сессии) устанавливаются рабочим учебным планом (РУП) или переходными учебными планами, разными для разных групп. Обязательное требование к расписанию – в течение семестра все занятия должны быть фактически проведены. РУП предусмотрены виды занятий: лекция, практикум, лабораторная работа.

Основные образовательные программы (ООП) закреплены за выпускающими кафедрами. Есть и не выпускающие кафедры. Дисциплины в РУП закреплены за кафедрами и, как правило, проводятся преподавателями кафедры, за которой закреплены, хотя могут быть и исключения.

Почти все аудитории закреплены за различными кафедрами и по-разному оборудованы. Есть компьютерные классы, аудитории с проекторами, потоковые аудитории. Емкость аудиторий – от 20 до 120 посадочных мест.

Преподаватели числятся на кафедрах. Кроме штатных преподавателей, есть преподаватели-совместители, загруженность которых по основному месту работы непредсказуема (командировки, мероприятия и все остальное), что мешает проведению занятий.

Существует расписание звонков, различное в разные дни недели. Однако каждая пара занятий может быть с учетом дня недели однозначно идентифицирована ее номером. В воскресенье занятия не проводятся.

Ставится задача составления хорошего расписания, обычная для социальных систем. Причем четкого понимания того, какое расписание является «хорошим», у постановщика задачи обычно не бывает. Более того, при ознакомлении с предметной областью, как правило, выясняется, что существует несколько точек зрения на то, как оценивать качество расписания, противоречащих одна другой.

2. Математическая модель и ограничения

Введем обозначения:

I – количество учебных групп студентов (сквозная нумерация по всем направлениям подготовки);

i – номер учебной группы;

J – количество преподавателей (сквозная нумерация по всем кафедрам);

$j = \overline{1, J}$ – уникальный номер преподавателя;

K_p – количество пар в день недели $p = \overline{1, 6}$;

$v = \overline{1, 3}$ – количество видов занятий (лекции, лабораторные, практические);

N – количество дисциплин (сквозная нумерация по всем направлениям подготовки); $n = \overline{1, N}$ – уникальный номер дисциплины;

M_i – количество семестров в учебном плане группы i (обычно $\forall i M_i \leq 12$), $m = \overline{1, M_i}$ – номер семестра;

L – количество аудиторий (сквозная нумерация в пределах вуза); $l = \overline{1, L}$ – уникальный номер аудитории.

В случае если расписание изменяется от недели к неделе, добавится номер недели $y = \overline{1, Y_i}$, где Y_i – количество недель в учебном году в соответствии с учебным планом группы i .

Математической моделью, отражающей факт проведения каждого занятия, является пересечение перечисленных выше множеств групп, преподавателей, дисциплин, аудиторий, семестра, недели, дня и пары:

$$Q = I \times J \times N \times L \times M_i \times Y_i \times P \times K_p.$$

Проводимое занятие характеризуется каким-то видом из множества V . Если ограничиться $v = \overline{1, 3}$, как предложено выше, то в расписание невозможно будет включить консультации, зачеты, экзамены, прием курсовых работ, мероприятия ИГА и т. п., поэтому будем для общности считать, что $v = \overline{0, V}$, где $v \equiv 0$ соответствует видам занятий, не предусмотренным учебным планом (дополнительные консультации, пересдачи и т. п.), а в реальности $V > 3$.

Введем булеву переменную $Q_{ijnlmypk}$ такую, что $Q_{ijnlmypk} = 1$, если у группы i преподаватель j ведет пару номер k в день недели p недели y семестра m по дисциплине n в аудитории l , причем существуют некоторые алгоритмы:

- определения вида занятия $v(Q_{ijnlmypk}) \in \begin{cases} 0 \Leftrightarrow Q_{ijnlmypk} = 0 \\ \overline{1, V} \Leftrightarrow Q_{ijnlmypk} = 1 \end{cases}$;
- определения номера корпуса $r(Q_{ijnlmypk}) \in \begin{cases} 0 \Leftrightarrow Q_{ijnlmypk} = 0 \\ \overline{1, R} \Leftrightarrow Q_{ijnlmypk} = 1 \end{cases}$,

где R – количество корпусов; причем для каждой пары корпусов известно расстояние (либо, что то же самое, время перемещения между ними) $r_{ij} = f(r_i, r_j)$; разумеется, $r(Q_{ijnlmypk}) \equiv r(l)$;

– определения количества студентов в группе i : $ICnt_i = f(i)$;

– определения количества посадочных мест в аудитории l : $LCnt_l = f(l)$;

– определения назначения (типа, вида) аудитории – множества $Z_l = N^* \times V^*$, определяющего, что в аудитории l можно вести занятия видов $v \in V^*$ по дисциплинам $n \in N^*$, причем $V^* \subseteq V$ и $N^* \subseteq N$;

– определения принадлежности аудитории кафедре $d = \overline{0, D}$, где D – количество кафедр: $LDep_l = f(l)$, причем $LDep_l \equiv 0$, если аудитория l не закреплена ни за какой кафедрой;

– определения принадлежности преподавателя кафедре $d = \overline{1, D}$: $JDep_j = f(j)$, причем $\forall j JDep_j > 0$.

В этом случае, например, количество занятий в день у одной группы можно подсчитать как

$$\sum_{k=1}^{K_p} Q_{ijnlmypk} \text{ при конкретных заданных } \{i, j, n, l, m, y, p\}.$$

Расписание является допустимым, если выполняется большое количество ограничений. Соответствие некоторым из ограничений проверяется по уже составленному расписанию, например:

1. Количество занятий каждого конкретного вида v^* по каждой дисциплине n в семестр m для группы i определено учебным планом: $Z_1^i = \sum_y \sum_p Q_{ijnlmpyk} : v(Q_{ijnlmpyk}) = v^*$ при конкретных заданных $\{i, v, n, m\}$; аналогично определено и количество занятий в неделю.

2. Нагрузка каждого преподавателя j , как правило, находится в интервале заданных значений: $\forall j Z_{20}^j \leq \sum_{i,p,v,n,m,y} Q_{ijnlmpyk} \leq Z_{21}^j$.

Другие ограничения должны применяться постоянно при составлении расписания. Например:

3. Ни одна группа не может находиться более чем на одном занятии в одно и то же время: $\forall \{j, n, l, m, y, p, k\} Q_{i(jnlmpyk)_1} \cdot Q_{i(jnlmpyk)_2} = 0$ (в дальнейшем изложении $\forall \{\dots\}$ опущено, определяется индексами в скобках).

4. Ни в одной аудитории не могут проводиться занятия по двум дисциплинам одновременно: $Q_{l(ijmypk)_{n_1}} \cdot Q_{l(ijmypk)_{n_2}} = 0 \Leftrightarrow n_1 \neq n_2$.

5. Ни один преподаватель не может проводить занятия по двум дисциплинам одновременно: $Q_{j(ilmypk)_{n_1}} \cdot Q_{j(ilmypk)_{n_2}} = 0 \Leftrightarrow n_1 \neq n_2$.

6. Ни один преподаватель не может проводить занятия в двух аудиториях одновременно: $Q_{j(inmypk)_{l_1}} \cdot Q_{j(inmypk)_{l_2}} = 0 \Leftrightarrow l_1 \neq l_2$.

7. Санитарно-гигиеническими нормами (СГН) и/или государственным образовательным стандартом (ГОС, ФГОС) ограничено количество аудиторных занятий у группы i в день $\sum_{k=1}^{K_p} Q_{(ijnlmpy)k} \leq Z_{3i}$ и в неделю $\sum_p \sum_k Q_{(ijnlmy)pk} \leq Z_{4i}$. Например, в соответствии с СГН, у студентов дневной формы обучения в день не должно быть больше 4 пар, в неделю – 20 пар; для вечерней (очно-заочной) формы, соответственно, 2 и 8.

8. Количество посадочных мест в аудитории l не должно быть меньше количества студентов в группе i : $ICnt_i \leq LCnt_l \Leftrightarrow Q_{il(jnmypk)} = 1$.

9. Количество занятий, проводимых в каждую пару k любого дня p любой недели у любого семестра m , не должно превышать количество аудиторий: $\sum_{ijnl} Q_{(ijnl)mypk} \leq L$.

Список не является исчерпывающим и может дополняться в зависимости от условий в конкретном вузе [1]. Выполнение этих ограничений (обозначим их $X_i \in \{0, 1\}, i = \overline{1, I_{\max}}$, где I_{\max} – количество строгих ограничений), то есть $\forall i \in \{1, I_{\max}\} X_i = 1$ означает, что построенное расписание допустимо.

