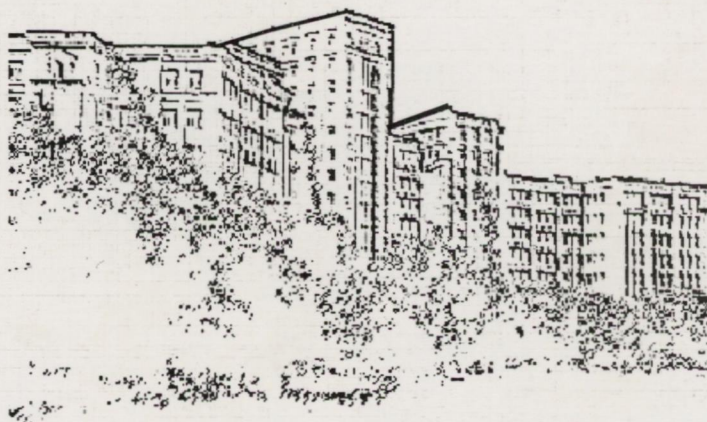


ВІСНИК

Харківського національного
університету



№ 733

Харків
2006

K-14038

П330469

V.N. Karazin Kharkiv National University



00563079

6

ISSN 0453-8048

Міністерство освіти та науки України
Харківський національний університет імені В. Н. Каразіна

Заснований у 1965 р.

ВІСНИК

Харківського національного
університету



№ 733

Серія

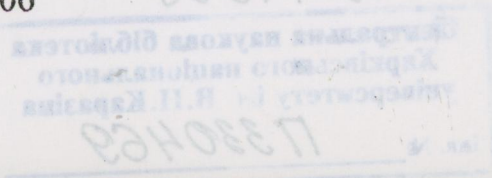
«Математичне моделювання.

Інформаційні технології.

Автоматизовані системи управління»

Випуск 6

Харків
2006



УДК 519.6+004.652/.942+519.217:681.3+621.793.184.06+378.1:159.816

До випуску увійшли статті, присвячені дослідженням у галузі математичного моделювання, які відкривають нові перспективи опису і керування фізичними, технічними, інформаційними та економічними процесами, а також дослідження з інформаційних технологій, зокрема таких, які націлені на підтримку досліджень і застосувань в галузі математичного моделювання.

Збірник включає статті, що були подані до розгляду редакційної колегії у період з 18.12.2005 по 15.06.2006 року.

Для викладачів, наукових працівників, аспірантів, працюючих у відповідних або суміжних напрямках.

Редакційна колегія:

Азаренков М.О., (гол. редактор), д.ф.-м.н., член-кор. НАН України, проф., ІВТ ХНУ імені В.Н. Каразіна

Гандель Ю.В., д.ф.-м.н., проф., ММФ ХНУ імені В.Н. Каразіна

Жолткевич Г.М., (заст. гол. редактора), д.т.н., проф. ММФ ХНУ імені В.Н. Каразіна

Золотарьов В.О., д.ф.-м.н., проф., ММФ ХНУ імені В.Н. Каразіна

Куклін В.М., д.ф.-м.н., проф., ІВТ ХНУ імені В.Н. Каразіна

Лазурик В.Т., д.ф.-м.н., проф., ІВТ ХНУ імені В.Н. Каразіна

Любарський М.Г., д.ф.-м.н., проф., ММФ ХНУ імені В.Н. Каразіна

Мацевитий Ю.М., д.т.н., академік НАН України, проф., фізико-енергетичний ф-т ХНУ імені В.Н. Каразіна

Міщенко В.О., (відпов. секретар), к.ф.-м.н., доц., ММФ і ФКН ХНУ імені В.Н. Каразіна

Раскін Л.Г., д.т.н., проф., Національний технічний університет "ХПІ"

Стрельнікова О.О., д.т.н., проф. Ін-т проблем машинобудування НАН України

Руткас А.Г., д.ф.-м.н., проф., ММФ ХНУ імені В.Н. Каразіна

Соколов О.Ю., д.т.н., проф., Національний аерокосмічний університет імені М.Є. Жуковського "ХАІ"

Стервасдов М.Г., к.т.н., доц., ІВТ ХНУ імені В.Н. Каразіна

Целуйко О.Ф., к.ф.-м.н., проф., ФКН ХНУ імені В.Н. Каразіна

Черваньов І.Г., д.т.н., проф., геолого-географічний ф-т ХНУ імені В.Н. Каразіна

Шейко Т.В., д.т.н., проф., фізико-енергетичний ф-т ХНУ імені В.Н. Каразіна

Щербина В.А., д.ф.-м.н., проф., ММФ ХНУ імені В.Н. Каразіна

Адреса редакційної колегії: 61077, м. Харків, пл. Свободи, 4, ХНУ імені В.Н. Каразіна, каф. Математичної фізики та обчислювальної математики, к.6-28

Тел. +38 (057) 707-52-02, Email: Victor.O.Mischenko@univer.kharkov.ua

Сайт: <http://mia.univer.kharkov.ua>

Друкується за рішенням Вченої Ради Харківського національного університету імені В.Н. Каразіна (Протокол № 8 від 30.06.2006 р.)

Свідectво про державну реєстрацію КВ № 4063 від 02.03.2000 р.

