

УДК 681.3(06)

## ЭФФЕКТИВНОСТЬ MIMD РЕШАТЕЛЕЙ УРАВНЕНИЙ АЭРОДИНАМИКИ ШАХТНЫХ ВЕНТИЛЯЦИОННЫХ СЕТЕЙ НА БАЗЕ MPI И OPENMP СТАНДАРТОВ

Смагин А.Н.

Кафедра ЭВМ, ДонНТУ

*Ця робота відноситься до проблематики побудови ПМС [1], орієнтованих на особливості топології, аерогазодинамічних процесів і засобів керування ними. Також розглядається оцінка трудоемкості та характеристик розпаралелювання MIMD вирішувачів рівнянь на базі MPI та OpenMP стандартів для ПМС.*

Решатель уравнений – аппаратно-програмное средство, которое реализует некоторый численный метод [2] решения сформированных уравнений. Система уравнений аэродинамики ШВС представляется в виде [3,4]:

$$\begin{cases} X = -W \cdot Y \\ \frac{dY}{dt} = TP \cdot H - RU \cdot Z \end{cases}, \text{ где } TP = (S_y K_y - S_x K_x W)^{-1} \cdot S, \quad RU = (S_y K_y - S_x K_x W)^{-1} \cdot S \cdot R, \quad W = A_x^{-1} \cdot A_y.$$

На подготовительном этапе формируется дерево и антидерево ШВС, которую можно представить в виде графа, с подвекторами потоков расходов воздуха в ветвях дерева  $X = (X_1, X_2, \dots, X_{n-1})^T$  и антидерева  $Y = (Y_1, Y_2, \dots, Y_{m-n+1})^T$ . Матрицы инцидентий  $A$ , независимых контуров  $S$  а также единичные матрицы физических параметров ШВС  $R$  и  $K$ , упорядочены относительно векторов  $X, Y$ .

Вычисления расходов воздуха в ветвях дерева графа назовем  $X$  компонентой. Для определения расходов воздуха в ветвях дерева,  $X$  компонента решателя уравнений выполняет умножение топологическую коммутирующую матрицу  $W$  на вектор значений расходов в ветвях антидерева. Вычисление расходов воздуха в ветвях антидерева назовем  $Y$  компонентой.  $Y$  компонента находит вектор значений функций производных  $dY/dt$  и с помощью выбранного численного метода определяет значения расходов воздуха в ветвях антидерева. Процесс решения повторяется пока переходной процесс не завершится. Таким образом алгоритм функционирования решателя уравнений состоит из двух частей: вычисление  $X$  и  $Y$  компонент.  $T_1$  - временная сложность решателя уравнений состоит из суммы  $T_1^X$  - временной сложности вычисления компоненты  $X$  и  $T_1^Y$  - временной сложности вычисления компоненты  $Y$ .

Оценим трудоемкость вычисления компоненты  $X$ . Компонента  $X$  вычисляется по следующей формуле:

$$X_k = - \sum_{p=1}^{\gamma} W_{kp} * Y_{p1}, k = 1, 2, \dots, n-1$$

Временная сложность этой части алгоритма будет:

$$T_1^X = (n-1) \cdot (\gamma \cdot t_{yM} + (\gamma-1) \cdot t_{cl})$$

Трудоемкость вычисления компоненты  $Y$  зависит от формулы правой части дифференциального уравнения -  $T_f^Y$  и от выбранного численного метода -  $T_{\times i}^Y$ . Временная сложность вычисления значения производной функции дифференциального уравнения будет:

$$T_f^Y = m \cdot t_{\hat{\alpha}} + \gamma \cdot (m \cdot t_{\hat{\alpha}} + (m-1) \cdot t_{\hat{n}\ddot{e}}) + \gamma \cdot t_{\hat{n}\ddot{e}}$$

Временная сложность вычисления производной в векторно-матричном виде с помощью численного метода Эйлера будет:

$$T_{\times i}^Y = \gamma \cdot T_{\dot{Y}}$$

где  $T_{\dot{Y}}$  - временная сложность метода Эйлера  $T_{\dot{Y}} = t_{\hat{\alpha}} + t_{\hat{n}\ddot{e}}$ .

Анализ последовательного решателя уравнений аэродинамики ШВС показал, что используется одна и та же последовательность инструкций только для разных данных. Следовательно в данном случае распараллеливание нужно выполнять SPMD по принципу [5].

Можно выделить два подхода к распараллеливанию РУ. Подход 1: компоненты  $X$  и  $Y$  не объединены предусмотрено  $m$  виртуальных процессов, каждый из которых ведет вычисление либо компоненты  $X$ , либо  $Y$ . После нахождения текущих значений компоненты  $X$  выполняется вычисление значений правых частей  $Y$  компоненты, в ходе которого происходит обмен информацией между процессами, дальше элементы  $Y$  - компоненты передаются виртуальным процессам компоненты  $X$ . Девиртуализация для подхода 1 заключается в объединении  $X$  - процессов в группы, что составят  $X$  компоненту, а  $Y$  - процессы -  $Y$  компоненту, с целью выровнять нагрузку процессоров двух компонент так как  $X$  - процессы простаивают часть времени, кроме того, количество операций, которая выполняется этими процессами, меньшая, чем у  $Y$  - процессов.