3. Свободные переменные

Не все варианты составления расписания являются независимыми. Если не учитывать подмену одних преподавателей другими (на время болезни, командировки и т. п.), то проведение занятий по дисциплине у группы в течение семестра жестко связано с конкретным преподавателем и нагрузкой. Если обозначить множество нагрузок в семестре m как $W_m = (I \times J \times N)_m$ мощностью $|W_m| = W_m$, равной количеству записей о нагрузках преподавателей в конкретном семестре, мощность множества Q можно уменьшить за счет уменьшения количества независимых переменных с $\{i, j, n, l, m, y, p, k\}$ до $\{w, l, m, y, p, k\}$ или до $\{w, l, y, p, k\}$ – при составлении нагрузки и расписания на конкретный семестр (или год, что не принципиально).

4. Критерий качества

Для *оптимизации* или сравнительной оценки разных вариантов расписания необходимо сформулировать критерий качества как суперпозицию различных подходов (точек зрения) к оценке качества расписания.

Перечислим разные точки зрения на этот вопрос, а также *нестрогие ограничения*, выполнение которых определяет качество расписания, важные с этих точек зрения.

1. Точка зрения студента.

1.1. Не должно быть в один день занятий в разных корпусах у одной группы $k_1 \neq k_2 \Rightarrow r(Q_{(jnlmy)ipk_1}) \neq r(Q_{(jnlmy)ipk_2})$.

1.2. Не должно быть слишком много занятий по одной дисциплине подряд у одной группы $Q_{(jlm)nipk} = Q_{(jlm)nip(k+1)} = Q_{(jlm)nip(k+2)} = \dots$

1.3. Не должно быть слишком много занятий у одного преподавателя подряд (если он ведет несколько дисциплин у этой группы) $Q_{(nlm)ijpk} = Q_{(nlm)ijp(k+1)} = Q_{(nlm)ijp(k+2)} = \dots$

1.4. Не должно быть «окон» в расписании, то есть длительных перерывов между занятиями у группы $Q_{(jnlmy)ipk} \neq 0$, $Q_{(jnlmy)ip(k+1)} = 0$, $Q_{(jnlmy)ip(k+2)} \neq 0$ и т. д.

2. Точка зрения преподавателя.

2.1. Не должно быть в один день занятий в разных корпусах (это допустимо, только если между занятиями есть длительный перерыв) $k_1 = k_2 + \varepsilon \Rightarrow r(Q_{(inlmy)jpk_1}) \neq r(Q_{(inlmy)jpk_2})$, где $\varepsilon > 0$ и $\varepsilon < \varepsilon_{\max} \approx \overline{1, 2, 3, \dots}$ – индивидуальная оценка на усмотрение j -го преподавателя.

2.2. Не должно быть слишком много занятий подряд у одной группы – не важно, по одной и той же дисциплинам или нескольким разным $Q_{(nlm)ijpk} = Q_{(nlm)ijp(k+1)} = Q_{(nlm)ijp(k+2)} = \dots$

2.3. Нельзя или нежелательно назначать занятия в неудобное для преподавателя время (перечень интервалов времен $\{M, Y, P, K\}_j$ может быть назначен каждым j -м преподавателем).

2.4. Обычно преподавателю нравится, если расписание по дням недели неравномерное, то есть оставляет свободным некоторый (некоторые) «методический» день недели p_j , так что $Q_{(ijnlmyk)p_j} = 0$.

2.5. Нежелательно назначать занятия в неудобной для j -го преподавателя аудитории l_j (или их множестве L_j): $Q_{(ijnmykp)l_j} = 0$. Так, одна из преподавателей, очень невысокая, просто не могла читать лекции в аудиториях, где доска висит высоко.

2.6. Желательно назначать занятия j -го преподавателя в тот же корпус, где находится его кафедра: $r(Q_{(inlmykp)_j}) = LDep(l)$.

3. Точка зрения кафедры.

3.1. Занятия преподавателям кафедры предпочтительно назначать в аудитории, закрепленные за кафедрой $Q_{(inmykp)jl} : JDep_j = LDep_l$.

3.2. Наоборот, в некоторые аудитории кафедры L^* (например, оснащенные дорогостоящим лабораторным оборудованием) нежелательно назначать занятия преподавателям других кафедр $Q_{(inmykp)jl}, l \in L^* : JDep_j \neq LDep_l$.

4. «Точка зрения» дисциплины (в связи с громоздкостью, формальное определение ниже опускается).

4.1. Лабораторные занятия должны назначаться в аудитории, оснащенные соответствующим оборудованием.

4.2. В частности, лабораторные и практические занятия, связанные с использованием персональных компьютеров (ПК), должны назначаться в компьютерные классы.

4.3. По дисциплинам, предусматривающим в одном и том же семестре весь теоретический материал и курсовую работу (КР), материал лучше «выдать» в начале семестра (то есть ставить занятия чаще), чтобы студенты быстрее могли приступить к выполнению КР.

4.4. Последовательность занятий разного вида должна обеспечивать, что необходимый теоретический материал выдается до того, как по нему проходят практические или лабораторные занятия. Обычно это приводит к тому, что в начале семестра должно быть больше лекций, чем в конце.

4.5. В случае, если изучение дисциплины n_1 предполагает знание основ другой дисциплины n_2 , а в РУП обе изучаются в одном и том же семестре, в начале семестра лучше провести занятия по дисциплине n_2 , а в конце – по n_1 .

5. Точка зрения администрации.

5.1. Расписание должно быть равномерным и не изменяться часто (в некоторых вузах считается, что расписание может меняться только один раз в середине семестра).

5.2. Для уменьшения (вернее, высвобождения) фонда оплаты труда предпочтительно объединять группы в потоки по одной дисциплине.

5.3. Расписание предпочтительно составлять таким образом, чтобы снижались издержки образовательного процесса. Например, скомпоновать выполнение лабораторных работ с использованием быстро портящегося материала во времени.

Список не является исчерпывающим. Могут быть и другие нестрогие ограничения, наподобие пожелания преподавателя, чтобы конкретная дисциплина ставилась в расписание в конкретный день недели, чтобы в какие-то дни недели не ставились какие-то конкретные пары (то есть, например, раньше обеда не ставить занятия), чтобы была высокая (примерно одно и то же количество пар во все дни недели) или низкая равномерность расписания (все пары в один день, много свободных дней) и т. п.

Приведенные показатели можно классифицировать как:

1) Непосредственно вычисляемые, например, 1.4 (продолжительность «окна»), 2.2 (количество пар «подряд») и т. д.

2) Определяемые как значения δ -функции («есть», «выполняется»; «нет», «не выполняется»), например, 1.1, 2.6, 3.1 и т. д.

3) Оцениваемые в терминах нечеткой математики или нечетких множеств, интервальных оценок («очень много», «много», «нормально», «мало», ...), например, 2.3, 4.3 и т. д.

Обозначим Y_k степень выполнения каждого из $k = \overline{1, K_{\max}}$ приведенных выше ограничений. Для непосредственно вычисляемых нестрогих ограничений $Y_k > 0$ (если ограничение не выполнено, $Y_k = 0$); для упрощения дальнейших расчетов разумно нормировать $Y_k^* = Y_k / \max(Y_k) \in [0, 1]$.

В ряде случаев судить, *насколько* выполнено ограничение, все же придется человеку, даже если ограничение выглядит вполне вычислимым. Например, значение $\max(Y_k)$ невозможно обоснованно вычислить, если речь идет о п. 2.2 или 3.2, поэтому в целом оценка начинает приближаться к экспертной («выполнено», «в целом выполнено», ... «не выполнено совсем»). Значения δ -функции могут быть представлены как $Y_k \in \{0, 1\}$, а нечетким экспертным оценкам может быть сопоставлено рациональное значение (или статистическое распределение вероятности оценки), несложно также нормировать его, чтобы $Y_k \in [0, 1]$. Это дает возможность сформулировать критерий оптимальности в виде *штрафной* функции [7], объединяющей строгие и нестрогие ограничения

$$S = M \sum_{i=1}^{I_{\max}} X_i - \sum_{k=1}^{K_{\max}} \alpha_k Y_k \rightarrow \min, \quad (1)$$

где α_k – вес (важность) соответствующего нестрогого ограничения, $M \gg 1$ (точнее, $M \gg \sum_{k=1}^{K_{\max}} \alpha_k Y_k$) – большая константа, существенно увеличивающая значение S в случае, если хотя бы одно из строгих ограничений не выполнено.