© Харківський національний університет імені В.Н.Каразіна, 2006

Ж-14038



Вісник Харківського національного університету
Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»

УДК 681:3.06

№ 733, 2006, с.3-16

Використання jBPM для моделювання бізнес-процесів в розподілених інформаційних системах

М. О. Алексєєв, Л. С. Глоба, Ю. М. Молчанов

*Національний технічний університет України «Київський політехнічний інститут»,
Інститут телекомунікаційних систем, Україна*

Using of jBPM for business-process management of the distributed information telecommunication systems is investigated. The research is based on the experience of organizing the automated information interchange between the software of Central and regional segments of the Chernobyl database of the Ministry of Emergency of Ukraine.

Актуальність використання BPM-технологій

Відомо, що поява нових технологій призводить до дійсно серйозних змін у сфері бізнесу й діловодства. Однією з таких технологій у наш час є автоматизоване керування діловими процесами (BPM).

Одержувати знання й засоби, необхідні для збереження конкурентоспроможності на нових світових ринках, найкраще допомагає співробітництво, формування партнерських відносин. Щоб досягти успіху в цій стратегії, підприємство повинно так представити свій бізнес, щоб залучити потенційних ділових партнерів. При цьому позитивним фактором є відкриття доступу до інформації про свої ділові процеси. Очевидно, що для цього потрібна універсальна мова опису процесів, метод їхнього визначення, так само точний й недвозначний, як мова програмування. З іншого боку, сучасні інформаційні системи, що використовуються підприємствами та компаніями для своїх бізнес-потреб є розподіленими у телекомунікаційному середовищі, складними за своєю архітектурою та інформаційною структурою. Тому дослідження можливостей сучасних BPM-систем щодо моделювання бізнес-процесів в таких системах також є задачею актуальною.

Аналіз існуючих систем керування бізнес-процесами

Оглядовий аналіз сучасних засобів BPM показав [1], що всі вони мають такі функціональні елементи як засоби проектування, движки й засоби моніторингу. По призначенню існуючі рішення BPM діляться на системи підтримки адміністративного керування (Administrative and Task Support), засоби підтримки колективної роботи (Collaborative BPM), прикладні BPM-Компоненти (Application-Specific or Preconfigured BPM), системи, націлені на інтеграцію автоматизованих систем (Integration-Focused BPM). Існують також універсальні рішення (Pure-Play BPM).

Всі існуючі системи BPM ґрунтуються на відповідних стандартах і специфікаціях мов моделювання бізнес-процесів. На даний момент відомо безліч таких стандартів і специфікацій. Найпоширенішими серед них є Business

Process Execution Language (BPEL), Business Process Modeling Language (BPML) [2, 3].

У доступному для огляду майбутньому ні однієї компанії, звичайно, не вдасться запропонувати системи, що задовольняють усім вимогам до системи керування бізнес-процесами розподілених інформаційних телекомунікаційних систем. Але вже сьогодні існують ефективні рішення, що забезпечують основні можливості по впровадженню, виконанню, підтримці й оптимізації, які можна використовувати разом з іншими технологіями, приклад з автоматизацією ділових процесів, керуванням ресурсами підприємства й ін., створивши в такий спосіб інформаційну інфраструктуру керування процесами.

Користувачі систем управління процесами повинні мати можливість стежити за ходом процесів на всіх рівнях і будувати консолі керування, щоб удосконалювати процеси. Їм також необхідні інструменти періодичної перевірки наскрізних процесів для пошуку нових можливостей, створення цілком змінених процесів, служб і продуктів. Основна роль тут за емуляцією.

Вимогам інтегрування, відкритості і здатність масштабуватися — задовольняють BPM-системи, реалізовані на платформі J2EE. Платформа J2EE заснована на відкритих, підтримуваних лідерами IT-галузі, стандартах, що сприяє інтеграції з корпоративними програмними засобами. Багаторівнева архітектура J2EE дозволяє досягати високої продуктивності і надійності за рахунок розподілу навантаження між серверами.

Лідером серед таких систем на сьогоднішній день є система JBPM [4], компанії JBoss. JBoss jBPM є гнучкою, розширюваною системою керування технологічним процесом. JBoss jBPM має інтуїтивну мову процесу для вираження ділових процесів графічно в термінах задач, стану чекання для асинхронної передачі, таймерів, автоматизації дій. Для зв'язку цих операцій JBoss jBPM має потужний і розширюваний механізм потоку управління.

Практична апробація системи JBoss JBPM

Для більш детального вивчення можливостей використання jBPM для моделювання бізнес-процесів в розподілених інформаційних системах, була обрана задача організації автоматизованого обміну інформацією між програмним забезпеченням Центрального і регіонального сегментів Чорнобильського банку даних МНС України. У здійсненій реалізації передбачалася можливість обміну документами і базами даних MS Office у виді файлових архівів, з використанням електронної пошти, через протокол http, а також можливість запуску і виконання функцій web-інтерфейсу.

Реалізація моделі розподіленої інформаційно – телекомунікаційної системи

На першому етапі рішення задачі за допомогою графічного проектувальника jBPM, убудованого в пакет розробки Eclipse, був спроектований граф відповідного бізнес-процесу для регіонального сегмента Чорнобильського банку даних МНС України, представлений на рис. 4.1.

