

УДК 004.932.2+004.932.72'1

**О. Э. Серeda, А.Ю. Иванов**Донецкий национальный технический университет, г. Донецк  
кафедра автоматизированных систем управления**КЛАСТЕРНАЯ МОДЕЛЬ ДИНАМИЧЕСКОЙ СИСТЕМЫ.****Аннотация**

*Серeda О.Э., Иванов А.Ю. Кластерная модель динамической системы. Разработка программно-аппаратных средств для исследования все более сложных динамических систем обеспечивает модельной поддержкой объекты техники в области механики твердых тел, молекулярной динамики, химических производств, электронных схем и других. При моделировании во временной области системы и объекты чаще всего описываются системами дифференциальных и дифференциально – алгебраических уравнений. Для численного интегрирования жестких систем и полунатурных объектов в реальном времени требуются значительные вычислительные мощности. Одним из способов повышения производительности комплексов моделирования является использование многопроцессорных, в частности кластерных, систем.*

**Постановка проблемы**

Целью данной работы было написать программное обеспечение, которое позволяло бы с минимальными задержками производить расчет характеристик динамической системы. В качестве динамического объекта была взята система уравнений Вандерполя[1], которая имеет вид:

$$\begin{cases} \dot{x}_1 = -w_1^2 x_1 \\ \dot{x}_2 = -w_2^2 x_2 \end{cases} \quad (1)$$

**1 Оптимизация обмена информацией на основе многопроцессорной системы**

Кластер — группа компьютеров, объединённых высокоскоростными каналами связи и представляющая с точки зрения пользователя единый аппаратный ресурс. Другими словами кластер является разновидностью параллельной или распределенной системы, которая состоит из нескольких связанных между собой компьютеров и используется, как единый унифицированный ресурс.

Кластер Донецкого национального технического университета представляет собой вычислительную систему MPP – архитектуры. Особенностью данного типа архитектур является то, что память физически разделена. Однако разделение памяти приносит задержку в вычислительные процессы связанную с увеличением времени обмена между процессорами. Архитектура Кластера ДонНТУ представлена на рисунке 1[2].

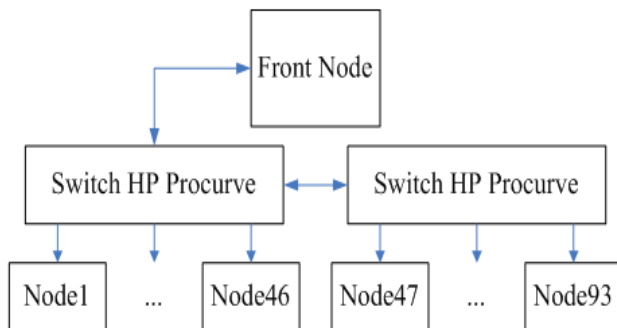


Рисунок 1. – Структура кластера NeClus.

При этом структура каждого узла Node*i* имеет вид представленный на рисунке 2.

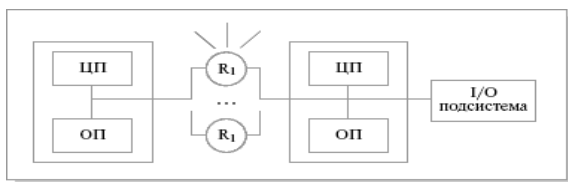


Рисунок 2. – Структура узлов кластера.

Анализ системы показывает, что для создания и решения требуемой задачи необходимо контролировать весь процесс решения на самом низком уровне для уменьшения задержки передачи данных по сети. Это говорит о том, что стандарты типа MPI не подходят для решения данной задачи в связи с невозможностью контроля длины пакетов и уровня синхронизации передачи сообщения. Отсюда был сделан вывод, что необходимо разработать открытое программное обеспечение, которое позволит контролировать данные аспекты работы динамической системы.

На кластере ДонНТУ используется ОС ScientificLinux 5.4. Для передачи и синхронизации обменов использовался протокол UDP и широковещательная рассылка, а в качестве средства передачи – сокеты.

## **2 Анализ математического описания модели динамической системы. Аналитическое решение системы дифференциальных уравнений.**

Для создания программной модели ДО выполняется численное интегрирование системы дифференциальных уравнений, полученных

понижением порядка уравнений Вандрополя (1), каждое из которых после понижения порядка преобразуется в два уравнения первого порядка:

$$\begin{cases} \dot{x}_1 = y_1 \\ y_1' = -\omega_1^2 x_1 \\ \dot{x}_2 = y_2 \\ y_2' = -\omega_2^2 x_2 \end{cases} \quad (2)$$

При получении аналитического решения системы была написана последовательная программа решения на одном процессоре для двух уравнений системы(2). В итоге получился эталонный график решения системы (2), который изображён на рисунке 3. Для решения первой пары уравнений  $\omega_1=1$ .

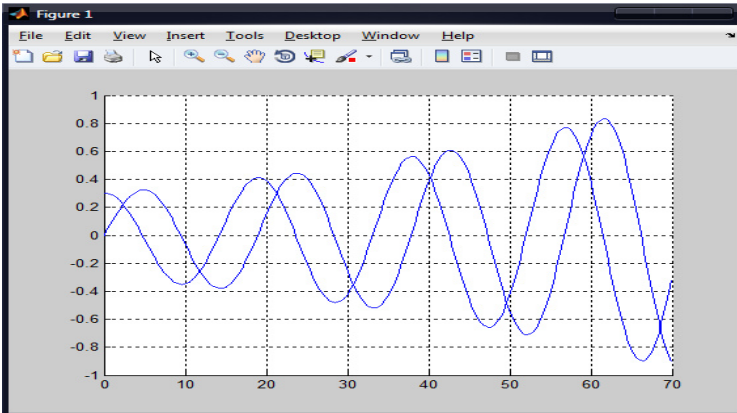


Рисунок 3. – Эталонное решение двух уравнений системы(2).