Очевидно, что задача Парето-оптимальная. Пожелания 4.4 и 5.1 прямо противоречат друг другу. Разным преподавателям может понадобиться на одну и ту же пару одна и та же мультимедийная аудитория: одному для лекций, другому для лабораторной работы. Для того, чтобы осуществить обоснованный выбор в последнем случае, введем понятие *приоритета j -го преподава-*

теля $P_j = \sum_{q=1}^Q \beta_q P_q^j$, где $Q > 0$ – количество факторов, определяющих приоритет преподавателя,

β_q – вес q -го фактора (для практического применения удобно нормировать веса, чтобы выполнялось $\sum_{q=1}^Q \beta_q = 1$), P_q^j – значение q -го фактора приоритета j -го преподавателя. Можно предполо-

жить, что в качестве факторов будут учитываться:

- 1) ученая степень преподавателя;
- 2) ученое звание преподавателя;
- 3) стаж работы;
- 4) административная должность преподавателя-совместителя (декан, заведующий кафедрой и т. п.);
- 5) неформальное положение преподавателя в коллективе и т. п.

В этом случае для всех преподавателей можно вычислить нестрогое ограничение как

$Y_k = \sum_{j=1}^J P_j (Y_k)_j$, где $(Y_k)_j$ – выполнение k -го нестрого ограничения для j -го преподавателя.

Недостаток такой штрафной функции очевиден: при определенных значениях весов можно добиться увеличения слагаемого $\sum_{k=1}^{K_{\max}} \alpha_k Y_k$ за счет одного-двух выполненных нестрогих ограничений.

Такая же ситуация возникает, когда в задаче оценки эффективности деятельности в социальной системе принимается критерий качества $K = \sum \alpha_i x_i$ и ставится задача $K \rightarrow \max / x_i \neq 0$. Включить

такие показатели в выражение критерия можно, приняв $K = \sum_i \alpha_i x_i + \sum_k \frac{1}{x_k + \varepsilon}$, где $0 < \varepsilon \ll 1$ для

предотвращения аварийных остановов при расчетах. Аналогично, из (1) можно получить

$$S = \sum_{i=1}^{I_{\max}} \frac{1}{(1 - X_i) + \varepsilon} - \sum_{k=1}^{K_{\max}} \alpha_k Y_k \rightarrow \min. \quad (2)$$

Задача в любой постановке, (1) или (2), не является задачей линейного программирования, так как все X_i и часть Y_k могут принимать только значения 0 или 1. Количество свободных переменных в задаче (2) определяется мощностью множества Q , иначе говоря, произведением:

- количества групп;
- количества преподавателей;
- количества дисциплин;
- количества аудиторий;
- количества семестров;
- количества недель в семестре;
- количества дней в неделе;
- количества пар в день.

Очевидно, что применение «фронтального перебора», а также применение многих других алгоритмов оптимизации к задаче (2) приведет к практически бесконечному времени расчета варианта расписания.

В ряде случаев математической модели в форме выражения для критерия качества недостаточно.

1. В задаче оптимизации движения городского транспорта [3] в качестве критерия оптимизации можно выбрать сумму времен движения $t_{ij}(t)$ от i -й остановки к j -й остановке, изменяющуюся в течение суток в зависимости от выхода на маршруты подвижного состава, и поставить

задачу оптимизации $T = \int_{t_0}^{t_1} \sum_i \sum_j t_{ij}(t) \Big|_{i \neq j} dt \rightarrow \min$.

Для расчета $t_{ij}(t)$ требуется проводить имитационное моделирование оборотов каждой единицы транспорта [19] с учетом времени разворотов, обеда, переходов с маршрута на маршрут и т. п.

2. В задаче оптимизации управления научно-исследовательской работой студентов [6] официальная оценка достижений подразделения производится по формуле $R = \sum_{i=1}^I \alpha_i \cdot N_i$, где N_i –

i -й показатель (например, количество опубликованных статей), α_i – вес показателя, $i = \overline{1, I}$. Для прогнозирования возможности опубликования результатов научной работы к заданному моменту времени требуется проводить имитационное моделирование процесса «продвижения» статьи по разным стадиям издания с учетом особенностей журнала. Подобные задачи возникают в любых рейтингах с линейной (или нелинейной) комбинацией критериев, каждый из которых «внутри себя» является довольно сложным, вычисляемым по отдельной модели. В частности, такая природа моделей показателей исключает использование методов линейного программирования для оптимизации.

3. В задаче оптимизации графика планово-предупредительных ремонтов (ППР) по экономическому критерию (обычно это Парето-оптимальные минимизация затрат и максимизация доходности производства) сами критерии вычисляются несложно, но для планирования ППР с учетом ресурсных ограничений (кадры, оборудование, сроки поставки запасных частей и т. п.) также требуется имитационное моделирование [3, 5].

Таким образом, во многих задачах управления как социальными, так и экономическими системами, имитационная модель (а точнее информационная система, частью которой является реализация имитационной модели), является неотъемлемой частью информационного обеспечения лица, принимающего решения [4].

5. Алгоритм оптимизации

Для задач большой размерности (в том числе, для составления расписаний вузов [17]) считаются пригодными следующие формальные методы оптимизации:

Метод «Монте-Карло», случайного поиска [8].

1. Формируется первое решение Q_1 (возможно, тривиальное или случайное) и запоминается как оптимальное $Q^* = Q_1$.

2. С использованием генератора случайных чисел формируется следующее n -е решение.

3. Для него рассчитывается критерий $S(Q_n)$.

4. Если $S(Q_n) < S(Q^*)$, то решение необходимо запомнить: $Q^* = Q_n$.

5. Повторять шаги 2–4 до выполнения одного из условий:

- критерий не улучшился при генерации наперед заданного количества решений $n, n + 1, \dots, n + N$;
- истекло время, отведенное на оптимизацию расписания;
- получено расписание, качество которого лучше наперед заданного $S(Q_n) < S_{\min}$.

Однако для большинства задач качество решения, полученного таким стохастическим методом, примерно пропорционально корню из количества сгенерированных решений, поэтому для улучшения качества вдвое понадобится вчетверо больше решений и т. п.

Генетические алгоритмы [11]. Некоторым образом формируется начальная популяция решений Q_1, Q_2, \dots, Q_n , записываемых в виде последовательности «генов» (обычно 0 и 1, что хорошо соответствует характеру X_i и δ -функций из числа Y_k). Затем путем применения генетических операторов (кроссинговера, мутации, инверсии и других) пытаются генерировать решения лучшие, чем уже находящиеся в популяции [14]. Так, если исходные «особи» (последовательности генов), называемые «родителями», представлены в виде $x_1 x_2 \dots x_k x_{k+1} \dots x_n$ и $y_1 y_2 \dots y_k y_{k+1} \dots y_n$, то в результате кроссинговера по хромосоме k будет получена пара $x_1 x_2 \dots x_k y_{k+1} \dots y_n$ и $y_1 y_2 \dots y_k x_{k+1} \dots x_n$. Методами выбора родителей могут быть:

- пропорциональный отбор;
- турнирный отбор;
- элитный отбор;

- случайный отбор;
- инбридинг;
- аутбридинг;
- селективный отбор.

Кроме того, каждый генетический оператор имеет множество своих настроек (так, для кроссинговера это одноточечный, многоточечный или унифицированный кроссинговер; размер кроссинговера; маска кроссинговера и т. п.). Поэтому практическое применение метода генетических алгоритмов для оптимизации расписания означает переход от множества неизвестных в решении ко множеству неизвестных параметров настройки алгоритма, при котором он будет за разумное время выдавать лучшие популяции [20].

Возможно применение и *других* методов [2, 13], например:

- переборы с отсекающими;
- декомпозиционные методы;
- разновидности метода неопределенных коэффициентов и т. д.

Однако на практике даже построение базового решения (то есть такого, что $\forall i \in \{1, I_{\max}\} X_i = 1$), является сложной задачей и может занимать продолжительное время.

В то же время эксперт-составитель расписания, руководствуясь трудно формализуемыми эвристиками, довольно быстро может составить вариант расписания Q_1 , удовлетворяющий большинству строгих ограничений, а затем путем перестановок занятий добиться выполнения всех строгих ограничений. Надо только, чтобы какой-то сервис (например, информационная система) быстро отвечал на два вопроса:

- какие именно строгие ограничения X_i не выполнены?
- как изменилось (стало ли лучше) значение критерия S после очередного изменения рас-

писания (то есть как изменилось слагаемое $\sum_{k=1}^{K_{\max}} \alpha_k Y_k$ в (2))?

