

Построение веб-сервисов систем автоматизации схемотехнического проектирования

В.Н. Гридин, Г.Д. Дмитриевич, Д.А. Анисимов

Аннотация: Рассматривается методика построения веб-сервисов на основе хранения и обработки данных в строчно-столбцовом фиксированном формате. Приводится двухэтапная технология модификации программного обеспечения на основе полных разреженных матриц к компактной форме. Предлагается процедура модификации веб-сервисов к компактной форме строчно-столбцового формата на основе их прототипов полного математического описания.

Ключевые слова: веб-сервисы, разреженные матрицы, сжатие данных, фиксированный формат, автоматизация схемотехнического проектирования.

Введение

Структура реальных объектов схемотехнического проектирования характеризуется слабой связью компонентов друг с другом, что приводит к математическому описанию, содержащему разреженные матрицы, в которых имеется значительное число нулевых элементов. Это обстоятельство требует изменения общепринятых методов формирования и решения систем уравнения, что вызывается, необходимостью экономии оперативной памяти, где нежелательно хранить нулевые элементы, а также требованием увеличения скорости вычислений, путем устранения операций с элементами, имеющими нулевые значения. Используемые для этого способы логической проверки элементов матриц не дают ожидаемого эффекта повышения быстродействия, поскольку для выполнения большого числа логических операций затрачивается значительное время.

В связи с этим возникает задача изменения традиционных подходов к формированию и решению систем уравнения, что объясняется, с одной стороны, требованием экономии памяти, которую нежелательно использовать для хра-

нения нулевых элементов, а с другой стороны, требованием повышения быстродействия за счет устранения арифметических операций с нулевыми элементами. Вопросам компактной обработки разреженных матриц на основе сжатия данных посвящено значительное число работ [1-5], однако конкретная технология реализации сжатия данных освещена в настоящее время недостаточно полно, что существенно затрудняет разработку высокоэффективных программных систем.

Вопросы повышения быстродействия особенно актуальны при организации работы с веб-сервисами [6], поскольку для повышения надежности их функционирования следует минимизировать время взаимодействия с сетью Интернет, что требует перехода к компактной форме хранения и обработки разреженных матриц на основе тех или иных методов сжатия данных.

1. Метод сжатия по Шерману

Существующие методы сжатия данных существенно отличаются друг от друга, при этом наибольшей эффективностью обладает метод компактного хранения, основанный на строчно-

столбцовом формате, называемый также методом сжатия по Шерману [4], который является наиболее экономичным из всех возможных методов компактной обработки разреженных матриц. Этот метод основан на использовании фиксированного формата, при этом исходная матрица должна быть структурно симметричной, так что для каждого ненулевого элемента w_{ij} можно поставить в соответствие элемент w_{ji} . В случае если такой элемент в исходной матрице отсутствует, то его необходимо создать искусственно путем включения в описание элемента $w_{ji} = 0$. Для хранения исходных элементов разреженной матрицы по схеме Шермана требуется три одномерных массива.

- **WD** – массив для хранения n диагональных элементов.
- **WU** – массив для хранения $nu=(m-n)/2$ ненулевых элементов, расположенных выше диагонали (наддиагональных элементов)
- **WL** – массив для хранения $nl=(m-n)/2$ ненулевых элементов, расположенных ниже диагонали (поддиагональных элементов)

При формировании наддиагональные элементы записываются по строкам, а поддиагональные элементы – по столбцам. Согласно такому порядку формирования относительный адрес некоторого элемента w_{ij} , расположенного в массиве **WU**, совпадает с относительным адресом элемента w_{ji} , расположенным в массиве **WL**, что существенно упрощает процедуру сканирования элементов в процессе программирования.

Для хранения индексов строк и столбцов ненулевых элементов используется некоторый массив (например, **WJI**), содержащий номера столбцов ненулевых элементов, расположенных выше диагонали (номера строк элементов, расположенных ниже диагонали). Для определения точки входа в строку матрицы выше диагонали (в столбец ниже диагонали) используется массив **ERC**, в последний элемент которого заносится значение $ERC[n] = nu + 1$.

