

*Санкт-Петербургский Государственный Научно
Исследовательский Университет Информационных
Технологий Механики и Оптики*

*СИНТЕЗ РЕГУЛЯТОРОВ С ПОМОЩЬЮ
ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ*

Автор

_____ *Кононенко С. А.*

“ _____ ” _____

Научный руководитель

_____ *Бойков В. И.*

“ _____ ” _____

Заведующий кафедрой

_____ *Бобцов А. А.*

“ _____ ” _____

СОДЕРЖАНИЕ

Введение	2
1 Формулировка проблемы	2
2 Программное обеспечение синтеза регулятора	3
3 Проверка работоспособности программного обеспечения	7
Выводы	9
Список литературы	9

ВВЕДЕНИЕ

При разработке современных систем управления возникает проблема отсутствия явного аналитического решения поставленной задачи. Так при синтезе регулятора, обычно, к качеству системы устанавливают большое число противоречивых требований. Например, к времени переходного процесса, перерегулированию, допустимой ошибке, максимальному уровню сигнала, ограничению допустимых значений коэффициентов передачи, интервалу съема информации и так далее. Такие задачи могут быть решены методом поиска оптимального значения, в частности генетическим алгоритмом (ГА).

В данной работе представлена разработка программного обеспечения (ПО) для синтеза пропорционального регулятора с использованием ГА поиска, а также определены основные свойства и параметры алгоритма.

1 ФОРМУЛИРОВКА ПРОБЛЕМЫ

Рассмотрим объект управления, который описывается системой дифференциальных уравнений первого порядка в форме Коши [3].

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \quad (1)$$

где u - вектор (набор) входных переменных (воздействий);

y - вектор выходных переменных (управляемых величин);

x - вектор переменных состояния (фазовых координат системы);

A - матрица коэффициентов системы, характеризующая её внутренние свойства;

B - матрица входных коэффициентов (матрица управления);

C - матрица выходных коэффициентов.

Пропорциональный регулятор формирует управление

$$u(t) = Kx(t). \quad (2)$$

Управление пропорционально измеряемому составляющему вектора состояния $x(t)$. Задача синтеза пропорционального регулятора является поиск коэффициентов закона управления K , при котором система будет удовлетворять заданным техническим требованиям.

Будем синтезировать регулятор, исходя из следующих требований:

а) Время переходного процесса t_s — должно быть минимально;

б) Задана величина допустимой ошибки системы, по одной из компонент вектора состояния (выходной величине)

$$e < e_{max}, \quad (3)$$

в) Заданы ограничения на области допустимых значений коэффициентов передачи регулятора

$$K < K_{imax}, \quad (4)$$

Необходимо разработать процедуру поиска решения поставленной задачи с использованием ГА.

2 ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ СИНТЕЗА РЕГУЛЯТОРА

Основным в задаче реализации генетического алгоритма является реализация целевой функции. Для данной задачи была разработано специальное ПО для моделирования динамических систем на основе GNU Scientific Library [1, 2].

Языком разработки ПО был выбран язык C. Так как язык C не является объектно ориентированным языком, классы и наследование симулировано с помощью структур и функций обработки каждой структуры. ПО состоит из двух частей: библиотека инструментов для моделирования динамики системы и библиотека компонентов алгоритма поиска.

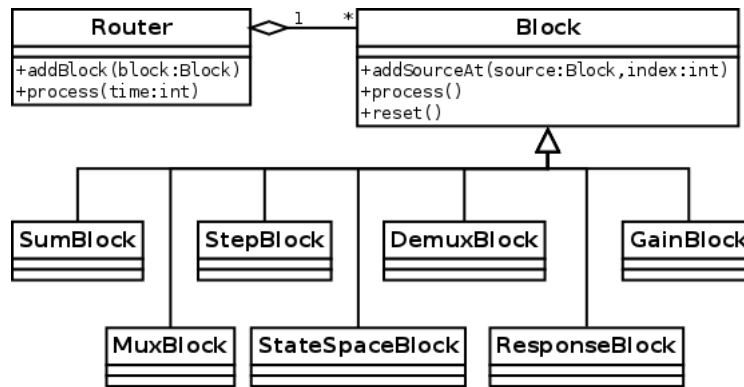


Рисунок 1 – Диаграмма классов системы моделирования.