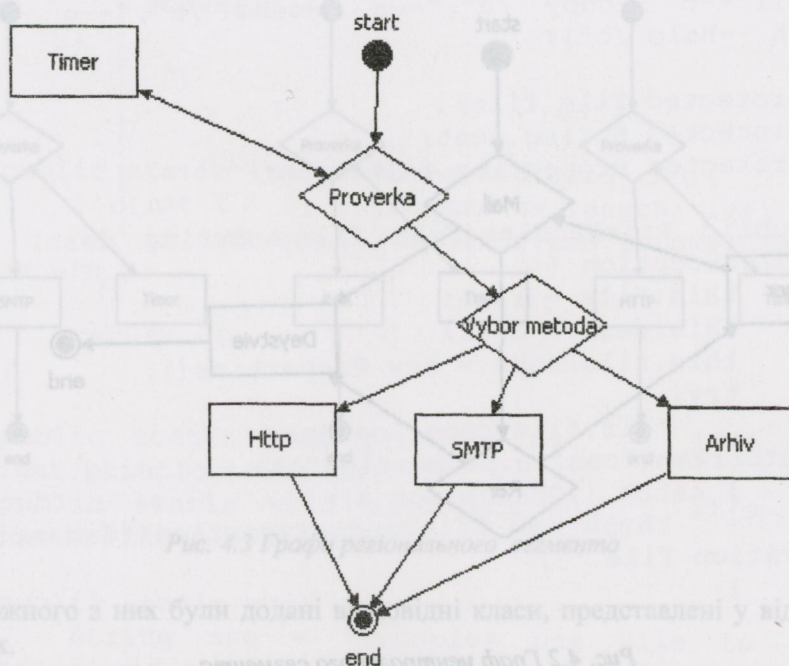


Рис. 4.1 Граф моделі бізнес-процесу регіонального сегмента

Даний граф складається з 8 вузлів і 6 переходів. Граф починається зі стартового вузла. Далі впливає вузол, у якому відбувається перевірка вмісту заздалегідь зазначеної папки на наявність у ній необхідного для продовження роботи файлу. У випадку, якщо такий мається, маркер виконання переходить на наступний вузол. Якщо файл відсутній, то виконання графа переходить у таймер, по витіканню котрого, виконання графа відновляється з моменту перевірки вмісту папки. Далі впливає вузол, на вході якого мається «екшен», що пропонує користувачеві вибрати один із трьох варіантів продовження роботи з файлом:

- заархівувати файл і перемістити в заздалегідь визначений каталог;
- відправити файл за допомогою електронної пошти за допомогою SMTP протоколу;
- передати файл по HTTP протоколу.

Далі відбувається обрана користувачем дія, після чого маркер графа переходить у кінцевий стан.

На другому етапі рішення задачі за допомогою графічного проектувальника jBPM, убудованого в пакет розробки Eclipse, був спроектований граф відповідного бізнес-процесу для центрального сегмента Чорнобильського банку даних МНС України, представлений на рис. 4.2

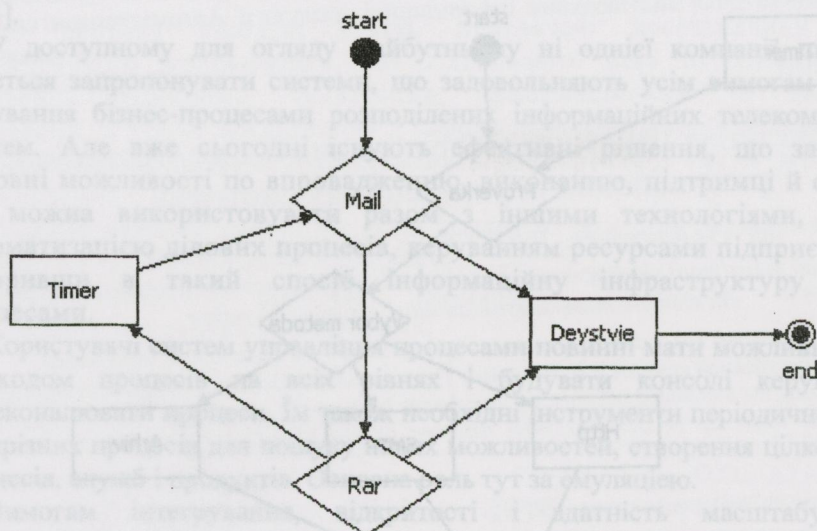


Рис. 4.2 Граф центрального сегмента

Даний граф складається з 6 вузлів і 6 переходів. Граф починається зі стартового вузла. Далі впливає вузол, у якому відбувається перевірка наявності у відповідній поштової скриньці необхідного для продовження роботи по даній гілці графа листа з прикріпленим файлом. У випадку, якщо такий мається, маркер виконання переходить на вузол, у якому виконується необхідне переміщення файлу. Якщо лист відсутній, то виконання графа переходить у вузол, у якому виконується перевірка наявності файлового архіву. Якщо архів відсутній, то маркер переходить у таймер, по витіканню якого виконання графа відновляється з моменту перевірки електронної пошти.