Подход 2: компоненты  $X$  и  $Y$  объединены и предусмотрено  $g$  виртуальных процессов, что соответствует размерности  $Y$  компоненты, каждый из которых ведет вычисление компонент  $X$  и  $Y$ . После нахождения текущих значений компоненты  $X$  выполняется вычисление значений правых частей  $Y$  - компоненты, в ходе которого происходит обмен информацией между процессами, дальше

виртуальные процессы обмениваются найденными элементами  $Y$  компоненты. Девиртуализация для подхода 2 заключается в том, что виртуальные процессы полностью идентичны, в  $i$ -м виртуальном процессе исчисляются  $X_i, Y_i$ . В ситуации  $\gamma < n-1$  целесообразно привязывать количество процессов к величине  $\gamma$ , а те  $X^{\gamma+1}, \dots, X_{n-1}$ , что остались не распределенными, распределить равномерно среди виртуальных процессоров. При  $\gamma > n-1$  можно допустить работу части  $Y$ -процессов без вычисления компонентов вектора  $X$ .

В соответствии с подходом 1, обозначим количество доступных процессоров на целевой параллельной вычислительной системе как  $s$ , тогда  $s_x$  - количество процессоров, на котором вычисляется  $X$  компонента, а  $s_y$  - количество процессоров, на котором вычисляется  $Y$  компонента и выполняется равенство  $s = s_x + s_y$ . Зная количество процессоров  $X$ - и  $Y$ -компоненты, можно найти  $k_x$  - количество элементов  $X$ -компоненты и  $k_y$  - количество элементов  $Y$ -компоненты, вычисляемые на одном процессоре:

$$k_x = \begin{cases} \left\lceil \frac{n-1}{s_x} \right\rceil, ((n-1) \bmod s_x) = 0 \\ \left\lceil \frac{n-1}{s_x} \right\rceil + 1, ((n-1) \bmod s_x) \neq 0 \end{cases} \quad k_x = \begin{cases} \left\lceil \frac{n-1}{s_x} \right\rceil, ((n-1) \bmod s_x) = 0 \\ \left\lceil \frac{n-1}{s_x} \right\rceil + 1, ((n-1) \bmod s_x) \neq 0 \end{cases}$$

Определим количество элементов  $X$ -компоненты, вычисляемые на одном процессоре и количество элементов  $Y$ -компоненты для подхода 2:

$$k_x = \begin{cases} \left\lceil \frac{n-1}{s} \right\rceil, ((n-1) \bmod s) = 0 \\ \left\lceil \frac{n-1}{s} \right\rceil + 1, ((n-1) \bmod s) \neq 0 \end{cases} \quad k_x = \begin{cases} \left\lceil \frac{n-1}{s} \right\rceil, ((n-1) \bmod s) = 0 \\ \left\lceil \frac{n-1}{s} \right\rceil + 1, ((n-1) \bmod s) \neq 0 \end{cases}$$

Рассмотрим время выполнения одной итерации цикла (определение компоненты  $X$  и  $Y$ ) РУ. В общем случае время выполнения  $T_s = T_s^X + T_s^Y$  складывается из времен вычисления компонент  $X$  и  $Y$ -компоненты.

Для подхода 1 время вычисления  $X$  компоненты будет

$$T_s^X = k_x \cdot (\gamma \cdot t_{\hat{\alpha}} + (\gamma-1) \cdot t_{\hat{\alpha}\hat{\alpha}})$$

Формула  $Y$  компоненты для случая аэродинамических уравнений (метод Эйлера) будет следующей:

$$T_f^{Y_s} = k_x \cdot t_{\hat{\alpha}} + \gamma \cdot (k \cdot t_{\hat{\alpha}} + (k-1) \cdot t_{\hat{\alpha}\hat{\alpha}}) + k_y \cdot t_{\hat{\alpha}\hat{\alpha}}$$

Для подхода 2 формула вычисления общего времени  $T_s$  и вычисления компоненты X совпадают с формулами подхода 1. Формула Y компоненты будет

$$T_f^{Y_s} = t_{\bar{a}} \cdot (k_x + k_y) + \gamma \cdot ((k_x + k_y) \cdot t_{\bar{a}} + ((k_x + k_y) - 1) \cdot t_{\bar{n}e}) + k_y \cdot t_{\bar{n}e}$$

Для оценки целесообразности применения параллельных подходов используются такие характеристики как ускорение и эффективность. Ускорение определяется как отношение времени выполнения последовательного алгоритма к времени параллельной реализации и показывает во сколько раз уменьшилось время выполнения при использовании заданного количества процессоров.