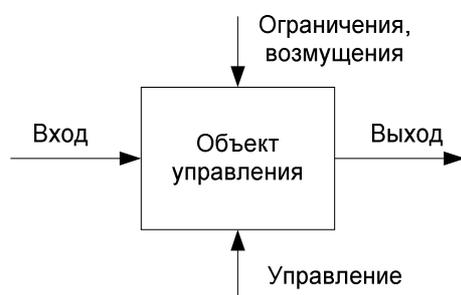


Рис. 1. «Общий вид» объекта управления

Таким образом, наконец-то можно перейти от построения формализованных и формальных моделей «к программированию», то есть к построению информационной системы (ИС), включающей вычисление критерия по модели (2). Сначала определимся с местом такой информационной системы в ходе управления некоторым объектом (рис. 1), критерием эффективности которого является (2).

Традиционным методом является включение в отрицательную обратную связь регулятора – а в случае социальных и организационных систем лица, принимающего решения (ЛПР), задачей которого является воздействие

на объект с целью приблизить значение на выходе к некоторому заданию (в организационных системах определяемого более высоким уровнем управления).

Имея начальный вариант расписания Q_1 , ЛПР пробует вносить в него какие-то изменения с целью улучшения, последовательно создавая расписания Q_2 , Q_3 и т. д. Информационная система должна *быстро* отвечать на вопрос: стало ли следующее расписание лучше (рис. 2)? Расплывчатый термин «быстро» здесь и далее означает, что общее время составления расписания, включающее многократную оценку пробных изменений в расписании, должно удовлетворять ЛПР. Следовательно, в общем случае, информационная система должна давать возможность:

1. *Быстро* вычислить значение критерия (2).
2. Для этого *быстро* определить значение всех входящих в него X_i и Y_k .
3. Для этого (в ряде случаев) *быстро* спрогнозировать изменения в объекте при помощи имитационной модели.
4. Для этого использовать накопленные в ИС данные:
 - а) как память для размещения текущих значений $Q_{ijnlmypk}$, X_i , Y_k , $\max_k (Y_k)$ и т. п.;

б) как память для оснащения правил $LDep_l = f(l)$, $r_{ij} = f(r_i, r_j)$, зависимостей наподобие $P_j = \sum_{q=1}^Q \beta_q P_q^j$ и т. п.;

в) как накопитель данных, на основании которых производятся регрессия и экстраполяция трендов при прогнозировании развития объекта;

г) как справочник для хранения множеств $\{I, J, N, L, M_i, Y_i, P, K_p, W\}$.

5. Кроме того, ИС может использоваться как хранилище удачных (или всех возможных) сценариев воздействия на объект управления [15].

Последний пункт требует пояснения. Действительно, в большинстве случаев эксперт, опираясь на существующее расписание Q_n , эвристически решает, каким образом его изменить (например, переставить местами два занятия), чтобы получилось лучшее расписание Q_{n+1} . Однако в случае ошибки эксперта (вероятность которой, по-видимому, возрастает вместе с ростом мощности множества допустимых воздействий) система начинает двигаться «в тупик», к некоторому локальному минимуму критерия, выйти из которого небольшими изменениями расписания уже не удастся. Ситуация еще более осложняется, если каждое действие эксперта требует имитационного моделирования статистического распределения значения S . Практика управления организационными системами показывает, что быстрое и сравнительно эффективное управление может быть достигнуто применением апробированных сценариев управления. В простейших случаях ситуационного управления удастся формализовать выбор сценария на основе набора правил «если – то». Но в общем случае из всего набора сценариев $C_k, k = \overline{1, K}$ эксперт должен как-то выбрать такой $C_k^* : (S(t_i) - S(t_{i+1})) \rightarrow \max$, где t_{i+1} и t_i являются границами интервала планирования, а $S(t_{i+1})$ и $S(t_i)$ рассчитываются по (2) с учетом изменения в объекте за период $[t_i, t_{i+1}]$. В нашем случае имитационный эксперимент не требуется, и надо найти $C_k^* : (S(Q_n) - S(Q_{n+1})) \rightarrow \max$.

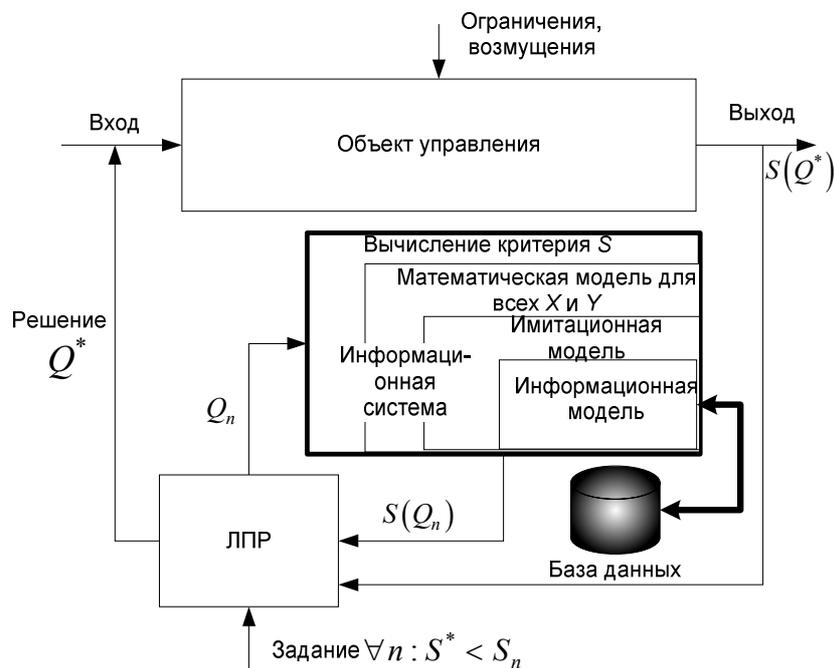


Рис. 2. Схема информационной поддержки ЛПР

Базовый набор сценариев может быть сгенерирован либо экспертами, либо путем наблюдения за работой ЛПР и реакциями системы. Роль ИС в любом случае заключается в следующем:

1. Хранить сценарии $C_k, k = \overline{1, K}$.
2. Исключать дублирующиеся сценарии $C_i = C_j \leftrightarrow i \neq j$.
3. Ранжирование сценариев (например, по частоте успешного применения).

4. Исключение сценариев:
 - а) не использовавшихся в течении времени, превышающем заданное;
 - б) приводящих к негативному воздействию на объект в течение количества попыток применения, превышающего заданное.

Алгоритм действий ЛПР и осуществляющей информационную поддержку ИС в любом случае следующий:

1. ЛПР формирует базовое решение; ИС информирует, не нарушены ли строгие ограничения. В случае составления расписания («с чистого листа») занятия одно за другим помещаются в сетку часов; интерфейс ИС, например, «подсвечивает» их зеленым или красным цветом.

2. ИС рассчитывает «базовое» значение критерия S .

3. ЛПР выбирает один из сценариев изменения расписания (эвристически или из ранжированного списка, предложенного ИС); ИС рассчитывает изменение критерия S ; интерфейс ИС информирует ЛПР:

- а) не нарушены ли вследствие применения сценария строгие ограничения;

- б) улучшилось ли значение S вследствие применения сценария.

4. Если ЛПР считает, что применен удачный сценарий, он запоминается вместо базового (одновременно ИС может изменить ранг сценария, если это предусмотрено).

5. Пункты 4 и 5 повторяются до окончания оптимизации расписания, то есть:

- а) до истечения физического времени, отведенного на решение задачи;

- б) до достижения требуемого качества расписания с точки зрения ЛПР;

- в) до достижения «гупика», из которого ЛПР не может вывести объект применением любого сценария; в последнем случае следует перейти к п. 1 и попробовать либо другое базовое решение, либо другой «путь», то есть последовательность применения сценариев.

6. Информационная модель

Как указано выше, для информационной поддержки ЛПР ИС должна хранить большое количество данных, а именно:

- учебные группы (множество L);
- преподаватели (множество J);
- дисциплины (множество N);
- аудитории (множество L) и корпуса (множество R);
- семестры (множество M);
- учебные недели (множество Y);
- дни недели (множество P);
- «звонки» – время начала каждой пары (множество K).

Для решения задачи оптимизации расписания необходимо сначала разобраться в структуре РУП¹. Он содержит список дисциплин, для каждой из которых указывается:

- к какому циклу принадлежит дисциплина: гуманитарно-социальному (ГСЭ), естественно-научному (ЕН), общепрофессиональному (ОПД), специальному (СД) или к дисциплинам специализации (ДС), которые имеются только в УП специалистов и магистров;

- к какому компоненту принадлежит дисциплина: федеральному (Ф), национально-региональному (Р) или является дисциплиной по выбору (В);

- сколько недельных часов в каждом семестре отводится на лекционные (Лк, $\nu = 1$), лабораторные (Лр, $\nu = 2$) и практические (Пр, $\nu = 3$) занятия.

