В процессе расчета система уравнений изменяется в моменты дискретных изменений. В некоторых случаях изменяется количество уравнений, что приводит к необходимости изменения структуры матрицы и вектора правой части. Максимальная размерность матрицы может быть определена заранее по количеству переменных всех устройств модели, вне зависимости от их состояний. Возможен подход с поддержанием постоянной максимальной размерности матрицы. Неактивные уравнения могут задаваться строкой матрицы с единичным диагональным элементом и нулевым значением правого вектора. В этом случае неактивная переменная будет константой.

Но такой подход неэффективен для крупных моделей, в которых значительно число устройств нормально отключено. К числу неактивных устройств можно также отнести узлы, входящие в суперузлы. В моделях с большим количеством выключателей таких узлов может оказаться достаточно много. Поэтому принят подход с перестроением структуры матрицы. Неактивные устройства исключаются из матрицы, что позволяет сократить ее размерность. В данном подходе, тем не менее, используется комбинация с единичной строкой. Если некоторое устройство изменяет состояние и при этом количество уравнений изменяется не более чем на одно или два, то перестраивать структуру матрицы неэффективно. Вместо этого модифицируется структура уравнений, но размерность матрицы остается постоянной. Изменение размерности происходит при значительных изменениях количества активных уравнений, например при коммутациях.

При изменении количества уравнений необходимо сохранять информацию об устройствах, которые исключаются из матрицы, так как в процессе расчета может оказаться что данные устройства должны быть вновь включены. Для этого организуется система из двух правых векторов: полного и рабочего.

Полный вектор строится для полного состава всех устройств модели со всеми переменными. Размерность этого вектора постоянна. Рабочий вектор строится только для устройств, которые на данном отрезке времени расчета активны и требуют ввода уравнений в матрицу. Класс рабочего вектора наследуется от класса полного, поэтому векторы включают в себя общую часть информации. Минимальный набор общей части – указатель на устройство и указатель на переменную правой части. Эти данные позволяют синхронизировать векторы и проверить корректность их синхронизации. В полном векторе также уместно хранить глобальные данные: например, количество ошибок по переменной и т.п.

При изменении количества уравнений в матрице до изменения данные из рабочего вектора сначала сохраняются в полном, далее выполняется изменение размерности, после чего данные из полного вектора синхронизируются с данными рабочего. Процессы синхронизации, таким образом, выполняются в двух направлениях. Оба процессе не требуют поиска пар указателей на устройства и переменные и выполняются просто по порядку устройств и переменных.

Алгоритм синхронизации из рабочего вектора в полный (до изменения размерности)

1. В цикле по всем устройствам полного вектора:
2. Проверяем, входит ли устройство в матрицу
3. Если да, синхронизируем все переменные из рабочего вектора с полным, если нет – пропускаем количество уравнений устройства из рабочего вектора в полном векторе

Алгоритм синхронизации из полного вектора в рабочий (после изменения размерности) выполняется также, за исключением направления синхронизации в п. 3.

В процессе синхронизации для контроля корректности можно сравнивать пары указателей на устройства и переменные, что даст гарантию соответствия элементов рабочего и полного векторов.

В модели есть типы устройств, количество которых заранее определить невозможно. К таким устройствам, например, относятся синхронные зоны, которые образуются в зависимости от изменения топологии сети. Устройства такого типа необходимо учитывать как Volatile устройства. Для них данные в полном векторе не резервируются. Записи таких устройств создаются только в рабочем векторе. В процессе синхронизации такие устройства обходятся также, как и устройства вне матрицы. Однако признак Volatile не совпадает с признаком ввода в матрицу: Volatile устройство входит в матрицу, но не входит в полный вектор.