Эффективность определяется как отношение ускорения на заданном количестве процессов к количеству процессоров. Характеристика показывает на сколько текущее ускорение отличается от идеального ускорения, определяет эффективность использования процессоров.

Входные параметры для математической модели были взяты для реальной шахтной вентиляционной сети, у которой количество ветвей  $m=117$ , узлов-  $n=61$ . На рисунке 1 показана зависимость ускорения и эффективности параллельного алгоритма решателя уравнений от количества процессоров и способа девиртуализации.

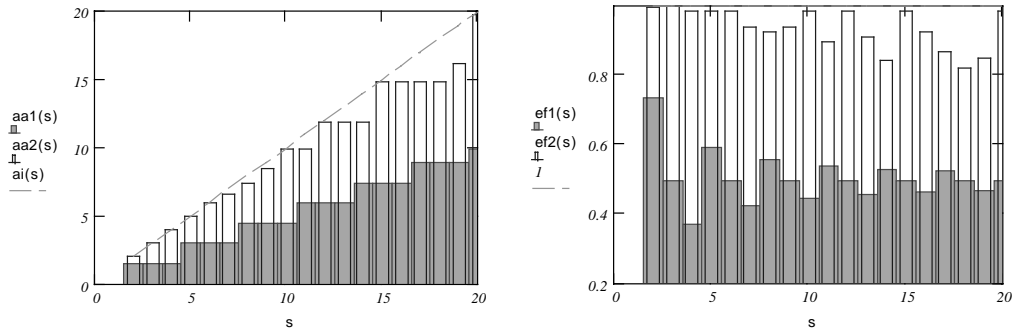


Рисунок 1 – Ускорение и эффективность параллельного алгоритма решателя уравнений в зависимости от количества процессоров и способа девиртуализации

Предложенные подходы к распараллеливанию были реализованы на базе стандартов параллельного программирования MPI и OpenMP. В качестве целевой параллельной машины был взят NEC-SX8 с гибридной архитектурой, на каждом узле 8 процессоров, которые работают через общую память. На рисунке 2 показана зависимость ускорения и эффективности параллельного алгоритма решателя уравнений от количества процессоров и способа девиртуализации реализованных на базе MPI и OpenMP стандартов.

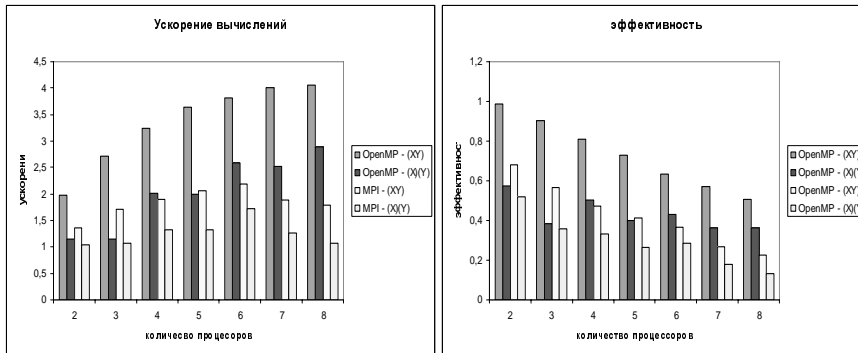


Рисунок 2 – Ускорение и эффективность параллельного алгоритма решателя уравнений в зависимости от количества процессоров и способа девиртуализации реализованных на базе MPI и OpenMP стандартов

Экспериментальные исследования показали что оба параллельных подхода более эффективны при использовании OpenMP стандарта нежели MPI. Эффективность 2-го подхода лучше 1-го подход на реализация и MPI и OpenMP и колеблется от 0.9 до 0.6, что является хорошим показателем. Можно сделать вывод что для моделирования задач с размерностью графа с более 500 ветвями лучше использовать совместно оба стандарта для реализации гибридных параллельных моделей. Предложенная математическая модель применяться для балансировки загрузки процессоров ПМС и определяет оптимальное количество процессоров, которые необходимо выделить ПМС, что позволяет более эффективно использовать дорогостоящие аппаратные ресурсы.

#### Библиографический список

1. Святный В.А. Проблемы параллельного моделирования складных динамических систем.- Наукові праці ДонДТУ, серія ІКОТ, вип. 6, Донецьк, 1999, С. 6-14.
2. Калиткин Н.Н. Численные методы. – М.: Наука, 1978. – 512 с.
3. Абрамов Ф.А., Фельдман Л.П., Святный В.А. Моделирование динамических процессов рудничной аэрологии / Киев, Наукова думка, 1981.
4. Святный В.А. Моделирование аэрогазодинамических процессов и разработка систем управления проветриванием шахт. Докт. диссертация. ДПИ, Донецк, 1985.
5. Святный В.А., Смагін О.М., Солонін О.М. Методи розпаралелювання вирішувача рівнянь MIMD-моделі мережних динамічних об'єктів, Наукові праці ДонДТУ, Серія «ІКОТ», вип. 70, 2003. – С. 20–29.

20.05.2008