

Мануал. Программа NeiSys

Трофимов Е.П.

Содержание

1	Практическое руководство	3
1.1	Запуск	3
1.2	Описание простейшего функционала	3
1.3	Общие правила	3
1.4	Операции с числами	6
1.5	Действия с матрицами	7
2	Руководство для программистов	8
2.1	Структура программы	8

1 Практическое руководство

1.1 Запуск

Для установки необходимо установить JVM (Java Virtual Mashine). Для этого заходим на [официальный сайт Java](#), скачиваем версию, соответствующую нашей операционной системе, устанавливаем и перезагружаем компьютер.

После этого можно запускать программу - открываем файл **NeiSys.jar**. После запуска должно появиться следующее окно:

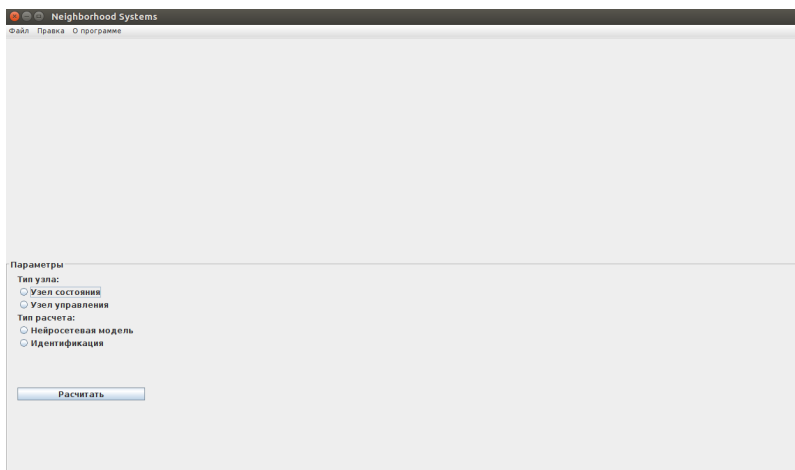


Рисунок 1 - Стартовая панель программы.

Если же данное окно не появилось, вероятнее всего что-то пошло не так при установке Java. Проверьте еще, правильно ли прошла установка. Возможно понадобится прописать переменную `JAVA_HOME` (возможные ошибки при установке Java не такая уж редкость, поэтому решение большинства из них можно прогуглить, например, [настройка системной переменной](#)).

1.2 Описание простейшего функционала

Данная программа предназначена для **для работы с окрестностными системами, параметры (узлов и связей) которых являются скаляры, векторы и матрицы.**

Алгоритм программы следующий: начиная с первого узла в направлении связей просчитывает параметры узлов, и для узлов конечных узлов состояния (State) считает выходной вектор. Конечными мы называем те состояний, которые не имеют выходящие связи. При этом на узел могут оказывать воздействие не только узлы, от которых идет связь, но входящий узел (правая кнопка мыши -> Задать входящий вектор).

Если у какого-либо узла параметры отсутствуют, то в процессе расчета программа должна автоматически решить задачу идентификации для этого узла.

Программа позволяет выполнять простейшие арифметические операции над скалярами и матрицами.

