

ОБОБЩЁННАЯ МАТЕМАТИЧЕСКАЯ ПОСТАНОВКА ЗАДАЧИ ОПТИМИЗАЦИИ РАЗМЕЩЕНИЯ ДАННЫХ В ИНФОРМАЦИОННОЙ СИСТЕМЕ ТРЕНАЖЁРНО-МОДЕЛИРУЮЩЕГО КОМПЛЕКСА

В.В. Янюшкин

ГОУ ВПО «Южно-Российский государственный технический университет (Новочеркасский политехнический институт)», г. Новочеркасск

Ключевые слова и фразы: математическая модель; многоуровневая система; распределение модельного мира.

Аннотация: На основе анализа классических архитектур и алгоритмов размещения данных в информационных средах предлагается обобщённая математическая постановка задачи размещения данных и алгоритм её решения, который отличается более высоким уровнем абстракции исходных данных и широким применением в сфере распределённых информационных систем различного назначения.

Цель данной статьи – создание универсального алгоритма размещения данных на уровне аппаратной конфигурации любой корпоративной информационной системы с учетом её архитектуры и ограничений доступа [1]. Аналогично проектирование распределённых тренажерно-моделирующих комплексов (ТМК) и наличие сложных зависимостей исходных данных от характера функционирования отдельных моделей определяет проблему состава данных модельного мира на узлах различной архитектуры и динамики обработки информации.

Анализ моделей и алгоритмов распределения данных в распределённой информационной системе выявил следующие положения:

1. Понятие модели размещения данных на концептуальном уровне уходит от понятия типа данных и постепенно типовая классификация трансформируется в безтиповое определение объектов модельного мира, при этом само понятие типа данных является второстепенным фактором для обеспечения стратегии размещения. В основу данного утверждения положены алгоритмы размещения, которые для оптимального распределения оперируют отдельными объектами модельного мира в зависимости от потребностей моделей.

2. Выделение на уровне аппаратной конфигурации иерархии уровней размещения данных, при этом в стандартной схеме существует всего один уровень (аппаратной конфигурации ТМК), тогда как в схеме облачных вычислений возможности размещения данных значительно шире и представлены облаком данных и удалёнными клиентами. Таким образом, каждая специфика архитектуры построения системы располагает своим набором, который обобщённо сводится к иерархии типов памяти уровней вычислительной системы, а задача размещения данных к задаче распределения файлов по уровням памяти (рис. 1 – самый высокий уровень определяет наивысшую скорость).



Рис. 1. Обобщение иерархии уровней размещения данных

3. Потребности моделей определяются пересекающимися множествами элементов модельного мира различных типов, поэтому частым приёмом в математических выкладках является оперирование понятиями теории множеств, основанными на выделении подмножества, размещённых на конкретном узле системы объектов множества определённого типа.

4. Алгоритм оптимизации размещения данных основывается на эффективной начальной стратегии размещения, которая предлагает приближённый способ, корректирующийся в процессе работы ГА с учётом введённых ограничений. Качество начальной популяции определяется именно логикой работы данного алгоритма, генетические операторы в данном случае являются удобным инструментом случайного поиска оптимального решения из всего множества потенциально возможных.

5. Распределение исполняемых моделей, которые являются основными потребителями информации модельного мира в системе, производится преимущественно на специализированных вычислительных узлах, распределённых по иерархии уровней. Остальные потребители балансируют нагрузку по поддержке динамически изменяющихся тиражированных данных или существенных временных задержках удалённого доступа.

6. Концепция задачи размещения данных модельного мира сводится к понятиям иерархии уровней и набора объектов модельного мира, которые должны быть размещены на каждом уровне в рамках назначения и функциональности уровня конфигурации системы [2]. При этом рациональное размещение ведётся не только в рамках всей иерархии уровней, но и каждого уровня отдельно, образуя, таким образом, многоуровневую многокритериальную задачу оптимизации.

Обобщённая математическая постановка задачи размещения данных

Задан структурированный состав модельного мира, иерархия уровней информационной системы с функциональным наполнением каждого уровня. Задача состоит в нахождении оптимального размещения сущностей модельного мира в распределённой информационной системе для обеспечения максимальной эффективности при соблюдении технических и программных ограничений.

