Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Кафедра программного обеспечения информационных технологий

ОТЧЁТ

по лабораторной работе №3

по курсу «Системный анализ и машинное моделирование»

на тему

«Построение аналитической модели дискретно-стохастической системы»

Вариант 1

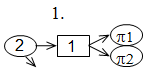
|  |  |
| --- | --- |
| Проверил:  Мельник Н.И. | Выполнил:  студент гр. 251004  Маршин А.С. |

Минск 2015

1. ЦЕЛЬ РАБОТЫ

Построение аналитической модели дискретно-стохастической системы и расчет параметров данной системы с использованием построенной модели.

1. ИСХОДНЫЕ ДАННЫЕ К РАБОТЕ



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Вариант | ρ | π1 | π2 | Цель исследования |
| 1 | - | 0,8 | 0,6 | Ротк, А ,Wоч |

1. РАСЧЕТ МОДЕЛИ Р-СХЕМЫ

Граф состояний кодируется четырехкомпонентным вектором *abcd,* где *a* – время до выдачи очередной заявки источником, *a*={2,1}; *b* – количество заявок, находящихся в накопителе (длина очереди), *b*={0,1}; *c* и *d* – состояние каналов обслуживания, принимает следующие значения: 0 – канал свободен, 1 – канал занят обслуживанием заявки.

Граф состояний представлен на рис.1.

Исходя из графа состояний, построим СЛАУ для нахождения вероятностей состояний:



и дополним ее нормировочным уравнением 

P2 = (1- π1)P3+(1- π1)(1- π2)P5+P1;

P3 = (1- π1)P4+(1- π1)(1- π2)P6+(1- π2)P7+P2;

P4 = π1P3+ π1(1- π2)P5+(1- π1)(1- π2)P8;

P5 = π1P4+ π1(1- π2)P6+ π2(1- π1)P6+ π2P7+(1- π1)(1- π2)P9;

P6 = π1π2P5+ π1(1- π2)P8+ π2(1- π1)P8;

P7 = π2(1- π1)P5;

P8 = π1π2P6+ π2(1- π1)P9+ π1(1- π2)P9+ π1π2P9;

P9 = π1π2P8;

P1+P2+P3+P4+P5+P6+P7+P8+P9=1;

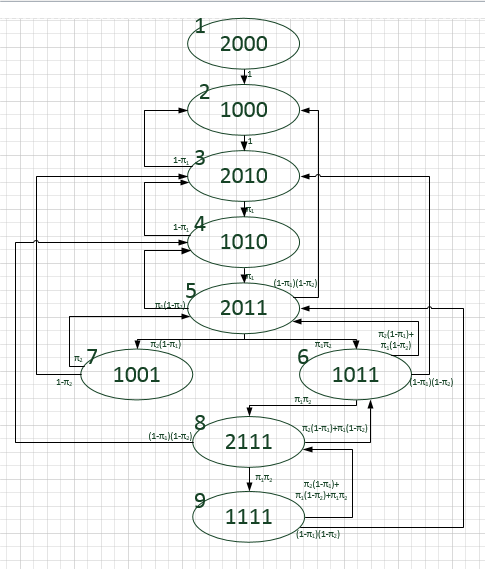


Рис.1. Граф состояний СМО.

Подставив в СЛАУ значения π1 = 0,8 и π2 = 0,6 и приведя СЛАУ к каноническому виду, получим:

P1 - P2 + 0,2P3 + 0,08P5 = 0;

P2 - P3 + 0,2P4 + 0,08P6 + 0,4P7 =0;

0,8P3 - P4 + 0,32P5 + 0,08P8 = 0;

0,8P4 - P5 + 0,44P6 + 0,6P7 + 0,08P9 = 0;

0,48P5 - P6 + 0,44P8 = 0;

0,12P5 - P7 =0;

0,48P6 - P8 + 0,92P9 = 0;

0,48P8 - P9 =0;

P1+P2+P3+P4+P5+P6+P7+P8+P9=1;

Решив данную СЛАУ с помощью пакета MATLAB, получим значения вероятностей состояний:

P1 = 0;

P2 = 0,03915;

P3 = 0,09945;

P4 = 0,16939;

P5 = 0,24077;

P6 = 0,18587;

P7 = 0,02889;

P8 = 0,15977;

P9 = 0,07669;

Рассчитаем:

– вероятность отказа:

Pотк = P9 \* π1π2 = 0,07669 \* 0,48 = 0,03681;

– абсолютную пропускную способность:

A =∑((1- π1)\*ci + (1- π2)\*di)Pi = (0,2\*0 + 0,4\*0)\*0 + (0,2\*0 + 0,4\*0)\* 0,03915 + (0,2\*1 + 0,4\*0)\* 0,09945+… = 0 + 0 + 0,01989 + 0,03388 + 0,14446 + 0,11152 + 0,09586 + 0,04601 + 0,01156 = 0,46319;