Инструменты моделирования позволяют вычислить переходную характеристику системы с конкретным регулятором и проанализировать ее показатели качества. Основными сущностями библиотеки моделирования представлены на рисунке 1, ими являются "роутер" и "блок".

Роутер представляет собой сущность, которая управляет временем моделирования и, в зависимости от топологии блоков, приемопередатчей сигналов каждого блока.

Блоки в свою очередь организуют топологию передачи сигнала. Конкретные блоки по-разному реализуют обработку сигнала и имеют свои собственные параметры входов и выходов. Каждый блок в зависимости от конкретной реализации обрабатывает входной и генерирует выходной сигнал. Если одна из функций блока не реализована, то он является либо только источником, либо только приемником (стоком) сигнала. Для успешного моделирования системы, должен существовать хотя бы один блок-приемник.

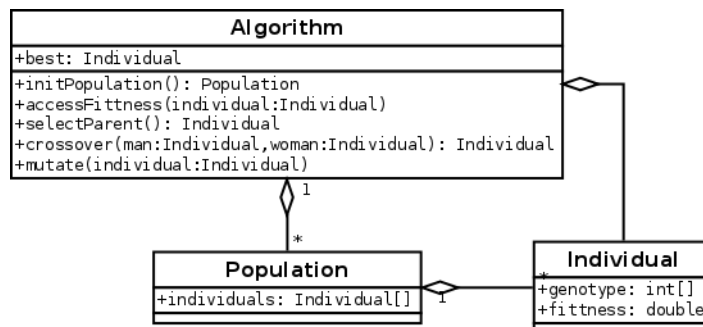


Рисунок 2 – Диаграмма классов генетического алгоритма.

Компоненты генетического алгоритма поиска позволяют осуществить поиск оптимального решения комбинируя элементы случайного и градиентного поиска [4].

Основные сущности ГА представлены на рисунке 2. Поиск реализуется с помощью трёх сущностей: алгоритм, популяция и особь. Популяция представляет собой множество особей, каждая из которых содержит генотип — кандидат на решение задачи поиска. Алгоритм — содержит собственно логику поиска оптимальных решений, логика поиска представлена на рисунке 3 и содержит в себе следующие основные действия [4]:

а) Генерация исходной популяции;