### 3 Исследование модели динамического объекта на много процессорной системе.

Программное обеспечение для распараллеливания процессов на кластерной системе было описано авторами в научно исследовательской работе принявшей участие в конкурсе НИРС ДонНТУ осенью 2012 года. На его базе была разработана система передачи сообщений для обеспечения обмена между процессорами вычислительной системы.

Для подключения к серверу и отдельным узлам кластера используется протокол SSH. Это сетевой протокол прикладного уровня, позволяющий производить удалённое управление операционной системой и туннелирование TCP-соединений (например, для передачи файлов). SSH позволяет безопасно передавать в незащищённой среде

практически любой другой сетевой протокол. Таким образом, можно не только удалённо работать на компьютере через командную оболочку, но и передавать по зашифрованному каналу звуковой поток или видео (например, с веб-камеры). При распараллеливании используется специальная команда Linux `nohup`, которая запускает указанную команду с игнорированием сигналов потери связи (`hangup`) таким образом, команда будет продолжать выполняться в фоновом режиме и после того, как пользователь выйдет из системы. Синхронизация запуска вычислений обеспечивается широкопередаточной рассылкой сообщений.[3] Протокол UDP позволил отправлять сообщения не дожидаясь их доставки, благодаря чему возросло быстродействие системы в целом. В конечном счете основная проблема заключалась в том, что в свойствах сокета для передачи дейтаграмм нельзя изменить размер буфера для одновременного хранения сообщений. В связи с этим, при использовании одного сокета, сервер не успевает обработать большой поток входящих сообщений и сообщения или затирают друг друга, или вместо корректных результатов сервер получает нулевые значения. Опытным путем было доказано, что сервер в состоянии обработать поток сообщений от 3 процессоров, поэтому для обеспечения корректного приема результатов от 4 процессоров (4 уравнения) потребовалось создать 2 сокета. Каждый сокет был привязан к отдельному порту. Это позволило системе работать корректно, а также закрепить за каждым сокетом прием результатов от попарного решения уравнений, т.е. два уравнения на одном сокете, а два на другом. Для еще большего увеличения производительности системы результаты не сразу записываются в файл. Решения каждого из уравнений сначала записываются в буферы, что ускоряет обработку входящих сообщений. В конце работы ПО данные из буферов заносятся в файл по порядку записи.

Было сделано решение, что на каждом процессоре решается свое уравнение дабы показать все возможности данной системы. Конечные результаты передаются на сервер где и формируется выходной файл результатов. Работу ПО и результаты решения системы можно увидеть на рисунках 4 и 5.

```
[student@neclus] ~/diplom $ ./p.exe 5
c.exe           100% 8307      8.1KB/s 0
c.exe           100% 8307      8.1KB/s 0
c.exe           100% 8307      8.1KB/s 0
ssh: connect to host node4 port 22: No route to host
lost connection
ssh: connect to host node4 port 22: No route to host
c.exe           100% 8307      8.1KB/s 0
OK
Broadcast message was sended!!!
Message from node 2:node2
Message from node 3:node3
Message from node 1:node1
Message from node 5:node5
[student@neclus] ~/diplom $ █
```

Рисунок 4. – Результаты работы программы тестирования распараллеливающего ПО.

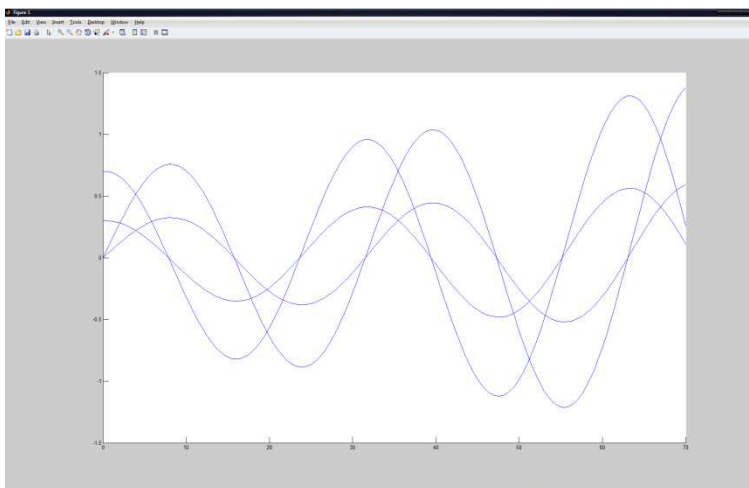


Рисунок 5. – Результат решения системы Вандерполя(2) на кластере ДонНТУ при  $\omega_1=\omega_2=1$ .

#### Выводы

Проведенные исследования показали работоспособность разработанного ПО и доказали возможность синхронного решения нескольких уравнений. Правильность решения выбранной динамической системы можно увидеть на представленных рисунках. Следующим шагом исследования ПО будет добавление пятого уравнения вида:

$$X_1^2 + X_2^2 = 1$$

Данное уравнение выражает зависимость синусоидальных переменных в гармоническом колебании. Решение системы из 5 уравнений позволит проследить поведение динамической системы в едином пространстве.

#### Список литературы

- [1] Максименко В. Л. Исследование и разработка специальных программных способов для моделирования автоколебаний. Портал магистров ДонНТУ.
- [2] Параллельная вычислительная система – кластер NeClus[электронный ресурс]. Тип доступа: <http://neclus.donntu.edu.ua/>
- [3] Шон Уолтон Создание сетевых приложений в среде Linux. Руководство разработчика: Москва/Санкт-Петербург/Киев, 2001г. – 116-213 с., 358-379с.