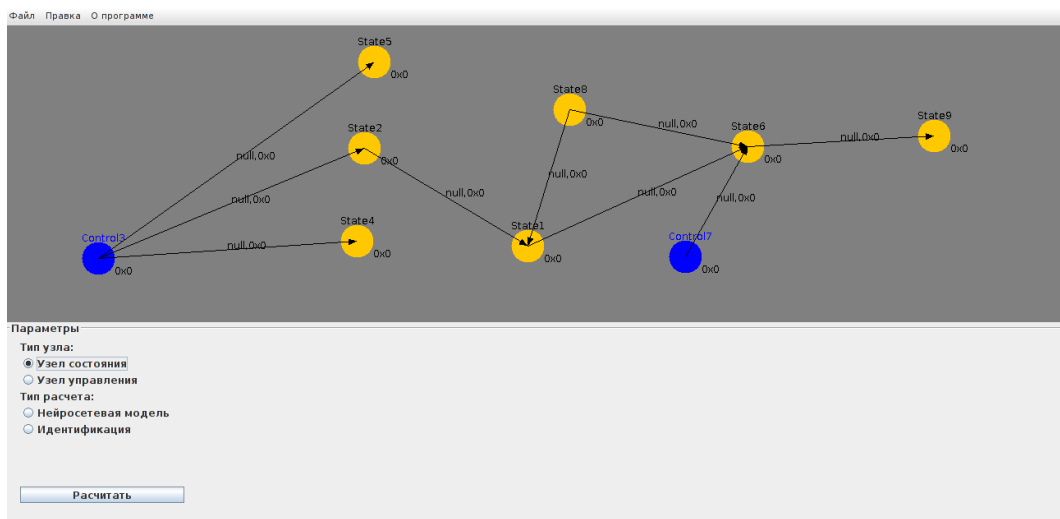


Рисунок 2 - Общий вид.

1.3 Общие правила

При соединении связи между двумя узлами: параметры узла из которого выходят перемножаются на параметры связи и воздействуют на второй узел в зависимости от типа связи: сложением или умножением.

Рассмотрим пример. Пусть имеется два связанных узла:

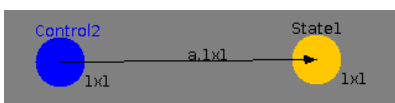
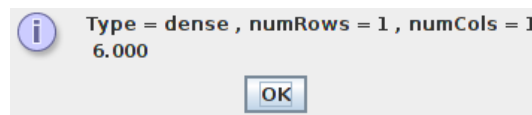


Рисунок 3 - Общий вид.

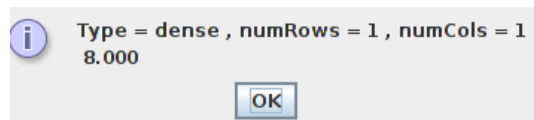
Параметры всех частей системы – скаляры и равны 2. Тогда, если тип связи аддитивный "a" то получим:

$$Out = P(Control2) * P(R) + P(State1) = 2 * 2 + 2 = 6.$$



Если же тип связи "m" (мультипликативный), тогда

$$Out = P(Control2) * P(R) * P(State1) = 2 * 2 * 2 = 8.$$



При этом, если узел имеет несколько входящих связей, то исходящий вектор получается путем суммирования всех входящих параметров. Для примера, добавим к предыдущему случаю еще один узел.

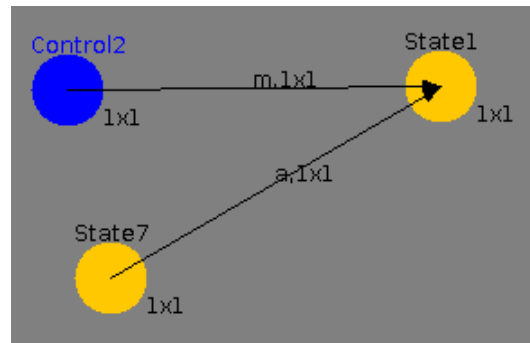
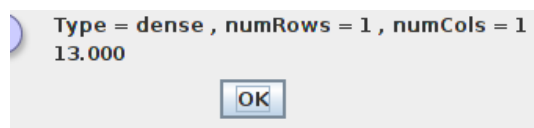


Рисунок 4 - Случай нескольких входящих связей.

Здесь параметр узла State7 равен 3, а параметр его выходящий связи равен 1. Тогда, выходящим вектором для State1 получим:

$$\begin{aligned} Out &= [P(Control2) * P(R_{Control2}) * P(State1)] + [P(State7) * P(R_{State7}) + P(State1)] = \\ &= [2 * 2 * 2] + [3 * 1 + 2] = 8 + 5 = 13. \end{aligned}$$



1.4 Операции с числами

Особенности данной программы позволяют выполнять простейшие арифметические действия над числами: сложение, умножение и деление.