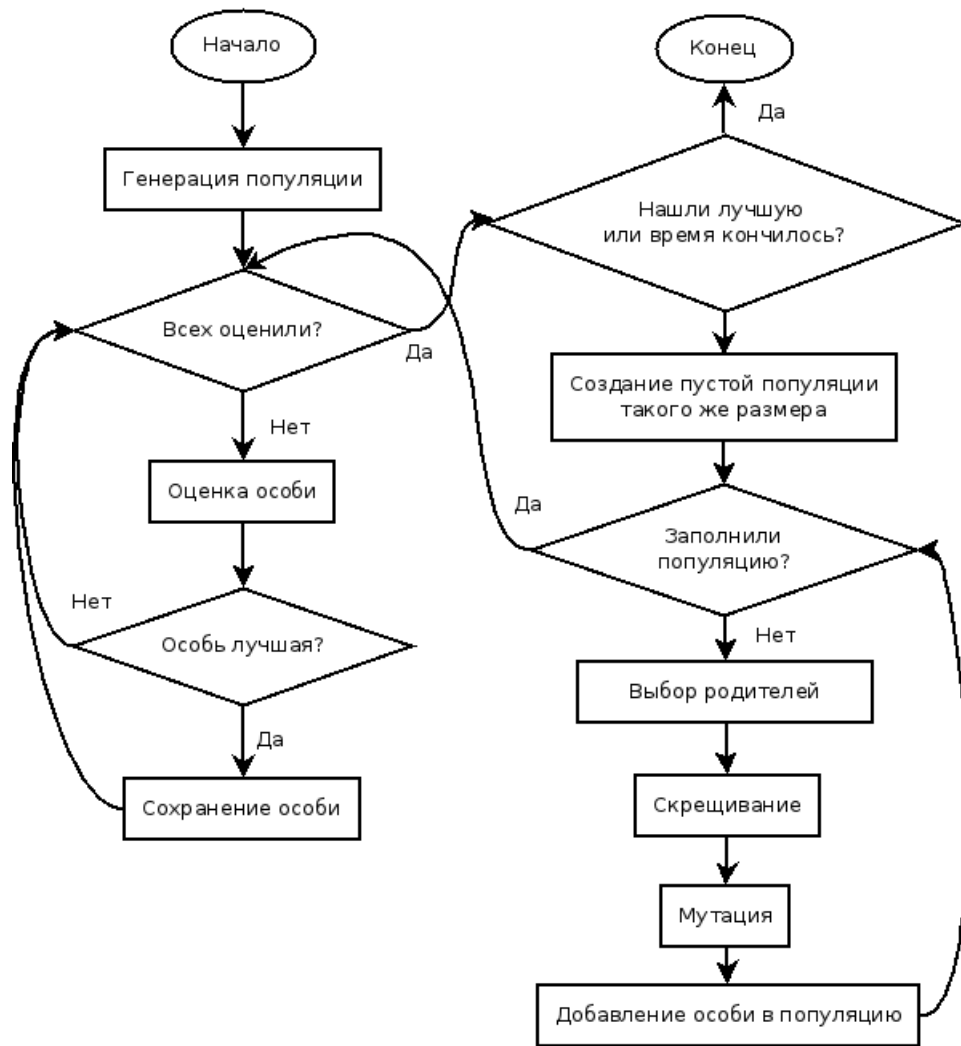


Рисунок 3 – Генетический алгоритм.

- б) Оценка особи;
- в) Выбор родителя;
- г) Скрещивание;
- д) Мутация.

Для конкретной задачи генерация исходной популяции представляет собой выбор N-ного числа случайных векторов из области

$$K_o = [-K_{imax}, K_{imax}], \quad (5)$$

что соответствует условию 4.

Для оценки особи производится моделирование системы представленной на рисунке 4, на которой блоки усиления K1 и K2 соответствуют составляющим вектора K . После моде-

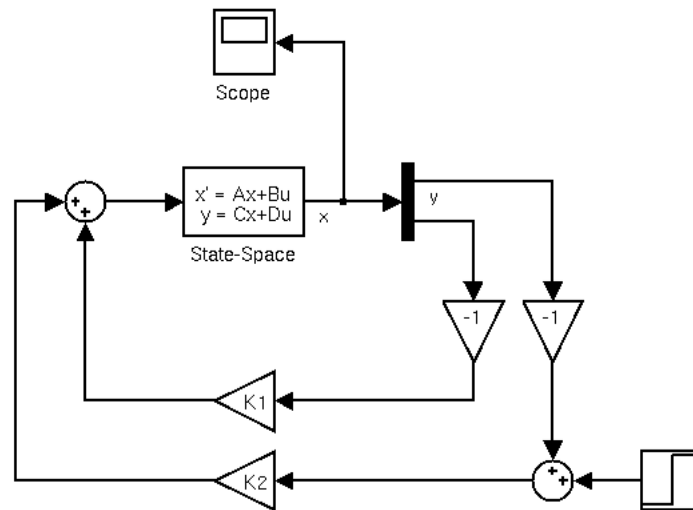


Рисунок 4 – Схема моделирования системы.

лирования с помощью блока анализа переходной функции определяется время стабилизации. Границы сигнала для времени стабилизации соответствуют условию максимальной ошибки 3.

Полученное время переходного процесса используется для определения приспособленности особи по формуле:

$$F = \left(\frac{1}{t_s} + c\right)^2, \quad (6)$$

где c - константа, устанавливающая минимальную приспособленность особи.

Выбор родителя в генетическом алгоритме должен решать две основные задачи: сохранение лучших особей и избежание вырождения. Для решения обеих задач используется следующий алгоритм. Каждой особи на числовой оси ставится в соответствие отрезок, размер которого пропорционален оценки особи. Затем случайным образом выбирается число на этой числовой оси. Особь которой соответствует выбранный отрезок выбирается для скрещивания.