Третій етап практичної частини полягав у додаванні до графів обох сегментів коду, написаного мовою Java з метою надавання графові необхідних програмних функціональних можливостей

Так, для регіонального сегмента потрібно додати 4 «екшена», реалізованих у виді класів мови Java. Однак на даному етапі виникла складність написання «екшена» реалізуючу функцію вибору користувачем методу доставки файлу в Центральний сегмент, тому що даний «екшен» може бути реалізований тільки мовою BeanShell.

Вихід був знайдений шляхом поділу графа на 3 окремих підграфи, кожний з яких реалізує окремий метод доставки файлу, що відображено на рис. 4.3.

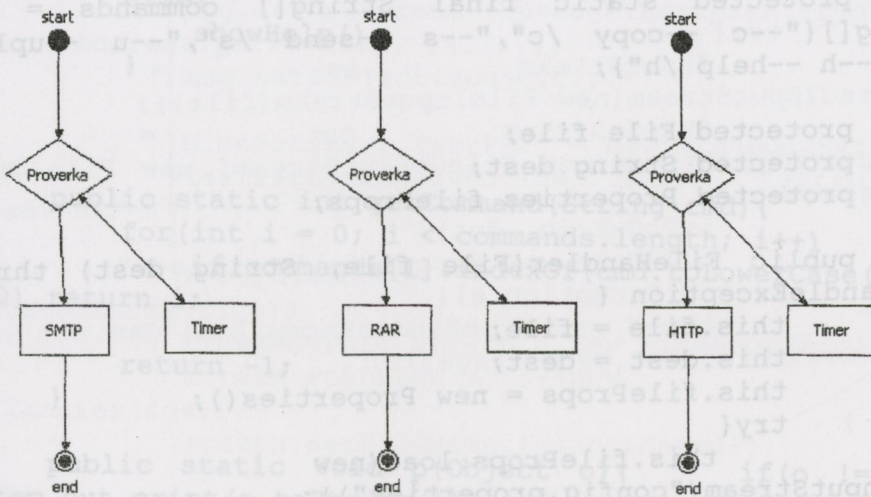


Рис. 4.3 Графи регіонального сегмента

До кожного з них були додані відповідні класи, представлені у відповідних лістингах.

Лістинг коду класа FileHandleException.class

```

public class FileHandleException extends Exception{
    public FileHandleException(String message) {
        super(message);
    }
}
  
```

Лістинг коду класа FileHandler.class

```

import javax.mail.internet.*;
import javax.mail.Message;
import javax.mail.Transport;
import javax.mail.Session;
import javax.mail.Provider;
import javax.activation.FileDataSource;
import javax.activation.DataHandler;
import java.io.*;
import java.util.Properties;
import java.net.Socket;

public class FileHandler {
  
```



```
protected static final String[] commands = new
String[]{"--c --copy /c", "--s --send /s", "--u --upload
/u", "--h --help /h"};
```

```
protected File file;
protected String dest;
protected Properties fileProps;
```

```
public FileHandler(File file, String dest) throws
FileHandleException {
    this.file = file;
    this.dest = dest;
    this.fileProps = new Properties();
    try{
        this.fileProps.load(new
FileInputStream("config.properties"));
    } catch (IOException e) {
        throw new FileHandleException("Can not load
configuration file.");
    }
}
```

```
public static void main(String[] args){
    if(args.length == 3){

        try{

            File file = new File(args[1]);
            if(! file.exists()) throw new
FileHandleException("File \""+ args[1] +"\" not found.");

            FileHandler fh = new FileHandler(file,
args[2]);

            switch(getCommand(args[0])){
                case 0: fh.copyFile(); break;
                case 1: fh.sendFile(); break;
                case 2: fh.uploadFile(); break;
                default: showHelp();
            }
        }catch(Exception e){
            pl(e.getMessage());
            pl("\nUse \"java FileHandler -h\" for
help.");
        }
    }
}
```

```

else{
    showHelp();
}

}

public static int getCommand(String cmd){
    for(int i = 0; i < commands.length; i++){
        if(commands[i].indexOf(cmd.toLowerCase())
        >= 0) return i;
    }
    return -1;
}

public static void p(Object o){        if(o !=null)
System.out.print(o.toString());    }
public static void pl(Object o){        if(o !=null)
System.out.println(o.toString());    }

public static void showHelp(){
    String msg = "\nCopies one file to another
location, sends by e-mail or uploads to file server via
TCP/IP connection.\n" +
        "\nUsage:  java FileHandler [command]
[file] [destination]\n" +
        "\n  command:\t" +
        "\n    \t-c copy    \t" + "Copies file to
destination on local system." +
        "\n    \t-s send  \t" + "Sends file by e-
mail      (need    to    specify    SMTP-host    in
\"config.properties\")."+
        "\n    \t-u upload\t" + "Uploads file to
the server."+
        "\n    \t-h help  \t" + "Shows this
message."+
        "\n    file      \t" + "Specifies the
file to be hanled." +
        "\n    destination\t" + "Specifies the
directory for the new file, e-mail for sending or host
for uploading." +
        "\n\nExample:  java FileHandler -u
\"C:\\My Documents\\Read Me.txt\" 127.0.0.1";
    pl(msg);
}

public void copyFile() throws FileHandleException {