К техническим ограничениям относятся допустимые размеры оперативной и дисковой памяти, количество компьютеров каждого уровня, скорости обмена данными при принятых проектных решениях между отдельными элементами системы. К программным ограничениям относятся функции поддержки целостности модельного мира и отдельных его объектов, максимально допустимые задержки получения данных отдельными моделями и затраты на использование той или иной технологии и протокола обмена. Под эффективностью понимается выполнение функциональности всех узлов системы с учётом текущих ограничений и минимальными затратами оперативной, дисковой памяти, процессорного времени и межмашинного трафика.

$L^i = (L_1^i, \dots, L_n^i)$, $i = \overline{1, num_level}$ – множество элементов информационной системы уровня иерархии i ;

num_level – общее число уровней иерархии информационной системы;

n^i – количество элементов уровня иерархии i ;

L_j^i – определённый элемент иерархии i с порядковым номером j ;

$V^i = (V_1^i, \dots, V_m^i)$, $i = \overline{1, num_level}$ – объёмы памяти для распределения модельного мира на уровне элементов иерархии i ;

$M^i = (M_1^i, \dots, M_{p^i}^i)$, $i = \overline{1, num_level}$ – набор моделей функционального наполнения уровня i ;

p^i – количество моделей, расположенных на уровне иерархии i ;

$MC_{k,l}^i$, $k = \overline{1, n^i}$, $l = \overline{1, p^i}$ – матрица расположения функциональных модулей по элементам уровней иерархии, элемент матрицы может принимать следующие значения:

$$MC_{k,l}^i = \begin{cases} 1, & \text{если модель расположена на } k \text{ элементе уровня } i \\ 0, & \text{в противном случае} \end{cases}$$

VD_j^i – объём объекта модельного мира типа $i = \overline{1, m}$ с индексом $j = \overline{1, n_data^i}$;

m – количество типов данных модельного мира;

n_data^i – количество объектов модельного мира типа i ;

$$MD_{k,l}^{i,j}, k = \overline{1, p^i}, j = \overline{1, m}, l = \overline{1, n_data^i}$$

матрица потребности моделей уровня i в объектах модельного мира, элемент матрицы может принимать следующие значения:

$$MD_{k,l}^{i,j} = \begin{cases} 1, k \text{ модели уровня } i \text{ нужен объект } l \text{ типа } j \text{ (режим чтения)}, \\ 2, k \text{ модели уровня } i \text{ нужен объект } l \text{ типа } j \text{ (режим записи)}, \\ 0, \text{ в противном случае.} \end{cases}$$

$$MDI_{k,l}^{i,j}, k = \overline{1, p^i}, j = \overline{1, m}, l = \overline{1, n_data^i}$$

матрица интенсивности использования моделей уровня i в объектах модельного мира;

$T_{j,num}^i = (T_{1,num}^i, \dots, T_{n_data^i,num}^i)$, $i = \overline{1, num_level}$ – максимальное время получения входных данных моделью j уровня i ;

$T_{j,mod}^i = (T_{1,mod}^i, \dots, T_{n_data^i,mod}^i)$, $i = \overline{1, num_level}$ – время обработки данных моделью j уровня i ;

$$T^{num} = (T_1^{num}, \dots, T_{num_level}^{num})$$

усреднённое время предоставления данных уровнем иерархии $i = \overline{1, num_level}$;

T^{num} – время получения данных уровнем при условии того, что они расположены за пределом

уровня i и на любом из анализируемых уровней T_j^{num} , $j = \overline{1, num_level}$ они гарантированно присутствуют. Альтернативным вариантом является вычисление времени подъёма на уровни $(i-1, \dots, 1)$, а в случае неудачи нахождения там необходимых данных, спуска на уровни $(i+1, \dots, num_level)$, как показано на рисунке 2.

Т.о. время для уровня i вычисляется по формуле:

$$T^{i,num} = T_i^{num} + \sum_{j=1}^i T_j^{num}$$

где s – индекс первого, начиная с 1 уровня иерархии с возможностью ответа на запрос уровня i .

Альтернативными вариантами является исследование окрестных уровней $(i+1)$, $(i-1)$, $(i+2)$, $(i-2)$ и т.д., т.к. стоимость взаимодействия уровней i и $(i-1)$, например, может быть ниже стоимости взаимодействия уровней i и s , т.к. при этом минует серия промежуточных уровней между s и i . Таким образом, в общей постановке поиск подходящих уровней взаимодействия также является оптимизационной задачей, в данном виде сводящейся к модифицированной задаче коммивояжёра (рис. 3).