Поскольку формат всех массивов жестко зафиксирован, и не может меняться произвольным образом в процессе расчета, отличительной особенностью метода является невозможность включения в описание дополнительных ненулевых элементов. Это обстоятельство ис-

ключает возможность непосредственного применения схемы Шермана для обработки информации при решении систем уравнений любым численным методом вследствие неизбежного появления новых ненулевых элементов в процессе этого решения, наличие которых исключает использование этого метода в исходной форме.

2. Модификация метода Шермана

Для возможности полноценного использования метода Шермана на всех шагах процесса моделирования необходима его модификация путем введения двухэтапной процедуры формирования компактного описания, включающей в себя символьный этап и численный этап. Целью символьного этапа является только определение размеров всех используемых массивов с учетом появления новых ненулевых элементов. Так как на символьном этапе не ставится задача формирования численных массивов, то его выполнение осуществляется путем обработки некоторой индексной матрицы **C**, элементы которой имеют только два произвольных значения (например, 0 и 1). При этом, если в исходной матрице некоторый элемент $w_{ij} \neq 0$, то для введенной индексной матрицы **C** соответствующий элемент $C_{ij} = 1$, а все остальные элементы равны 0. Следовательно, вместо рассмотрения исходной схемы, может рассматриваться некоторый «портрет» этой схемы, отображающий ее структуру, но не содержащий информации о численных значениях параметров. Формирование численных массивов осуществляется на втором этапе, при выполнении которого используются определенные на первом этапе форматы массивов, учитывающие зарезервированные места для всех новых ненулевых элементов.

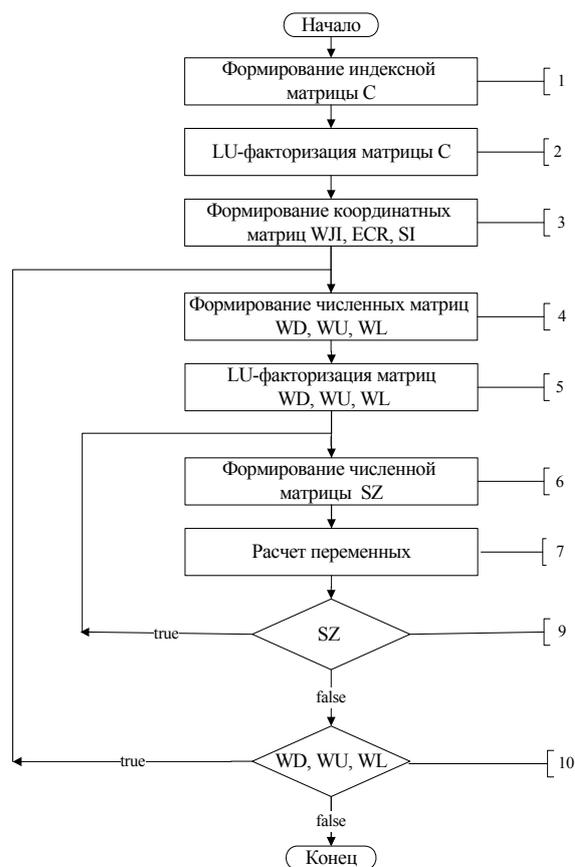
Таким образом, процесс двухэтапной компактной обработки на основе строчно-столбцового фиксированного формата может быть реализован согласно блок-схеме, приведенной на рисунке. В соответствии с блок-схемой, символьный этап (шаги 1–3) начинается с формирования индексной матрицы **C**, имеющей такие же размеры как исходная матрица. На следующем шаге символьного этапа проводит-

ся LU-факторизация индексной матрицы C , в процессе которой выполняется оптимальное упорядочивание строк и столбцов с целью минимизации числа новых ненулевых элементов. Здесь также строится вспомогательный вектор P , отображающий все реализуемые перестановки строк (столбцов) индексной матрицы C . На третьем шаге символьного этапа осуществляется построение координатных матриц WJI , ERC , SI , где будет содержаться информация о структуре индексной матрицы C , и, следовательно, матрица C может быть удалена.

На численном этапе (шаги 4–10) в соответствии с уже известным форматом описания, открываются массивы для формирования численных матриц WD , WU , WL и выполняется формирование этих массивов. На этом же этапе приводится численная LU-факторизация, при этом, поскольку исходная матрица в описании не существует, то LU-факторизация выполняется виртуально путем обработки компактных массивов WD , WU , WL , т. е. для этой цели необходима разработка специального алгоритма LU-факторизации, отличающегося от известных алгоритмов общего вида.