```



```

    try{
        InputStream      in      = new
BufferedInputStream(new FileInputStream(file));
        OutputStream     out     = new
BufferedOutputStream(new FileOutputStream( new File(dest,
file.getName()) ));
        inToOut(in, out);
        pl("File was successfully copied.");
    }catch(Exception e){
        throw new FileHandleException("Can not copy
file. Reason: " + e.getMessage());
    }
}

```

```

public void sendFile() throws FileHandleException {
    try{
        String      host      =
fileProps.getProperty("mail.smtp_host");
        String      mailfrom  =
fileProps.getProperty("mail.from");
        String      mailtext  =
fileProps.getProperty("mail.text");
        if(mailtext == null)
            mailtext = "";
        String      subject   =
fileProps.getProperty("mail.subj");
        if(subject == null)
            subject = "File sedned by application
\\FileHandler\\";

        Properties props = new Properties();
        props.put("mail.transport.protocol", "smtp");
        props.put("mail.smtp.host", host);

        Session sess = Session.getInstance(props,
null);

        MimeMessage msg = new MimeMessage(sess);

        msg.setFrom(new      InternetAddress(mailfrom)
);
    }
}

```

```

        msg.setRecipients(Message.RecipientType.TO,
InternetAddress.parse(dest, false));
        msg.setSubject(subject);

        MimeBodyPart textPart = new MimeBodyPart();
        textPart.setContent(mailtext, "text/plain;
charset=UTF-8");

        FileDataSource fds = new
FileDataSource(file);
        MimeBodyPart attach = new MimeBodyPart();
        attach.setDataHandler(new
DataHandler(fds));
        attach.setFileName(fds.getName());

        MimeMultipart content = new
MimeMultipart();
        content.addBodyPart( textPart );
        content.addBodyPart( attach );
        msg.setContent(content);

        Transport.send(msg);

        pl("File was sended by e-mail.");

    } catch (AddressException e) {
        throw new FileHandleException("Wrong e-mail
address.");
    } catch (Exception e){
        throw new FileHandleException("Can not send
e-mail, check your mail configuration.");
    }

}

public void uploadFile() throws FileHandleException
{
    try{
        Socket socket = new Socket(dest,
Integer.parseInt(fileProps.getProperty("fileservr.port")
));

        OutputStream out =
socket.getOutputStream();

```



```

        InputStream in = new
        BufferedInputStream(new FileInputStream(file));
        OutputStreamWriter nameWriter = new
        OutputStreamWriter(out);
        nameWriter.write(file.getName()+"\n");
        nameWriter.flush();

        inToOut(in, out);

        pl("File was successfully uploaded.");
    } catch (Exception e) {
        throw new
        FileHandleException(e.getMessage());
    }
}

protected void inToOut(InputStream in, OutputStream
out) throws IOException {
    byte[] buffer = new byte[1024];
    int len = 0;
    try
    {
        while ((len = in.read(buffer)) > 0)
        {
            out.write(buffer, 0, len);
        }
    } finally
    {
        out.close();
        in.close();
    }
}
}

```

Лістинг коду класа SimpleFileServer.class

```

import java.net.ServerSocket;
import java.net.Socket;
import java.io.*;

public class SimpleFileServer {

    public static void main(String[] args) {
        ServerSocket ss;

```

```

Socket socket;
try{
    Integer port = Integer.parseInt(args[0]);
    File incoming = new File(args[1]);

    try{
        if(!incoming.exists() ||
!incoming.isDirectory())
            throw new Exception("wrong
uploading path.");

        pl("File server started...");
        while(true){
            ss = new ServerSocket(port);
            socket = ss.accept();

            pl("Client connected: " +
socket.getRemoteSocketAddress());

            receiveFile(socket.getInputStream(), incoming);

            socket.close();
            ss.close();
        }
    }catch(Exception e){
        pl("\nError: " + e.getMessage());
    }
    }catch(Exception e){
        pl("\nUse: java SimpleFileServer [port]
[uploading path]\n" +
            "Example: java SimpleFileServer
2909 C:\\MyServer\\incoming");
    }

}

public static void p(Object o){ if(o !=null)
System.out.print(o.toString()); }
public static void pl(Object o){ if(o !=null)
System.out.println(o.toString()); }

protected static void receiveFile(InputStream in,
File incoming) throws Exception {

    BufferedReader bin
        = new BufferedReader(new
InputStreamReader(in));

```



```
String filename = bin.readLine();
OutputStream out = new BufferedOutputStream(new
FileOutputStream( new File(incoming, filename )));

byte[] buffer = new byte[1024];
int len = 0;
try
{
    while ((len = in.read(buffer)) > 0)
    {
        out.write(buffer, 0, len);
    }
}
finally
{
    out.close();
    in.close();
}

pl("Received file: " + filename);
}
```



```

C:\WINDOWS\system32\cmd.exe
Can not copy file. Reason: d:\test\media.txt (The system cannot find the path specified).
Use "java FileHandler -h" for help.
E:\app\FileHandler>java FileHandler -c e:\media.txt d:\test
Can not copy file. Reason: d:\test\media.txt (The system cannot find the path specified).
Use "java FileHandler -h" for help.
E:\app\FileHandler>java FileHandler -c e:\media.txt e:\test
File was successfully copied.
E:\app\FileHandler>java FileHandler -s e:\media.txt melchanov@ua.fm
File was sent by e-mail.
Use "java FileHandler -h" for help.
E:\app\FileHandler>