Кроме того, РУП содержит рабочий график учебного процесса (календарный учебный график, КУГ), который состоит в указании количества недель Y_i в каждом семестре M на учебный процесс, на каждую практику (учебную, учебно-научную, производственную, производственно-технологическую, преддипломную и т. п.), на выполнение выпускной квалификационной работы (ВКР) и государственную итоговую аттестацию (ГИА, ранее ИГА).

¹ Использована терминология ГОС второго поколения. Однако при наблюдаемой быстрой смене версий ГОС/ФГОС принцип их формирования не меняется вплоть до ФГОС 3++, вступающих в действие с 2019 года – ГСЭ, ЕН и т. д. просто поменялись на Б1, Б2, а затем и просто на «дисциплины». Существенных изменений в информационных моделях и их использовании для поддержки решений не вызывает.

Несмотря на то, что учебный план составляется для основной образовательной программы (ООП) по конкретной специальности или направлению, на практике учебные планы время от времени корректируются в рамках одного и того же ФГОС, составляются так называемые «переходные» РУП с одной версии ФГОС на другую и т. д., поэтому рациональнее связать их с сущностью «Группа». Это дает возможность гибкой настройки учебного процесса, хотя (в идеальном случае) за это приходится платить избыточным хранением одинаковых УП Z_i^i для каждой i -й группы.

Для решения этой проблемы предлагается следующий технологический прием. Свяжем РУП как с ООП, так и с группой, и будем использовать следующий алгоритм поиска учебного плана для конкретной группы:

- если существует УП, в котором в поле «УИД группы» находится уникальный идентификатор (первичный ключ, уникальный идентификатор – далее «УИД») искомой группы, значит, это учебный план для этой группы (исключение из общего случая);
- иначе необходимо найти в таблице ООП (специальность, направление) по вторичному ключу в искомой записи в таблице «Группа» специальность, и для нее найти УП;
- если УП не найден ни для группы, ни для специальности, значит, для группы не существует УП;
- если в УП заполнены и группа, и специальность, это ошибка.

Таким образом, получим модель для хранения учебного плана, показанную на рис. 3. Она полностью нормализована, хотя и не лишена недостатков:

- все возможные значения циклов и компонентов перечислены выше и не меняются в течение действия ГОС. Следовательно, можно внести в код программы их расшифровки, удалить таблицы «Цикл» и «Компонент», а в сущности «Дисциплина» заменить их вторичные ключи на неатомарный атрибут «Код дисциплины», содержащий одновременно компонент, цикл и номер дисциплины внутри компонента и цикла, например «ЕН.Ф.02» или «ОПД.В.01.2» (второй вариант первой дисциплины по выбору в цикле ОПД). Это нарушает нормализацию, но поскольку порядок следования циклов и компонентов predetermined выше и никогда не меняется, а «разобрать» этот неатомарный атрибут не представляет труда, пользоваться таблицами станет удобнее;
- количество семестров M_i также никогда не превышает 12. Это дает возможность денормализовать учебный план, чтобы ускорить работу ИС.

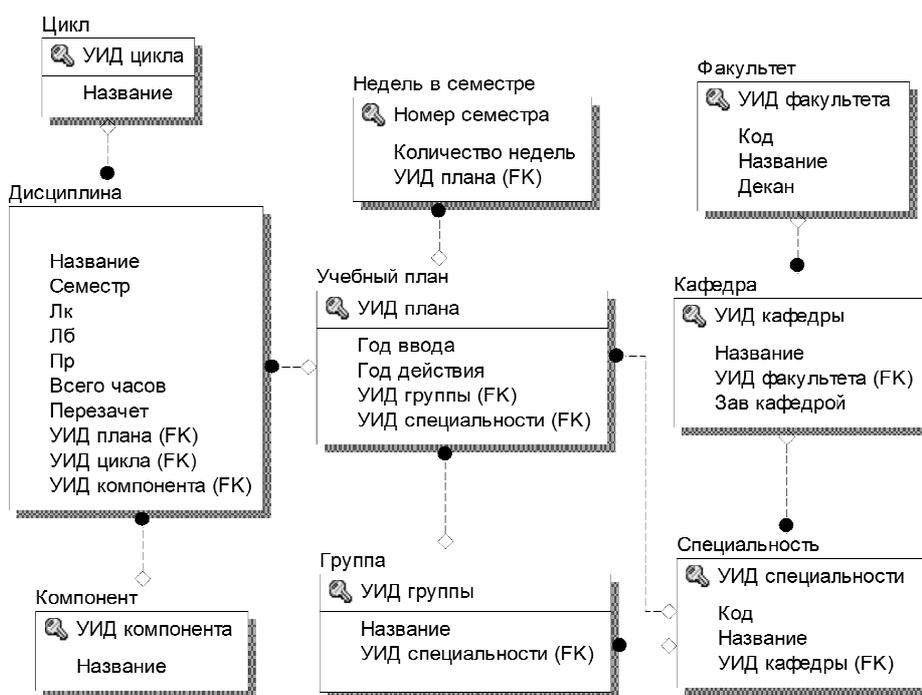


Рис. 3. Информационная модель учебного плана

Управление в социально-экономических системах

На основе учебного плана производится распределение нагрузки между преподавателями. При этом используются перечисленные выше справочники дисциплин (N), групп (I), кафедр ($Ldep$), преподавателей (J), учебных планов (Z_1). Остается создать таблицу, содержащую в себе информацию о распределенной нагрузке (W). В результате получим информационную модель (рис. 4).

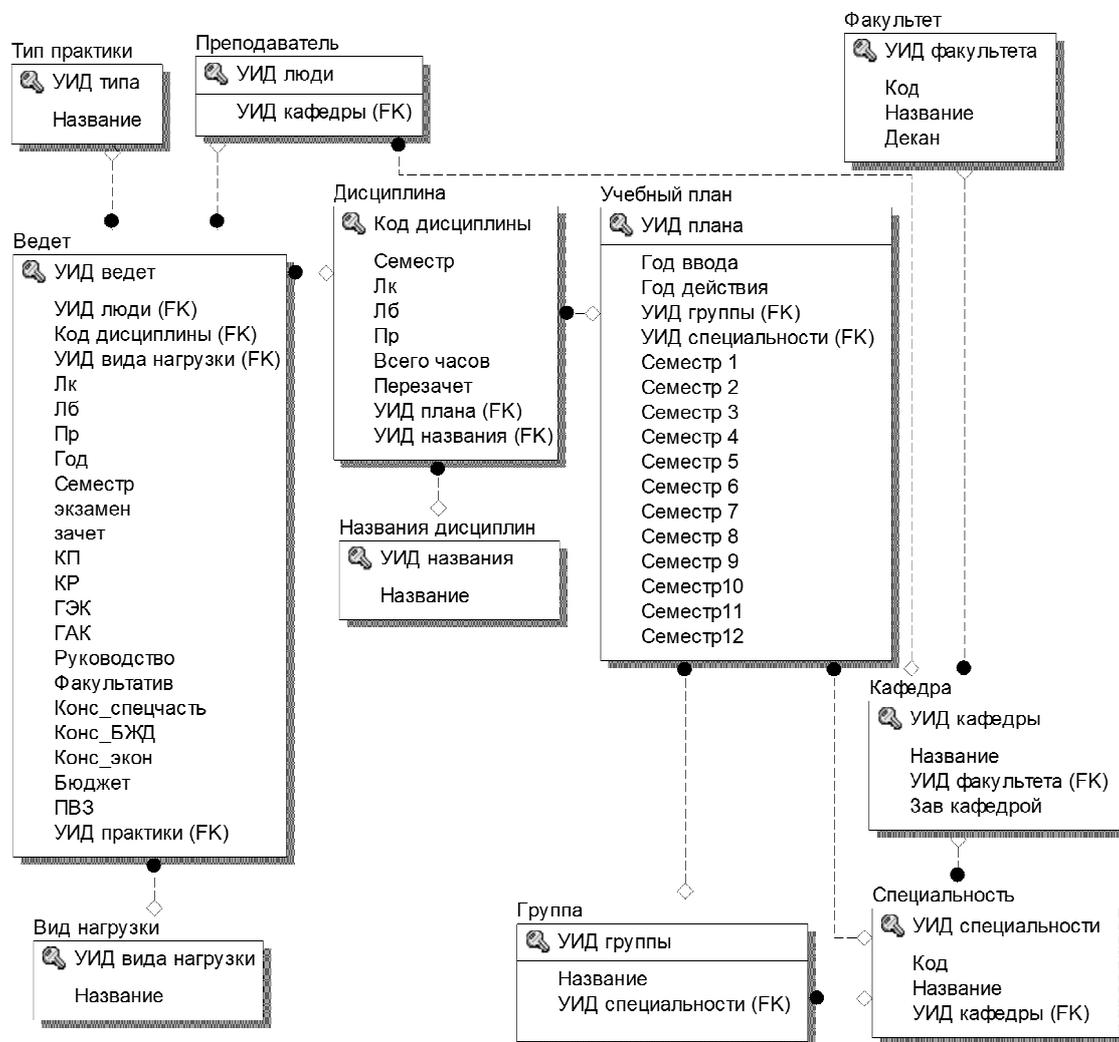


Рис. 4. Информационная модель распределения нагрузки

Основной сущностью является таблица «Ведет». В ней хранится информация о распределенной нагрузке. Таблица «Учебный план» хранит количество семестров M_i для каждой i -й группы и их продолжительность в неделях Y_i (если группа учится меньше 12 семестров, начиная с какого-либо семестра, указывается нулевая продолжительность).