– среднюю длину очереди (необходимо для вычисления среднего времени ожидания в очереди):

Lоч = P8 + P9 = 0,15977 + 0,07669 = 0,23646;

– среднее время ожидания в очереди:

Wоч = Lоч/A = 0,23646/0,46319 = 0,51050;

1. ВЫВОДЫ

В данной работе была исследована дискретно-стохастическая модель (Р-схема). Для нее был построен граф состояний, на основе которого затем была составлена СЛАУ, из которой были получены вероятности каждого из состояний системы. Были посчитаны необходимые характеристики системы, которые затем будут сравнены с результатами, получившимися при имитационном моделировании.

1. КОД ПРОГРАММЫ

Модуль MainForm.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.IO;

namespace SAMM\_3

{

public partial class MainForm : Form

{

Random rnd;

const int numtacts = 100000;

List<Label> lbs = new List<Label>();

public MainForm()

{

InitializeComponent();

rnd = new Random();

#region lbsadd

lbs.Add(label5);

lbs.Add(label6);

lbs.Add(label7);

lbs.Add(label8);

lbs.Add(label9);

lbs.Add(label10);

lbs.Add(label11);

lbs.Add(label12);

lbs.Add(label13);

#endregion

}

private void button1\_Click(object sender, EventArgs e)

{

int[] prob = new int[9];

Scheme qs = new Scheme(Double.Parse(textBox1.Text), Double.Parse(textBox2.Text));

int[] firstValue = new int[4] {2, 0, 0, 0 };

int[] nv = qs.Tact(firstValue);

double MeanTP=0;

for (int i = 0; i <= numtacts; i++)

{

int t = nv[0];

nv = qs.Tact(nv);

if ((nv[0] == 2) && (t == 1)) MeanTP++;

string s = Convert.ToString(nv[0]) + Convert.ToString(nv[1]) + Convert.ToString(nv[2]) + Convert.ToString(nv[3]);

switch (s)

{

case "2000":

prob[0]++;

break;

case "1000":

prob[1]++;

break;

case "2010":

prob[2]++;

break;

case "1010":

prob[3]++;

break;

case "2011":

prob[4]++;

break;

case "1011":

prob[5]++;

break;

case "1001":

prob[6]++;

break;

case "2111":

prob[7]++;

break;

case "1111":

prob[8]++;

break;

}

}

for (int j =0;j<=8;j++)

{

lbs[j].Text = Convert.ToString(j + 1) + " | " + Convert.ToString((double)prob[j] / numtacts);

}

MeanTP -= nv[1];

MeanTP -= nv[2];

MeanTP -= nv[3];

MeanTP -= qs.otkaz;

label3.Text = "A="+Convert.ToString(MeanTP / numtacts);

label4.Text ="Pотк="+Convert.ToString((Double)qs.otkaz/ numtacts);

label14.Text = "Lоч=" + Convert.ToString((Double)qs.w / numtacts);

label15.Text = "Wоч=" + Convert.ToString((Double)qs.w/MeanTP);

}

}

}

Модуль Scheme.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace SAMM\_3

{

class Scheme

{

double \_pi1, \_pi2;

Random rnd;

public Int32 otkaz,

w;

public Scheme(double pi1, double pi2)

{

\_pi1 = pi1;

\_pi2 = pi2;

rnd = new Random();

}

public int[] Tact(int[] prevState)

{

int[] newState = new int[4];

int pi1Prob = rnd.Next(100) + 1;

int pi2Prob = rnd.Next(100) + 1;

if ((pi1Prob > \_pi1 \* 100) && (prevState[2] == 1))

newState[2] = 0;// заявка в первом канале выполнена

else if (prevState[2] == 1)

newState[2] = 1;

if ((pi2Prob > \_pi2 \* 100) && (prevState[3] == 1))

newState[3] = 0;// заявка во втором канале выполнена

else if (prevState[3] == 1)

newState[3] = 1;

if (prevState[0] == 1 && prevState[1] == 1 && newState[2] == 1 && newState[3] == 1)

{

otkaz++;

}

switch (prevState[1])

{

case 0:

break;

case 1:

if (newState[2] == 0)

newState[2] = 1;

else

{

if (newState[3] == 0)

newState[3] = 1;

else

newState[1] = 1;

}

break;

default:

break;

}

if (prevState[0] == 2)

{

newState[0] = 1;

}

else

{

if (prevState[0]==1)

{

if (newState[1] != 1)

{

if (newState[2] == 0)

{

newState[2] = 1;

}

else if (newState[3] == 0)

{

newState[3] = 1;

}

else

{

newState[1] = 1;

}

}

newState[0] = 2;

}

}

if (newState[1] == 1) w++;

return newState;

}

}

}