```

Рис. 4.5 Результат програвання гілки графа у випадку вибору користувачем методу пересилання по електронній пошті з використанням протоколу SMTP

```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Версия 5.1.2600.1
(C) Корпорация Майкрософт, 1985-2001
C:\Documents and Settings\Администратор>cd /d E:\app\FileHandler\
E:\app\FileHandler>java FileHandler -u e:\media.txt 127.0.0.1
File was successfully uploaded.
E:\app\FileHandler>

```

Рис. 4.6 Результат програвання гілки графа у випадку вибору користувачем методу пересилання з допомогою протоколу HTTP

На сервері центрального сегмента був запущений HTTP файл-сервер, написаний мовою Java. Даний сервер працює у фоновому режимі і призначений для доставки відправленого з регіонального сегмента файлу і переміщення цього файлу в заздалегідь визначену в конфігураційному файлі директорію

У результаті програвання моделі центрального сегмента були отримані результати, представлені на рис. 4.7.


```

C:\WINDOWS\system32\cmd.exe - java SimpleFileServer 2909 c:\kazan

E:\app\FileHandler>java FileHandler -c e:\media.txt d:\test
Can not copy file. Reason: d:\test\media.txt <неСххх эх сфрхGg эршш сьрчрээш
aeCHD

Use "java FileHandler -h" for help.

E:\app\FileHandler>java FileHandler -c e:\media.txt e:\test
File was successfully copied.

E:\app\FileHandler>java FileHandler -c e:\media.txt e:\test
File was successfully copied.

E:\app\FileHandler>java FileHandler -c e:\media.txt molchanov@ua.fm
Can not send e-mail. check your mail configuration.

Use "java FileHandler -h" for help.

E:\app\FileHandler>cd /d E:\app\SimpleFileServer\

E:\app\SimpleFileServer>java SimpleFileServer 2909 c:\kazan
File server started.
Client connected: 127.0.0.1:1128
Received file: media.txt

```

Рис. 4.7 Результат роботи HTTP сервера центрального сегмента

Висновки по даному дослідженню й перспективи подальших досліджень

Досвід, отриманий у процесі використання JBPM показав, що, зокрема, завдяки використуванню Java і XML технологій, система є такою, що розвивається і перспективною. JBPM є машино-незалежною і об'єктно-орієнтованою, порівняно з розглянутими у оглядовій частині дипломної роботи, легкою у використанні професіональними програмістами. Однак можна зробити висновок, що на даному етапі свого розвитку система є складною для використання менеджерами, що не знайомі з технологією Java у досконалості, оскільки надає дружній інтерфейс тільки в убудованому графічному проектувальнику, що сам по собі дозволяє лише створювати графічне представлення послідовності бізнес процесу. Більш деталізоване моделювання інформаційних систем на рівні форм і функцій (об'єктів і методів) можливо тільки при участі Java-програмістів. Також слід зазначити відсутність у JBPM прийнятних засобів для моделювання бізнес-процесів розподілених інформаційних систем, що робить складним розробку корпоративних інформаційних систем рівня підприємства, призначених для роботи в розподіленому інформаційно-телекомунікаційному середовищі.

В розширеній версії моделі регіонального сегменту рекомендується використовувати паралельний граф з функцією вибору наступної дії користувачем, написаний на скриптовій мові BeanShell. Оскільки робота виконана в операційній системі Windows 2003 Server, рекомендується провести фінальний тест моделі на Unix-подібних операційних системах, а також на MacOS.

ЛІТЕРАТУРА

1. Что такое BPMS. - <http://www.bpms.ru/intro/>
2. Спецификация языка Business Process Execution Language - <http://bpel.com>
3. Спецификация языка Business Process Modeling Language - <http://bpmi.org>
4. Ю.Н.Молчанов, Авторский перевод документации Jboss jBPM - <http://citforum.ru>

Вісник Харківського національного університету імені В. Н. Каразіна
Серія «Математичне моделювання. Інформаційні технології.
Автоматизовані системи управління»

УДК 004.652:519.872

№ 733, 2006, с.17-24

Моделирование многофазных обслуживающих комплексов с постоянным циклом

Али Найф Халил Альхжуж, С. Ю. Игнатов

Харьковский национальный университет им. В.Н. Каразина, Украина

The problem of the mathematical modeling of the multiphase service complexes with the constant cycle is considered in the present paper. The mathematical model and the algorithm of the structure synthesis of the multiphase service complexes with the constant cycle have been built as a result. The period (length of cycle) of the requests service for such complexes was developed too.

1. Общая постановка задачи и её актуальность

В настоящей работе предлагается общая математическая модель многофазного обслуживающего комплекса с постоянным циклом. Показано, как в терминах этой модели могут быть сформулированы задачи проектирования цифровых управляющих систем и производственных систем дискретного типа с постоянным циклом обслуживания. Приведено определение сбалансированной структуры многофазного обслуживающего комплекса с постоянным циклом. Построен алгоритм синтеза структуры и приведен расчёт длительности цикла управления.