В отличие от предыдущей модели, предусмотрена таблица «Названия дисциплин» (для хранения множества N): ведь дисциплина «Информатика» содержится во многих учебных планах, и такой прием позволяет как унифицировать названия, так и уменьшить объем таблицы «Дисциплина». Для решения задач, связанных с хранением и оптимизацией расписания, построим несколько вариантов информационной модели, опираясь на модель нагрузки (несущественные в данном случае атрибуты и сущности далее опущены).

Вариант 1 (рис. 5) является наиболее приемлемым при соблюдении всех правил проведения учебного процесса. Денормализующие поля «Начало» и «Окончание» позволяют быстрее вывести разные расписания без обращения к таблице «Звонки» (K_p). Однако такая модель не позволяет, например, следующее:

- назначить лабораторные работы преподавателю (или учебному мастеру), если их нет в нагрузке – однако такое бывает необходимо при организации отработки лабораторных работ;
- планировать факультативные (отсутствующие в нагрузке) и сборные занятия (например, почти все студенты групп занимаются английским языком у нескольких преподавателей, а несколько человек приходят на занятие немецким языком);
- планировать исключительные ситуации, наподобие собрания нескольких групп;
- затрудняет контроль допустимости расписания. Например, сложно отличить потоковое занятие (несколько групп в одной аудитории занимаются предметом с одним названием – заметим, что УИДы предметов в УП разных групп будут разными!) от ошибочного совмещения двух групп в одной аудитории в одно время.

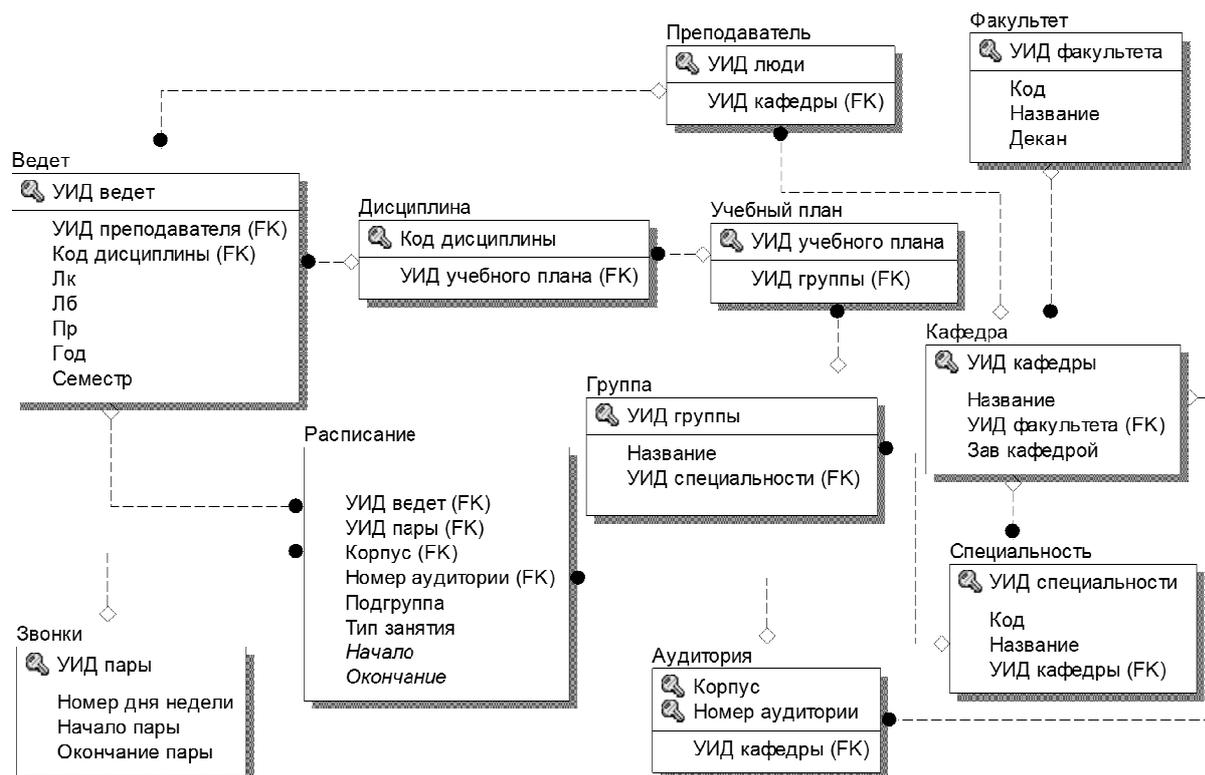


Рис. 5. Вариант № 1 информационной модели расписания

Универсального способа решения данных проблем, вероятно, не существует: каждый подход будет иметь свои плюсы и минусы. Более-менее универсальным методом, как и в любой ситуации, является усложнение сущности «Расписание» денормализующими атрибутами, позволяющими отслеживать исключительные ситуации. Естественно, одновременно разработчик должен предусматривать все более сложные и объемные алгоритмы проверки, позволяющие контролировать непротиворечивость значений исходных и денормализованных полей.

Один из реалистичных примеров модели, так или иначе решающей эти проблемы, приведен на рис. 6.

Направлениями дальнейшего развития модели являются, например:

- формирование заявления на оплату преподавателя-почасовика;
- формирование сводки остатка часов по дисциплине;
- учет неотработанных часов;
- учет тематики занятий, контроль ее соответствия рабочей учебной программе.

Наконец, она дает возможность подойти к решению задачи оценки качества расписания с учетом пожеланий преподавателей $(Y_k)_j$ и их сравнительной важности P_j , а также других нестрогих ограничений, перечисленных в разделе «Критерий качества» (рис. 7).