Существует следующее концептуальное соответствие: каждый город маршрута представлен уровнем иерархии (число городов соответствует числу num_level); полный маршрут коммивояжёра проходит через все города, начиная с 1 и заканчивая num_level ; длина маршрута составляет сумму времён получения данных с каждого уровня T^{num} ; матрица расстояний между городами-уровнями представляется суммами затрат на связь соответствующих уровней.

Отличие задачи состоит в следующем: изначально коммивояжёр стоит в точке i маршрута (источник запросов), при этом на маршруте существует ряд базовых городов – уровней, где существует потенциальная возможность ответа на запрос (на уровень размещены соответствующие объекты модельного мира).

Задача состоит в том, чтобы коммивояжёру можно было бы достичь одного из базовых городов, используя минимальный маршрут. Иными словами, уровень i должен получить данные с любого уровня, обладающего необходимыми структурами, за счёт минимальных временных задержек

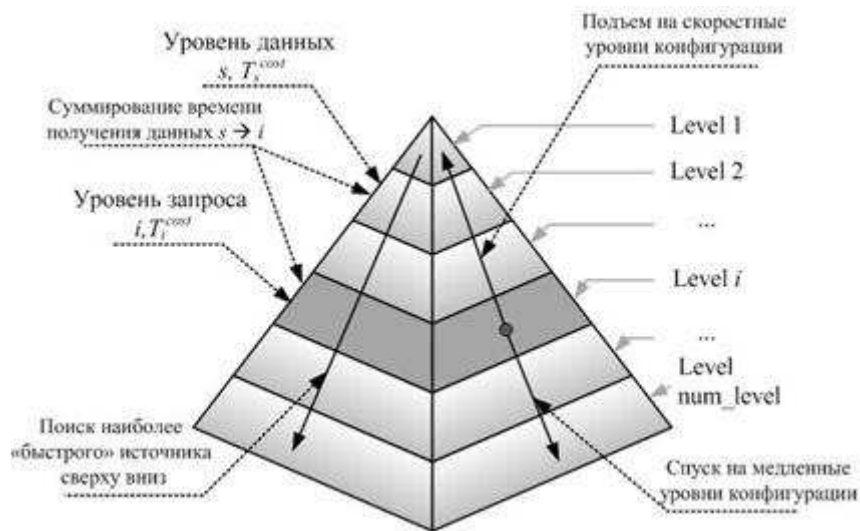


Рис. 2. Варианты вычисления затрат на связь между уровнями

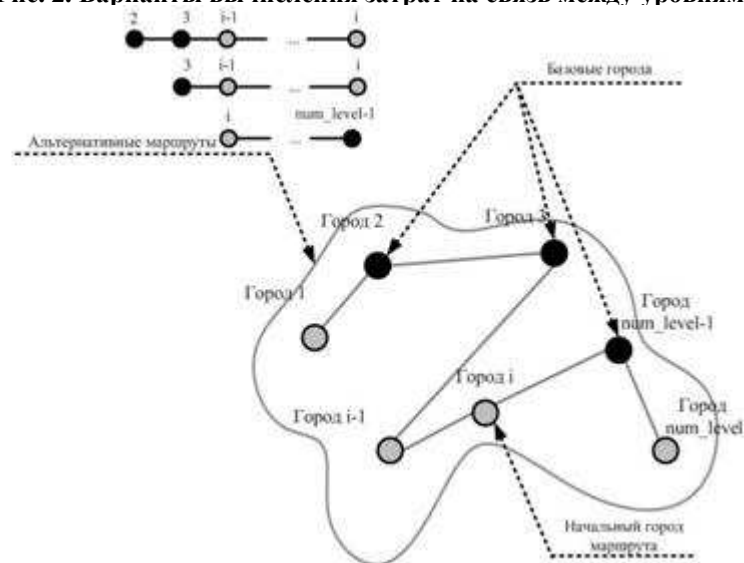


Рис. 3. Графическая интерпретация задачи коммивояжёра

создаваемых суммированием числа промежуточных уровней T_i^{mult} .