Комплексом называться набор приборов, способных обслуживать некоторые требования. Фаза обслуживания одного требования одним конкретным прибором (от начала до конца) будет называться операцией. Набор операций по обслуживанию конкретного требования будет называться маршрутом.

Целью обслуживания требований является формирование на их основе некоторого набора конечных (сборочных) пакетов требований.

Для систем цифрового управления реальными объектами в качестве требований могут выступать кадры телеметрии, а конечными сборочными пакетами могут быть кадры управляющих воздействий. Для дискретных производственных систем требованиями являются заготовки, детали, узлы, а конечными сборочными пакетами – изделия.

Номенклатура, количества и сроки формирования конечных сборочных пакетов определяются портфелем заявок обслуживающего комплекса.

Для синтеза обслуживающих комплексов необходимо использовать следующие соображения:

- определить тип обслуживающей системы (на одних и тех же приборах или оборудовании и с одним и тем же персоналом невозможно обслуживать управление спутниками связи и доменными печами, как и невозможно производить, например, мотоциклы и электробритвы – хоть одновременно, хоть поочередно);
- проанализировать количества и сложность объектов управления или провести маркетинговые исследования сбыта выбранной продукции и

определить номенклатуру, объёмы и периодичность пакетов кадров телеметрии и кадров управляющих воздействий или выпуска изделий;

- определить количество управляющих серверов или комплексов конечной сборки изделий (один - с переналадкой на обслуживание различных модификаций управляемых объектов или выпускаемых изделий, или несколько - параллельно обслуживающих различные наименования объектов или выпускающих различные изделия);

- произвести спецификацию кадров телеметрии по группам функционального подобию или разузлование изделий, группировку узлов по принципу технологического подобию сборки (здесь же определяются задания для служб снабжения на покупные узлы), и определить состав промежуточных сборочных участков;

- произвести спецификацию кадров в группах по алгоритмическому подобию или раздетализацию узлов, группировку деталей по принципу подобию технологических маршрутов (здесь же определяются задания для служб снабжения на покупные детали), и определить составы ЛВС спецвычислителей или производственных участков по принципу технологического подобию (токарные, корпусные, металлического или пластмассового литья и т.п.).

В процессе этих работ зачастую возникают ситуации, когда одно и то же требование участвует в формировании всех или большей части конечных пакетов и, следовательно, повторяемость такого требования достаточно высока. Для обслуживания именно этого требования обычно создают специальный комплекс, который называется обслуживающим комплексом с постоянным циклом. Это может быть специальный мультипроцессорный комплекс для решения системы дифференциальных уравнений определённого вида или линия по производству автомобильных шаровых опор и т.п. Задача определения структуры и задача управления обслуживанием требований для таких комплексов являются актуальными в данных областях применения.

2. Классификация обслуживающих комплексов с постоянным циклом

Различают два основных типа обслуживающих комплексов с постоянным циклом [3, 4, 5]:

- непрерывно-поточные;
- прерывно-поточные.

К непрерывно-поточным относятся комплексы, у которых транспортировка требований совмещена с обслуживающими операциями, а длительности операций равны или кратны друг другу. Типичным представителем многофазных обслуживающих комплексов этого типа являются роторные линии [1, 2]. Общая схема роторной линии представлена на рис.1.

Роторная линия разрабатывается специально, как единое целое, и её такт обслуживания равен времени одного угла поворота самого «медленного» из роторов.

Непрерывно-поточные обслуживающие комплексы обладают высокой производительностью и надёжностью обслуживания требований, но требуют огромных затрат на проектирование и создание уникальных роторов и дополнительных устройств. Кроме того, никакие изменения в маршруте

требования, особенно во временных характеристиках операций и порядке их следования, практически невозможны. Поэтому непрерывно-поточные обслуживающие комплексы создаются для обслуживания требований от сотен тысяч до десятков миллионов штук в год и с гарантией отсутствия существенных изменений в маршрутах в течение 5 – 10 лет.

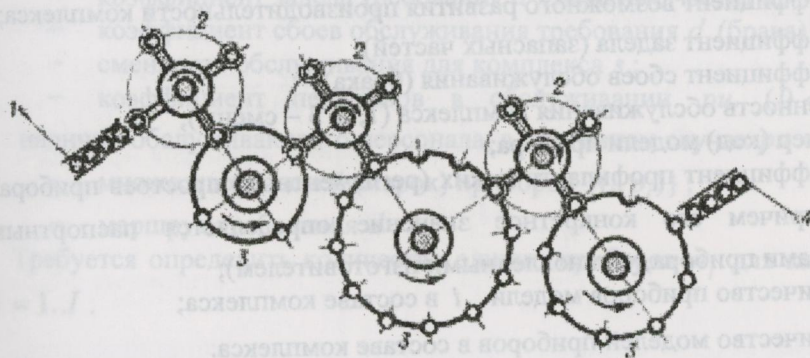


Рис. 1. Схема роторной линии: 1 — питающие устройства;

2 — транспортный ротор; 3 — рабочий ротор; 4 — приёмные устройства.