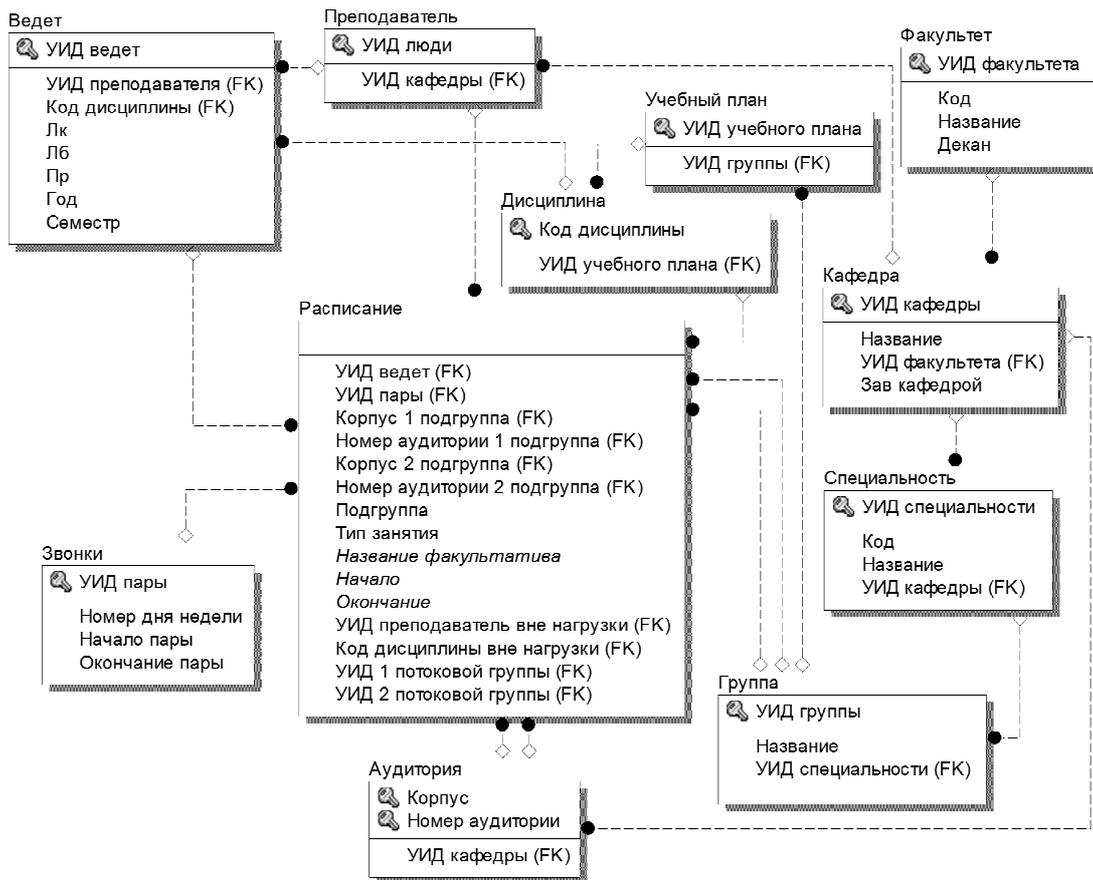


Рис. 6. Вариант № 2 информационной модели расписания

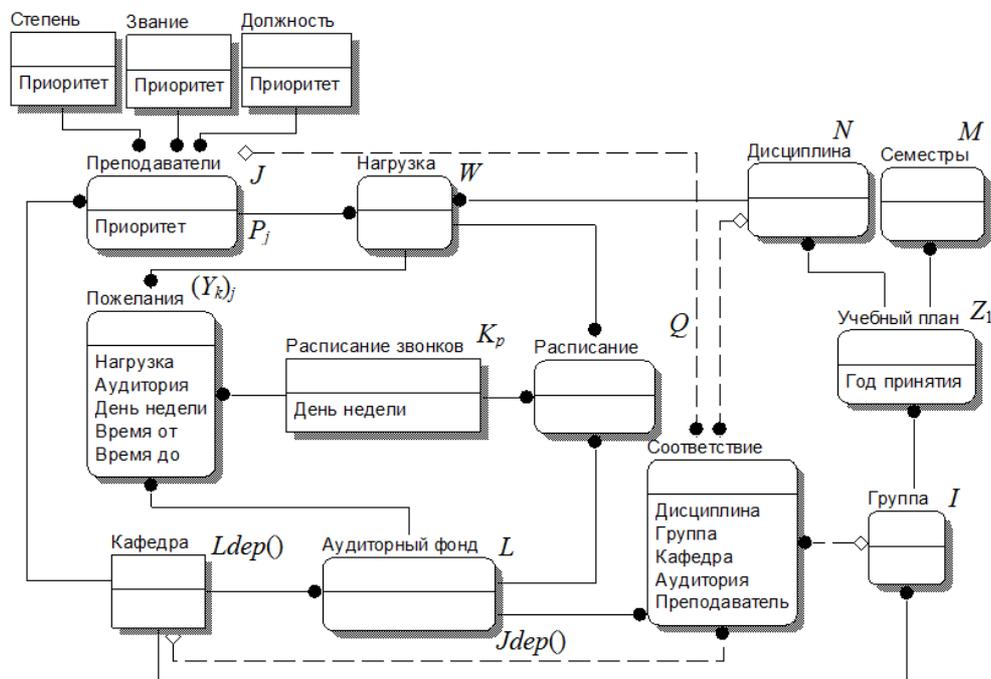


Рис. 7. Фрагмент информационной модели для оценки качества расписания на основе соответствия аудиторному фонду, пожеланий преподавателей и их приоритета

Для наглядности на рис. 7 накопители промаркированы символами множеств, элементы которых в них хранятся.

Заключение

Приведенный перечень математических и структурных моделей не является исчерпывающим, но дает представление о сложности и вариативности решений задач, связанных с автоматизацией всего одного процесса в деятельности организационной системы. Это иллюстрирует, что в сложных предметных областях может существовать множество решений задач моделирования, обладающих сопоставимыми достоинствами и недостатками. Выбор конкретного варианта для реализации в этом случае производится исключительно на основе опыта разработчика и, как показывает практика, далеко не всегда оказывается подходящим при расширении функционального охвата ИС. Поэтому очень важной в таких случаях является оценка возможностей «роста» информационной модели вслед за внутренними изменениями вуза или изменениями внешних условий и возможностей следовать за этим ростом без изменения хотя бы основных ключевых атрибутов и связей между ними.

Рассмотренный метод связывания математических, имитационных и информационных моделей с целью улучшения качества принимаемых управленческих решений показывает «место в жизни» для современного информатика. Ведь большинство учетных задач за последние 10–15 лет решено, а попасть в разработку крупной корпоративной ИС или информационной системы удается далеко не каждому. В то же время множество организационных систем (часть которых перечислена выше) ожидают решения проблемы повышения эффективности их деятельности или же отдельных, важных организационных процессов.

Литература

1. Варламова, С.А. Информационная поддержка принятия решений при управлении филиалом вуза / С.А. Варламова, А.В. Затонский, Е.В. Измайлова. – М.: ИНФРА-М, 2014. – 333 с.
2. Васильев, Ф.П. Методы оптимизации / Ф.П. Васильев. – М.: Факториал-Пресс, 2002. – 824 с.
3. Володина, Ю.И. Модели и методы информационной поддержки управления транспортным обслуживанием населения больших и средних городов / Ю.И. Володина // Современная наука: актуальные проблемы теории и практики. Серия: Естественные и технические науки. – 2016. – № 1. – С. 33–43.
4. Затонский, А.В. Информационные технологии. Разработка информационных моделей и систем / А.В. Затонский. – М.: ИНФРА-М, 2014. – 344 с.
5. Затонский, А.В. Оптимизация модели информационной системы поддержки техобслуживания и ремонта оборудования / А.В. Затонский // Информационные технологии. – 2007. – № 3. – С. 2–7.
6. Иванова, Е.В. Методы формализации самооценки на примере научно-исследовательской работы студентов / Е.В. Иванова, А.В. Затонский // Информатизация образования и науки. – 2011. – № 11. – С. 110–116.
7. Измайлова, Е.В. Исследование критерия оптимальности модели системы управления научно-исследовательской работой студентов вуза / Е.В. Измайлова // Новый университет. Серия «Технические науки». – 2011. – № 5 – С. 3–5.
8. Куцелап, К.А. Составление производственного расписания с использованием алгоритма направленного случайного поиска / К.А. Куцелап, В.П. Вороненко, А.Э. Шалдов // Известия Тульского государственного университета. Технические науки. – 2015. – № 12-1. – С. 14–23.
9. Клеванский, Н.Н. Формирование расписания занятий высших учебных заведений / Н.Н. Клеванский // Образовательные ресурсы и технологии. – 2015. – № 1. – С. 34–44.
10. Клоков, И.В. Бизнес-план на компьютере / И.В. Клоков. – СПб.: Питер, 2007. – 176 с.
11. Рутковская, Д. Нейронные сети, генетические алгоритмы и нечеткие системы / Д. Рутковская, М. Пилиньские, Л. Рутковский. – М.: Горячая линия – Телеком, 2006. – 452 с.
12. Goal solver: a hybrid local search based solver for high school timetabling / G.H.G. Da Fonseca, H.G. Santos, T.A.M. Toffolo et al. // Annals of Operations Research. – 2014. – P. 1–21.
13. Skoullis, V.I. Solving the high school timetabling problem using a hybrid cat swarm optimization based algorithm / V.I. Skoullis, I.X. Tassopoulos, G.N. Beligiannis // Applied Soft Computing. – 2017. – Vol. 52. – P. 277–289. DOI: 10.1016/j.asoc.2016.10.038
14. Raghavjee, R. A genetic algorithm selection perturbative hyper-heuristic for solving the school

timetabling problem / R. Raghavjee, N. Pillay // *ORiON*. – 2015. – No. 31. – P. 39–60. DOI: 10.5784/31-1-158

15. *A simulated annealing with a new neighborhood structure based algorithm for high school timetabling problems* / D. Zhang, Y. Liu, R. M'Hallah, C.H.S. Leung // *Eur. J. Oper. Res.* – 2010. – No. 203. – P. 550–558.