После завершения численного этапа LU-факторизации на шестом шаге осуществляется формирование массива задающих источников SZ , после чего на седьмом шаге выполняется расчет всех переменных системы и перекодировка этих переменных согласно проведенным на символьном этапе перестановкам строк и столбцов. Такая задача в соответствии с общей методикой LU-факторизации решается при помощи обратного и прямого хода по строкам исходной матрицы. Поскольку исходная матрица отсутствует, то прямой и обратный ход должны выполняться на основании компактного описания, что так же требует создания специальных алгоритмов, отличающихся от известных алгоритмов решения уравнений на основе полных матриц.

На девятом шаге осуществляется проверка необходимости изменения содержимого массива SZ задающих источников и в случае необходимости выполняется возврат к шестому шагу. На десятом шаге осуществляется проверка необходимости изменения содержимого массивов численных параметров схемы, содержащихся в компактных массивах WD , WU , WL , и в случае



Блок-схема двухэтапной компактной обработки данных

необходимости выполняется возврат к четвертому шагу. Существенным достоинством рассмотренной двухэтапной процедуры является разделение ее на две независимые части символьного и численного этапа. Так как практически все реальные задачи проектирования электронных схем связаны с многовариантным расчетом схемы одной и той же структуры, то символьный этап выполняется для каждой структуры единственный раз, в то время как численный этап реализуется десятки, сотни, а часто и тысячи раз. Вместе с тем, двухэтапная процедура характеризуется довольно сложной логикой построения программного обеспечения и переход к компактному описанию является весьма трудоемким процессом.

Построение веб-сервиса на основе сжатия данных существенно упрощается при наличии его прототипа на основе полного математического описания задачи. При наличии такого прототипа при решении задачи модификации

веб-сервиса за основу следует положить блок-схему, показанную на рисунке, которая описывает символичный и численный этапы, а также блок-схему конкретного прототипа модифицируемого метода и его программное обеспечение.

На первом шаге символьного этапа выполняется формирование индексной матрицы $short[][]C$. Основная информация о включении компонентов схемы хранится в строках и столбцах, индексы которых находятся в интервале от 1 до n , в нулевом столбце массива хранится информация о месторасположении источников, а в нулевой строке – содержание вектора перестановок P , который вначале инициализируется натуральным рядом целых чисел и в котором хранится информация о всех перестановках строк и столбцов матрицы W . Для формирования индексной матрицы при решении задачи модификации веб-сервиса на основе имеющегося прототипа метода, следует использовать имеющиеся в этом прототипе функции формирования $form_str$ и построить на их основе новую совокупность функций $sform_str$, единственное отличие которых от прототипа заключается в замене оператора формирования элемента массива $w[i][j]$ оператором формирования соответствующего элемента индексной матрицы $C[i][j] = 1$.

Для выполнения символьного этапа LU-факторизации необходимо построить вспомогательную функцию, осуществляющую замену процедуры реальной LU-факторизации ее эквивалентом на основе структурного «портрета» схемы, отображенной в индексной матрице. Обработка «портрета» схемы должна выполняться одновременно с проведением оптимального упорядочивания нумерации строк и столбцов матрицы S . Для такого упорядочивания наиболее эффективным является алгоритм частичного упорядочивания, который реализуется исходя из выбора минимального значения произведения величины $r_k * c_k$, где r_k – число ненулевых элементов в k -й строке на k -ом шаге, а c_k – число ненулевых элементов в k -ом столбце. Применительно к структурно-симметричной матрице выполняется соотношение $r_k = c_k$ и алгоритм частичного упорядочивания приводит к требованию минимизации числа элементов в k -й строке. При этом для сохранения структурной симметрии в про-

цессе оптимального упорядочивания необходимо осуществлять перестановку как строк, так столбцов. Поскольку перестановка столбцов приводит к перестановке вектора переменных, то информацию о выполненных перестановках необходимо обязательно сохранять до конца расчета, для чего используется нулевая строка массива S .