$T_{k,l}^{i,j}$, $k=1, \overline{p^i}$, $j=1, \overline{m}$, $l=1, \overline{n_data^i}$ – время обработки входных данных модели k , расположенной на уровне i по всем объектам модельного мира, элемент матрицы может принимать следующие значения:

$$T_{k,l}^{i,j} = \begin{cases} VD_k^i * T_i^{mult}, & k \text{ модели нужен объект } VD_k^i \\ 0, & \text{иначе} \end{cases}$$

Критерий эффективности задачи F состоит в поиске минимальной конфигурации расположения объектов модельного мира в системе иерархий уровней:

$$F = \min \left(\sum_{i=1}^{num_level} \left(\sum_{k=1}^{p^i} T_{k,l}^{i,j} + \sum_{\substack{l=1, \overline{n_data^i} \\ \substack{MDI_{k,l}^{i,j} \\ \substack{MDI_{k,l}^{i,j} \\ \substack{MDI_{k,l}^{i,j}}}}}} \right) + \sum_{\substack{l=1, \overline{n_data^i} \\ \substack{MDI_{k,l}^{i,j} \\ \substack{MDI_{k,l}^{i,j} \\ \substack{MDI_{k,l}^{i,j}}}}}} \right) + A$$

где A – дополнительные затраты на запись и изменение данных на других уровнях системы:

$$A = \sum_{i=1}^{num_level} \sum_{\substack{l=1, \overline{n_data^i} \\ \substack{MDI_{k,l}^{i,j} \\ \substack{MDI_{k,l}^{i,j} \\ \substack{MDI_{k,l}^{i,j}}}}}} VD_k^i * MDI_{k,l}^{i,j} * T_{k,l}^{i,j}$$

при выполнении ограничений: ограничение 1 – суммарный объём модельного мира, помещённый на функциональный элемент f уровня иерархии i не должен превышать заранее установленного ограничения:

$$\sum_{i=1}^n \sum_{f=1}^{n'} VD_i^f \leq V_i^f, \quad i = \overline{1, \text{min_level}}, \quad f = \overline{1, n'}$$

ограничение 2 – время получения входных данных моделью j , расположенной на уровне i не должно превышать заранее заданного предела:

$$\sum_{i=1}^n \sum_{j=1}^{p'} T_{ij}^f \leq T_{max}^f, \quad i = \overline{1, \text{min_level}}, \quad j = \overline{1, p'}$$

Данное ограничение охватывает все возможные модели на всех уровнях системы, в том числе и потенциальное присутствие на одном из внешних уровней (или как составной части системы) модели оператора-человека с индивидуальной степенью реакции на события и отклик системы.

Такие условия, как целостность модельного мира и отдельных типов данных, в данной постановке задачи не проверяются, т.к. предполагается наличие уровней системы (на нижних уровнях иерархии), заранее хранящих весь возможный состав модельного мира, к примеру, в архитектуре облачных вычислений данная функция возложена на системы хранения (*Data Storage*).

Концепция обобщённого алгоритма решения задачи размещения данных

Если принять матрицы размещения моделей по уровням системы, потребности в объектах модельного мира и интенсивности обращений как исходную информацию задача ставится в создании оптимальной делённой структуры модельного мира в соответствии с введённым критерием эффективности и принятыми ограничениями математической постановки задачи.

Распределённая информационная система рассматривается как многокритериальная оптимизационная задача расположения на уровнях иерархии объектов модельного мира, позволяющих оптимизировать потоки запросов по критерию времени в рамках функционирования на уровнях моделей (рис. 4), при этом основное внимание уделяется критерию скорости выполнения запросов, как основному ограничению для систем реального времени.