16. *Ahmed, L.N. Solving high school timetabling problems worldwide using selection hyper-heuristics* / L.N. Ahmed, E. Özcan, A. Kheiri // *Expert Syst. Appl.* – 2015. – No. 42. – P. 5463–5471. DOI: 10.1016/j.eswa.2015.02.059

17. *A computational study of local search algorithms for Italian high-school timetabling* / P. Avella, B. D'Auria, S. Salerno, I. Vasilâev // *J. of Heuristics*. – 2007. – No. 13. – P. 543–556. DOI: 10.1007/s10732-007-9025-3

18. *Tassopoulos, I.X. Using particle swarm optimization to solve effectively the school timetabling problem* / I.X. Tassopoulos, G.N. Beligiannis // *Soft Comput.* – 2012. – No. 16. – P. 1229–1252. DOI: 10.1007/s00500-012-0809-5

19. *Bouzidi, A. Discrete at swarm optimization to resolve the traveling salesman problem* / A. Bouzidi, M.E. Riffi // *Int. J. Adv. Res. Comp. Sci. Softw. Eng.* – 2013. – No. 3. – P. 13–18.

20. *Beligiannis, G.N. A genetic algorithm approach to school timetabling* / G.N. Beligiannis, C.N. Moschopoulos, S.D. Likothanassis // *J. Oper. Res. Soc.* – 2009. – No. 60. – P. 23–42.

Затонский Андрей Владимирович, д-р техн. наук, профессор, зав. кафедрой автоматизации технологических процессов, Пермский национальный исследовательский политехнический университет, Березниковский филиал, г. Березники, Пермский край; z xenon@narod.ru.

Варламова Светлана Александровна, канд. техн. наук, доцент кафедры автоматизации технологических процессов, Пермский национальный исследовательский политехнический университет, Березниковский филиал, г. Березники, Пермский край; varlamovasa@mail.ru.

Поступила в редакцию 16 апреля 2018 г.

DOI: 10.14529/ctcr180310

INFORMATION BASIS OF DECISION SUPPORT SYSTEM FOR HIGH SCHOOL TIMETABLE CREATION AS AN EXAMPLE

A.V. Zatonkiy, z xenon@narod.ru,
S.A. Varlamova, varlamovasa@mail.ru

Perm National Research Politechnic University, Berezniki branch, Berezniki, Perm region, Russian Federation

This article is about a task of learning of effective use of information technologies with the decision support. The criteria for evaluating the activity of economic and social systems are systematized, their classification and areas of application of particular criteria of different nature are given. It is proposed to integrate knowledge in the field of information technologies in order to decision support on example of an actual task of scheduling lessons. The initial data are determined, the factors of the model are classified, a mathematical model is designed, its limitations are proposed. Methods for obtaining schedule quality criteria from various points of view and integral quality criterion design are presented. The limitations of possible solutions are considered and classified. Some examples of formula criteria unusability are given. The selection of optimization algorithms based on an integral criterion is substantiated. The transition from structural models of the activity of decision makers to information models of decision support systems is shown. The concept of an expert decision support is offered by an estimation of its difficultly formalized actions on change of para-

metric values of the system. Sets of objects of the system are compared to the essences of the information model, and its improving transformations are given. The variants of the information model with different degrees of normalization are considered; their comparative advantages and disadvantages are shown. The directions of development of the information model in the structure of the automated management system of the university are determined. As a conclusion a possibility of increasing the efficiency of the activities of complex socio-economic systems through the rational use of mathematical, information and simulation tools is proved. It is shown that this area of activity is currently promising from the point of view of the effective application of the efforts of a specialist in the field of informatics.

Keywords: high school, automation, decision support.

References

1. Varlamova S.A., Zatonskiy A.V., Izmaylova E.V. *Informatsionnaya podderzhka prinyatiya resheniy pri upravlenii filialom vuza* [Informational Support of a Decision Making for Management of Branch of Higher Education Institution]. Moscow, INFRA-M Publ., 2014. 333 p.
2. Vasil'ev F.P. *Metody optimizatsii* [Optimization Method]. Moscow, Faktorial-Press Publ., 2002. 824 p.
3. Volodina Yu.I. [Models and Methods of Informational Support of Management of a Transport Upkeep of the Population of the Big and Average Cities]. *The Modern Science: Current Problems of the Theory and Practice. Series: Natural and Technical Science*, 2016, no. 1, pp. 33–43. (in Russ.)
4. Zatonskiy A.V. *Informatsionnye tekhnologii. Razrabotka informatsionnykh modeley i sistem* [Informational Technologies. Development of Informational Models and Systems]. Moscow, INFRA-M Publ., 2014. 344 p.
5. Zatonskiy A.V. [Optimization of Model of an Information System of Support of Servicing and Repair of an Inventory]. *Informational Technologies*, 2007, no. 3, pp. 2–7. (in Russ.)
6. Ivanova E.V., Zatonskiy A.V. [Methods of Formalization of a Self-Rating on the Example of Research of Students]. *Informatization of Science and Education*, 2011, no. 11, pp. 110–116. (in Russ.)
7. Izmaylova E.V. [Research of Criterion of Optimality of Model of a Control System of Research of Students of Higher Education Institution]. *New University. Series "Technical Science"*, 2011, no. 5, pp. 3–5. (in Russ.)
8. Kutselap K.A., Voronenko V.P., Shaldov A.E. [Drawing up the Production Schedule with Use of an Algorithm of a Directional Random Search]. *News of the Tula State University. Technical Science*, 2015, no. 12-1, pp. 14–23. (in Russ.)
9. Klevanskiy N.N. [Formation of the Lesson Schedule of Higher Educational Institutions]. *Educational Resources and Technologies*, 2015, no. 1, pp. 34–44. (in Russ.)
10. Klokov I.V. *Biznes-plan na komp'yutere* [The Business Plan on the Computer]. St.Petersburg, Piter Publ., 2007. 176 p.
11. Rutkovskaya D., Pilin'skie M., Rutkovskiy L. *Neyronnye seti, geneticheskie algoritmy i nechetkie sistemy* [Neuronic Networks, Genetic Algorithms and Indistinct Systems]. Moscow, Goryachay liniya-Telekom Publ., 2006. 452 p.
12. Da Fonseca G.H.G., Santos H.G., Toffolo T.A.M., Brito S.S., Souza M.J.F. Goal Solver: a Hybrid Local Search Based Solver for High School Timetabling. *Annals of Operations Research*, 2014, pp. 1–21.
13. Skoullis V.I., Tassopoulos I.X., Beligiannis G.N. Solving the High School Timetabling Problem Using a Hybrid Cat Swarm Optimization Based Algorithm. *Applied Soft Computing*, 2017, vol. 52, no. 3, pp. 277–289. DOI: 10.1016/j.asoc.2016.10.038
14. Raghavjee R., Pillay N. A Genetic Algorithm Selection Perturbative Hyper-heuristic for Solving the School Timetabling Problem. *ORiON*, 2015, no. 31, pp. 39–60. DOI: 10.5784/31-1-158
15. Zhang D., Liu Y., M'Hallah R., Leung C.H.S. A Simulated Annealing with a New Neighborhood Structure Based Algorithm for High School Timetabling Problems. *Eur. J. Oper. Res.*, 2010, no. 203, pp. 550–558.
16. Ahmed L.N., Özcan E., Kheiri A. Solving High School Timetabling Problems Worldwide Using Selection Hyper-Heuristics. *Expert Syst. Appl.*, 2015, no. 42, pp. 5463–5471. DOI: 10.1016/j.eswa.2015.02.059

17. Avella P., D'Auria B., Salerno S., Vasilâev I. A Computational Study of Local Search Algorithms for Italian High-School Timetabling. *J. of Heuristics*, 2007, no. 13, pp. 543–556. DOI: 10.1007/s10732-007-9025-3
18. Tassopoulos I.X., Beligiannis G.N. Using Particle Swarm Optimization to Solve Effectively the School Timetabling Problem. *Soft Comput.*, 2012, no. 16, pp. 1229–1252. DOI: 10.1007/s00500-012-0809-5
19. Bouzidi A., Riffi M.E. Discrete at Swarm Optimization to Resolve the Traveling Salesman Problem. *Int. J. Adv. Res. Comp. Sci. Softw. Eng.*, 2013, no. 3, pp. 13–18.
20. Beligiannis G.N., Moschopoulos C.N., Likothanassis S.D. A Genetic Algorithm Approach to School Timetabling. *J. Oper. Res. Soc.*, 2009, no. 60, pp. 23–42.

Received 16 April 2018

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Затонский, А.В. Информационное обеспечение поддержки принятия решений на примере составления расписания занятий образовательной организации / А.В. Затонский, С.А. Варламова // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». – 2018. – Т. 18, № 3. – С. 88–106. DOI: 10.14529/ctcr180310

FOR CITATION

Zatonskiy A.V., Varlamova S.A. Information Basis of Decision Support System for High School Timetable Creation as an Example. *Bulletin of the South Ural State University. Ser. Computer Technologies, Automatic Control, Radio Electronics*, 2018, vol. 18, no. 3, pp. 88–106. (in Russ.) DOI: 10.14529/ctcr180310