Для скрещивания, в связи с малыми размерами генотипа, был выбран пособ случайной перестановки составляющих родительских векторов.

Мутация выражается формулой:

$$K_{im} = K_i * (1 + m * R), \quad (7)$$

где K_{im} — новое значение составляющей вектора,

K_i — старое значение составляющей вектора,

R — равномерно распределённая в интервале $[-1, 1]$ случайная величина,

m — коэффициент мутации,

при условии что новые значения удовлетворяет условию 4.

3 ПРОВЕРКА РАБОТОСПОСОБНОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Работоспособность разработанного ПО была проверена на задаче синтеза пропорционального регулятора для объекта управления второго порядка. В результате синтеза необходимо найти значения коэффициентов закона управления $2 K = [K_1, K_2]$, соответствующие условиям 4 и 3:

$$|K_1| < 100 \quad (8)$$

$$|K_2| < 100 \quad (9)$$

$$e < 1\% \quad (10)$$

В ходе проведения машинных экспериментов было отмечено следующее.

Время расчета влияет на оценку приспособленности особи в популяции. Уменьшение этого времени ведет к увеличению локальности поиска так как особи с большим временем переходного процесса окажутся оценены так же, как и особи с неустойчивым характером поведения. Время расчета переходного процесса варьировалось от 10 до 30 с.

Шаг моделирования очевидным образом влияет на точность оценки, поэтому принципиально не влияет на оптимальное решение. В данной работе шаг составил 0.05 с.

Важным параметром алгоритма является число особей в популяции. Основная задача этого параметра – выявить в первой популяции особей с оценкой выше минимальной. При выполнении данного условия, градиентная составляющая поиска получит возможность направленного движения. Для поиска решений в области заданной ограничениями 8 и 9 достаточное число особей равно 100.

Минимальная приспособленность особи параметр влияет на возможность особи участвовать в скрещивании. При существенном увеличении данного параметра, поиск смещается в сторону случайного поиска, что увеличивает время поиска решения. В данной работе $c = 0.001$.

Коэффициент мутации является основным параметром градиентной составляющей поиска. Диапаон возможной мутации влияет на плавность схождения к результату. В данной работе $m = 0.05$.

Время поиска является основным ограничивающим фактором цикла поиска решений. Поэтому для различных типов сходимости к результату выбирается подходящее время. В проведенных экспериментах время варьировалось от 1 до 3 минут. За это время проводилось от 5 до 10 генераций новых популяций. В зависимости от них меняется скорость поиска и качество найденного решения.

На рисунках 5, 6 и 7 ипоказаны типовые переходные характеристики синтезируемых регуляторов для трех объектов с различной динамикой.

$$A_1 = \begin{vmatrix} -0.45 & 1 \\ -0.05 & 0 \end{vmatrix}, \quad B_1 = \begin{vmatrix} 0 \\ 0.1 \end{vmatrix}. \quad (11)$$

$$A_2 = \begin{vmatrix} 1 & 5 \\ 2 & 1 \end{vmatrix}, \quad B_2 = \begin{vmatrix} 0 \\ 1 \end{vmatrix}. \quad (12)$$

$$A_3 = \begin{vmatrix} 0 & 1 \\ -15 & -5 \end{vmatrix}, \quad B_3 = \begin{vmatrix} 0 \\ 1 \end{vmatrix}. \quad (13)$$