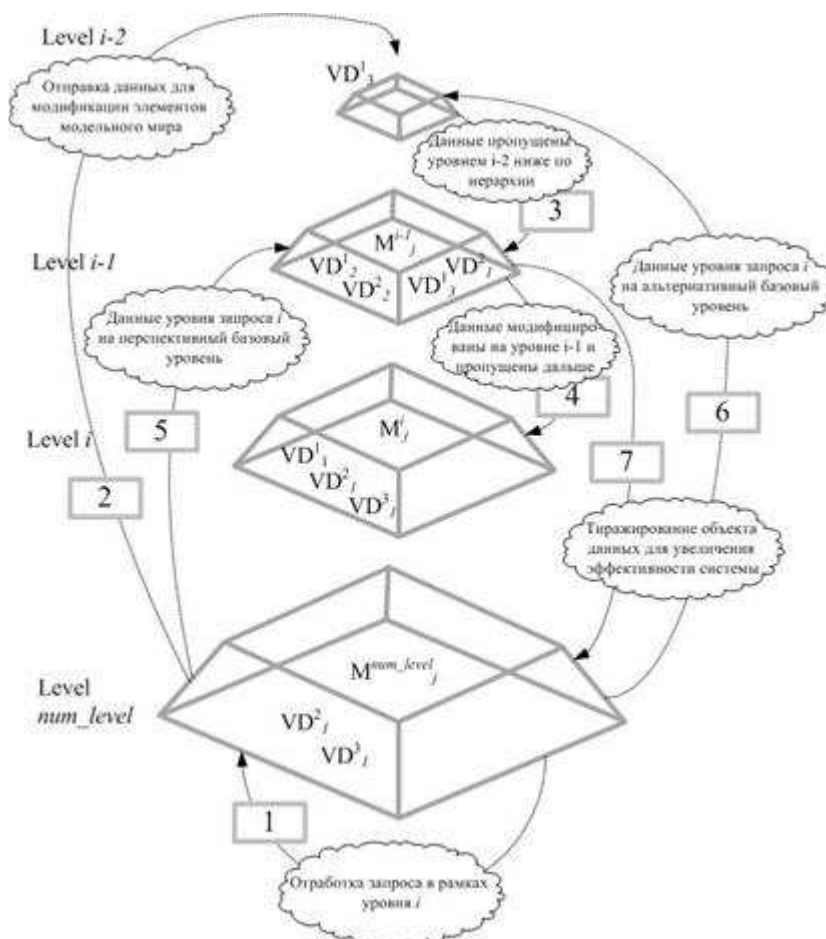


Рис. 4. Концепция алгоритма решения задачи размещения данных

Как видно из приведённого рисунка, взаимодействие уровней системы осуществляется в контексте потребностей моделей каждого уровня некоторых объектов модельного мира. Поток 1 отображает обработку запросов модели $M_j^{num_level}$ на уровне num_level , т.е. происходит локальный в рамках одного уровня запрос на объекты данных VD_j^2 и VD_j^3 , объект VD_j^2 является динамической изменяющейся моделью, следовательно, поток 2 отправляет изменённую информацию на вершину уровней для обновления данных на тиражированных элементах (посылка на вершину уровней обусловлена необходимостью поиска всех элементов в системе). Анализ состава данных на уровне $i-2$ показал отсутствие объекта данных, произведена отсылка по иерархии ниже – поток 3. Анализ уровня $i-1$ показал наличие объекта – он был модифицирован, и данные отправлены дальше – поток 4 – вплоть до уровня num_level . Альтернативным вариантом решения данной задачи является хранение на узлах номеров уровней с реплицированными фрагментами данных. Также моделью $M_j^{num_level}$ был произведён запрос объекта данных VD_j^1 , который отсутствует на уровне num_level , поэтому определён набор базовых уровней для получения данных (в данном случае $i-1$ и $i-2$) – потоки 5 и 6 соответственно. На основе модифицированного алгоритма задачи коммивояжёра, перспективным определён путь 5 и уровень $i-1$. В соответствии с этим решением вычисляется стоимость запроса на уровень $i-1$. На последующем этапе оптимизации структуры системы и вычисления критерия эффективности F объект данных VD_j^1 для повышения готовности системы может быть скопирован на уровень num_level (поток 6 или 7).

Список литературы

1. Янюшкин, В.В. Распределённое информационное пространство и портал современного тренажёра / В.В. Янюшкин // «Программные продукты и системы». – 2009. – № 3. – С. 67–71.

2. Янюшкин, В.В. Оптимизация размещения данных модельного мира в распределённой информационной системе тренажёрно-моделирующего комплекса / В.В. Янюшкин // Известия Вузов. Северо-Кавказский регион. Технические науки. – 2008. – № 4. – С. 25–28.

Data Accommodation Optimization Problem in Generalized Mathematical Statement of Information System in Simulators

V.V. Yanushkin

*South-Russian State Technical University (Novocherkassk Polytechnic Institute),
Department of «Automated Management Systems», Novocherkassk*

Key words and phrases: mathematical model; multilevel system; distribution of the modelling world.

Abstract: On the basis analysis classical architecture and algorithms of data accommodation in information environments the generalized mathematical statement of a problem data accommodation and its decision algorithm which differs higher level abstraction initial data and wide application in sphere of various purpose distributed information systems is offered.

© В.В. Янюшкин, 2009