1.4.1 Сложение двух чисел

Для того, чтобы сложить два числа нужно добавить на панель два узла, тип связи выбрать аддитивным и параметром связи указать равным 1. Пример:

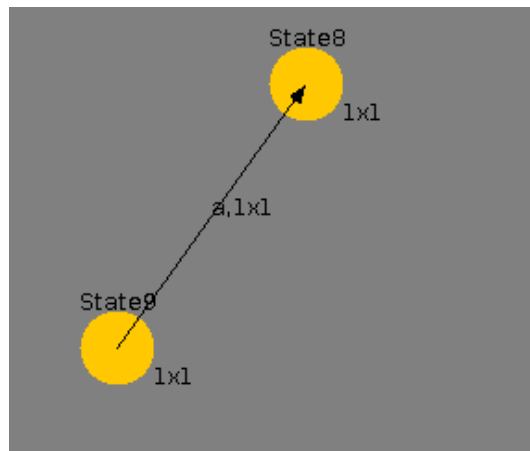


Рисунок 5 - Сложение двух чисел.

Здесь $P(State9) = 3$, $P(R) = 1$, $P(State8) = 2$. Тогда получим:

$$P(State9) * P(R) + P(State8) = 2 * 1 + 3 = 5.$$

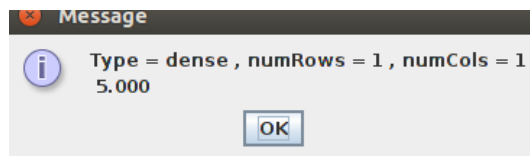


Рисунок 6 - Сложение двух чисел.

1.4.2 Умножение двух чисел

Для того, чтобы умножить два числа нужно выполнить все действия как и для сложения, только вместо аддитивной связи выбрать мультипликативную. Тогда пример выше даст следующее:

$$P(State9) * P(R) * P(State8) = 2 * 1 * 3 = 6.$$

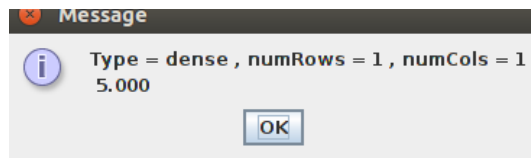


Рисунок 7 - Умножение двух чисел.

1.4.3 Деление двух чисел

Деление двух чисел в данной программе отличается от сложения и умножения. Здесь деление реализовано через решение уравнения $ax = b$. Для того, чтобы найти b/a нужно добавить всего один узел. И задать для него не параметры, а входящий вектор a и исходящий вектор b . При этом на экране у узла должны появиться параметры "I" и "O" с указанием размерности вектором.

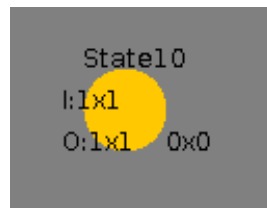


Рисунок 8 - Деление двух чисел.

Далее тип расчета отмечаем "Идентификация" и нажимаем "Расчитать". В качестве примера возьмем входящий вектор равным 3, а исходящий – 6. В результате расчета:

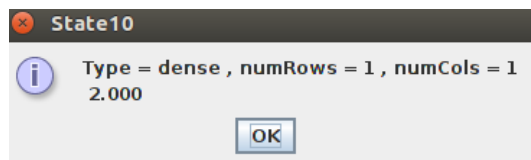


Рисунок 9 - Деление двух чисел.

1.5 Действия с матрицами

В данной программе действия с матрицами аналогичны действиям с числами. Единственное, что стоит упомянуть отдельно, что в случае задачи идентификации узел будет выдавать матрицу перехода в евклидовом базисе в векторном случае и псевдообратную в матричном.

2 Руководство для программистов

2.1 Структура программы

UML-диграмма:

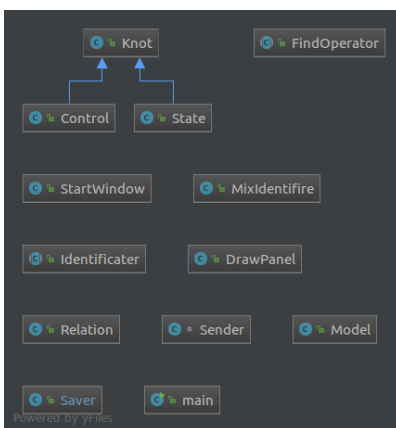


Рисунок 10 - UML-диаграмма.