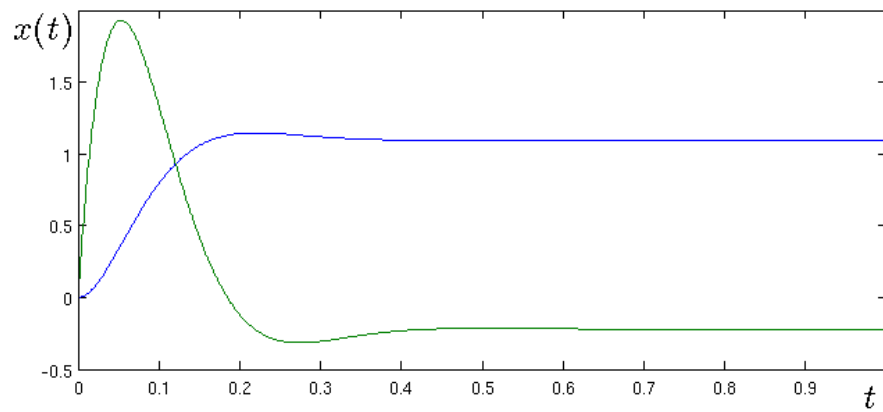


Рисунок 5 – Моделирование системы 11 с регулятором.

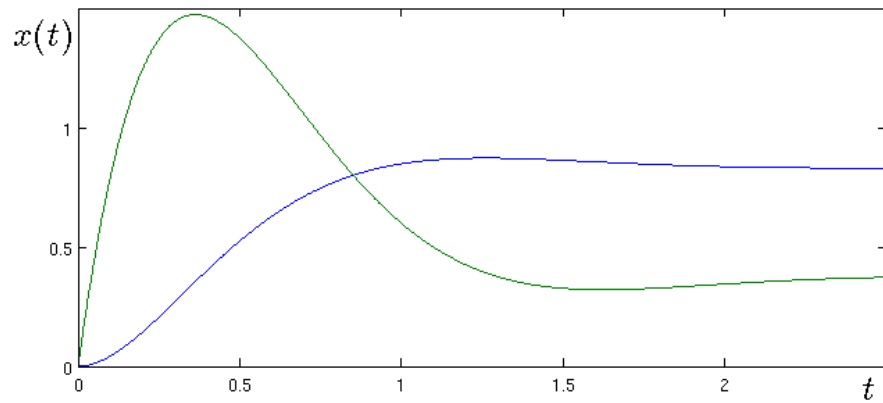


Рисунок 6 – Моделирование системы 12 с регулятором.

При различии времени переходного процесса на порядок время поиска оптимального решения остается примерно одинаковым.

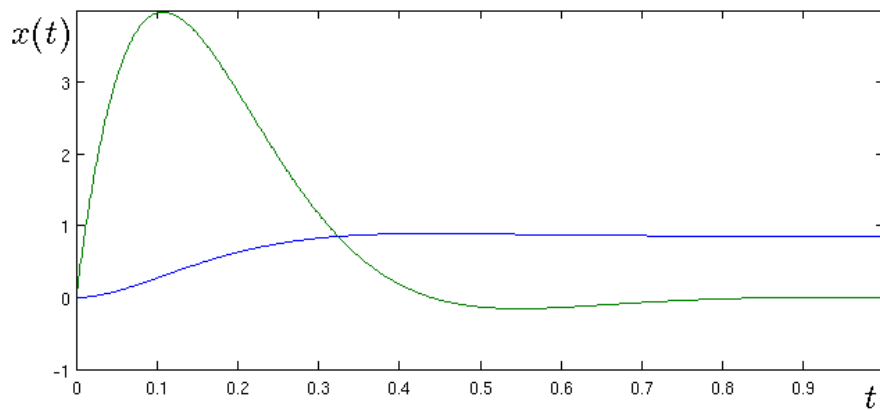


Рисунок 7 – Моделирование системы 13 с регулятором.

ЗАКЛЮЧЕНИЕ

Разработанное программное обеспечение позволяет производить синтез регуляторов на базе генетических алгоритмов. Алгоритм сходится при условии наличия оптимального решения. Конечное время сходимости обеспечивает наличие в первой популяции особи с оценкой выше минимальной.

Полученный результат свидетельствует о возможности дальнейшего развития и применения генетических алгоритмов в задачах управления.

СПИСОК ЛИТЕРАТУРЫ

1. Кононенко С. Система моделирования динамических систем. <https://github.com/kononencheg/Air-Fish>.
2. Проект GNU. Научная библиотека gnu. <http://www.gnu.org/software/gsl/>.
3. Мирошник И. В. *Теория автоматического управления. Линейные системы*. 2005.
4. Люк Ш. *Основы метаэвристик*. 2009.