Помимо решения задачи оптимального упорядочивания строк и столбцов исходя из критерия минимизации появления новых ненулевых элементов, при выборе ведущего элемента на символьном этапе LU-факторизации следует также обеспечить ненулевое значение этого элемента в полном описании матрицы W . Очевидно, что только такое условие позволит выполнить пересчет элементов на численном этапе LU-факторизации. Вместе с тем, при описании систем в расширенном базисе узловых потенциалов, не равные нулю значения диагональных коэффициентов матрицы, обеспечиваются только для значений n , не превышающих величины nv , где nv – число узлов схемы. При $n > nv$ диагональные элементы могут иметь нулевые значения, и, следовательно, выбор их в качестве ведущих недопустим. Чтобы устранить указанное препятствие, следует соответствующим образом организовать работу оптимального упорядочивания строк и столбцов в процессе символической LU-факторизации.

С этой целью в начале упорядочивания следует провести перестановки строк и столбцов индексы, которые не превышают значение nv , а затем осуществить процедуру упорядочивания строк и столбцов, изменяя индексы в пределах от $nv+1$ до n . Если следовать такому алгоритму оптимального упорядочивания, то в процессе пересчета элементов при $n \leq nv$, все диагональные элементы для $n > nv$ приобретут ненулевые значения и работа на численном этапе окажется возможной.

После проведения символической LU-факторизации необходимо провести подсчет числа ненулевых наддиагональных элементов ni в индексной матрице S , а также число ненулевых элементов ns в нулевом столбце, при этом значение ni определяет формат массива наддиагональных элементов WU и массива поддиагональных элементов WL , а значение ns – формат массива SI .

Итогом формирования координатного описания является построение массивов WJI, ERC, SI, отображающих соответствующие матрицы, формат которых уже определен на предыдущих шагах символического этапа. Компактный массив WJI, содержит индексы столбцов исходной полной матрицы в ее наддиагональной части и, вследствие структурной симметрии исходной матрицы, одновременно содержит так же и индексы столбцов ненулевых элементов ее поддиагональной части. Связанный с индексом j -го столбца относительный адрес aw может быть найден путем последовательного сканирования строк индексной матрицы C , с подсчетом текущего суммарного значения числа aw элементов $C_{ij} = 1$, расположенных выше диагонали ($j = \overline{i+1, n}$), что позволяет определить элемент $WJI[aw] = j$. Формирование координатной матрицы ERC, содержащей относительные адреса входа в строку, осуществляется в одном цикле с формированием WJI путем нахождения относительного адреса aw , при этом сканирование i -й строки определяет значение $ERC[i] = aw + 1$.

Исходными данными для формирования описания компонентов на численном этапе являются массивы in_str и z_str , кроме того, известными являются содержание массивов WJI и ERC. Конечной целью является описание компонентов в компактных массивах WD, WU, WL, для чего необходимо решить две вспомогательные задачи. Во-первых, необходимо определить новые значения ii индексов строк и столбцов jj , которые получаются после многократных перестановок этих строк и столбцов в процессе их оптимального упорядочивания, информация о которых содержится в векторе P . Наиболее эффективным способом решения указанной задачи перекодировки индексов каждой переменной, является включение в текст программы фрагментов

$ii = i; \text{ while } (P[ii] \neq i) \text{ } ii = P[ii];$

$jj = j; \text{ while } (P[jj] \neq j) \text{ } jj = P[jj];$

Второй задачей, возникающей на этапе численного формирования, является определение относительного адреса, т.е. местоположения элемента в массивах WL, WU, куда необходимо занести значение параметра w_{ij} описываемого компонента. Так, если требуется найти относи-

тельный адрес для некоторого элемента w_{ij} , для которого значения индексов строк и столбцов изменены с i на ii и с j на jj , то такая задача может быть решена путем включения в программу фрагмента $a = ERC[ii]; \text{ while } (WJI[a] \neq jj) \text{ } a++$; Очевидно, что при выходе из цикла значение относительного адреса a будет соответствовать элементу, расположенному в (ii) -й строке и (jj) -м столбце. Вследствие структурной симметрии, этот же относительный адрес определяет положение элемента в ii -м столбце и jj -й строке ниже диагонали.

В соответствие с изложенной стратегией, нетрудно построить совокупность функций формирования для всех компонентов str схемы, структура которых в целом будет повторять структуру, аналогичную соответствующим функциям формирования полной матрицы и отличается от последних только наличием, приведенных выше фрагментов для выполнения перекодировки и нахождения относительных адресов.

Для численной LU-факторизации необходима разработка алгоритмов поиска элементов $w_{kk}, w_{ki}, w_{kj}, w_{ij}$ непосредственно в компактных массивах WD, WU, WL.

Элемент w_{kk} легко выделяется из компактного массива WD по известному значению переменной k , т.е. $w_{kk} = WD[k]$. Элементы w_{kj} и w_{ik} находятся сканированием массива WJI в границах $a_{k1} = ERC[k], a_{k2} = ERC[k+1]-1$, что позволяет найти относительные адреса a_i и a_j расположения индексов i и j . В отличие от элементов w_{kj} и w_{ik} , расположение которых относительно диагонали заранее известно, элемент w_{ij} может находиться в любом месте j -го столбца. При этом, возможны три случая

а) $i=j$, при этом элемент следует искать в массиве WD, т.е. $w_{ii} = WD[i]$;

б) $i < j$, при этом поиск надо осуществлять в i -ой строке j -ого столбца, начиная с относительного адреса $a_i = ERC[i]$, и искомым элемент извлекается из массива WU.

в) $i > j$, поиск необходимо осуществлять в j -ом столбце и i -ой строке. Так как ниже диа-

гонали элементы расположены по столбцам, то относительный адрес входа в столбец определяется выражением $a_j = \text{ERC}[j]$, и искомый относительный адрес находится в пределах границ a_{j_1} и a_{j_2} , зная значение относительного адреса можно извлечь искомый элемент из WL.

Таким образом, численная LU-факторизация может быть осуществлена на основе компактных численных массивов WD, WU, WL и координатных массивов ERC, WII без отображения в памяти компьютера полной разреженной матрицы W. Заключительным шагом численного этапа является расчет вектора базисных переменных X, в процессе которого выполняется прямой ход и обратный ход, реализация которых должна быть также осуществлена в компактной форме.

Заключение

Разработанный алгоритм формирования и решения уравнений, представленных в компактной форме на основе строчно-столбцового фиксированного формата, реализован в виде функции, которая получает в качестве своих

аргументов значения координатных массивов WJI, ERC, SI, P и преобразованных в процессе LU-факторизации массивов WD, WU, WL.

Предложенная методика позволяет модифицировать программное обеспечение веб-сервисов распределенных систем автоматизации схмотехнического проектирования к компактной форме, что значительно повышает их эффективность.

Литература

1. Кнут Д. Искусство программирования для ЭВМ. Том 1: Основные алгоритмы, пер. с англ. - М.: Мир, 1976.
2. Норенков И.П. Введение в автоматизированное проектирование технических устройств и систем. - М.: Высшая школа, 1986.
3. Эстербю О., Златев З.. Прямые методы для разреженных матриц, пер. с англ. - М.: Мир, 1987.
4. Писсанецки С. Технология разреженных матриц, пер. с англ. - М.: Мир, 1988.
5. Влах И., К.Сингхал.; Машинные методы анализа и проектирования электронных схем, пер. с англ. - М.: Радио и связь, 1988.
6. Гридин В.Н., Дмитриевич Г.Д., Анисимов Д.А. Построение систем автоматизированного проектирования на основе Web-технологий, Информационные технологии, №5, 2011, С. 23-27.

Гридин Владимир Николаевич. Директор Центра информационных технологий в проектировании РАН. Окончил Московский авиационный технологический институт в 1972 году. Доктор технических наук, профессор. Автор 200 печатных работ. Область научных интересов: информационные технологии в проектировании, безопасность систем. E-mail: info@ditc.ras.ru

Дмитревич Геннадий Данилович. Профессор Санкт-Петербургского электротехнического университета. Окончил Ленинградский электротехнический институт в 1960 году. Доктор технических наук, профессор. Автор 150 печатных работ. Область научных интересов: моделирование систем, методы оптимизации. E-mail: gddm@inbox.ru

Анисимов Денис Андреевич. Аспирант Санкт-Петербургского электротехнического университета. Окончил Санкт-Петербургский электротехнический университет в 2010 году. Автор 14 печатных работ. Область научных интересов: моделирование систем, веб-ориентированные системы. E-mail: anisimovdenis2009@yandex.ru