Прерывно-поточные обслуживающие комплексы разделяют транспортировку и обслуживание требований и не предъявляют ограничений на соотношение длительностей операций.

Из соображений возможности изменений в маршруте обслуживаемого требования, продиктованных объективной необходимостью, следует отдать предпочтение прерывно-поточным комплексам, создаваемым в основном на базе серийно производимых обслуживающих приборов. Общая схема прерывно-поточного комплекса представлена на рис.2.

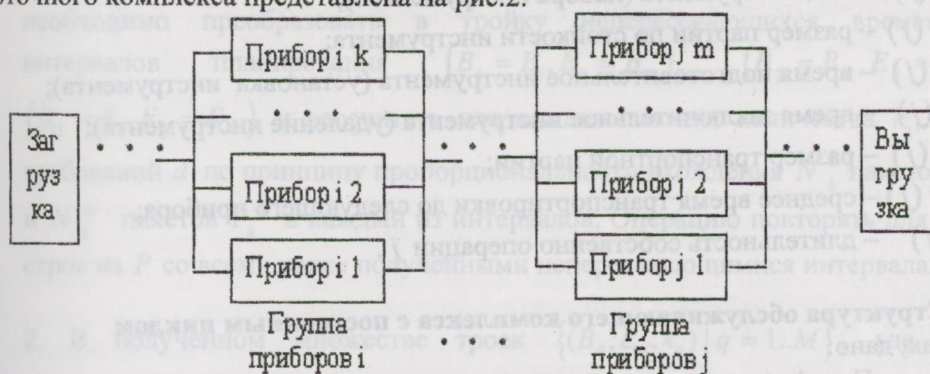


Рис. 2. Схема прерывно-поточного комплекса на базе серийных обслуживающих приборов.

3. Построение математической модели обслуживающего комплекса с постоянным циклом

Комплекс с постоянным циклом, обслуживающий требование d есть множество

$$W = (d, p_i, m_p, k_z, k_b, s, \{(i, p_i, n_i) \mid i = 1, \dots, I\}),$$

где:

p_i – коэффициент перерывов в обслуживании ($0 < p_i < 1$ при наличии обслуживающего персонала, в противном случае $p_i = 0$);

m_p – коэффициент возможного развития производительности комплекса;

k_z – коэффициент задела (запасных частей);

k_b – коэффициент сбоев обслуживания (брака);

s – сменность обслуживания комплекса (1, 2, 3 – смены);

i – номер (код) модели прибора;

p_i – коэффициент профилактических (регламентных) простоев прибора i ($0 < p_i < 1$, причем его конкретное значение определяется паспортными характеристиками прибора, установленными изготовителем);

n_i – количество приборов модели i в составе комплекса;

I – количество моделей приборов в составе комплекса.

Представим маршрут обслуживания требования d на комплексе с постоянным циклом в виде множества:

$$d = \{(j, i(j), ki(j), ps(j), pd(j), zd(j), pi(j), zi(j), t(j)) \mid j = 1..J\},$$

Где:

j – номер операции;

$i(j)$ – модель обслуживающего прибора ($i(j) = 1..I$);

$pd(j)$ – время подготовительное требования (загрузка требования);

$zd(j)$ – время заключительное требования (выгрузка требования);

$ki(j)$ – код инструмента (набора инструментов);

$ps(j)$ – размер партии по стойкости инструмента;

$pi(j)$ – время подготовительное инструмента (установка инструмента);

$zi(j)$ – время заключительное инструмента (удаление инструмента);

$pt(j)$ – размер транспортной партии;

$mt(j)$ – среднее время транспортировки до следующего прибора;

$t(j)$ – длительность собственно операции j .

4. Структура обслуживающего комплекса с постоянным циклом

Итак, дано:

– Портфель заявок в виде множества

$$P = \{(Z_l, B_l, E_l, \{(P_k^l, N_k^l, k_k^l) \mid k = 1..K\}) \mid l = 1..L\},$$

где: Z_l – номер заявки; B_l, E_l – календарные сроки начала и конца обслуживания заявки Z_l , P_k^l – код конечного (сборочного) пакета, N_k^l – требуемое количество экземпляров пакета P_k^l , k_k^l – количество требований d_l ,

необходимых для формирования одного пакета P_k^l , K - количество различных пакетов в заявке Z_l , L - количество заявок в P ;

- коэффициент возможного развития производительности комплекса tr ;
- коэффициент задела требования d (запасные части) kz ;
- коэффициент сбоев обслуживания требования d (брака) kb ;
- сменность обслуживания для комплекса s ;
- коэффициент перерывов в обслуживании pu ($0 < pu < 1$ при наличии обслуживающего персонала, в противном случае $pu = 0$);
- множество моделей (марок) приборов $\{(i, p_i)\}$;
- маршрут требования d .

Требуется определить количество единиц приборов n_i для каждой группы $\forall i = 1..I$.

5. Алгоритм синтеза структуры обслуживающего комплекса с постоянным циклом

1. Преобразовать портфель заявок

$$P = \left\{ (Z_l, B_l, E_l, \{ (P_k^l, N_k^l, k_k^l) \mid k = 1..K \}) \mid l = 1..L \right\}.$$

Для этого любую пару пересекающихся по времени заявок из P (рис. 3)

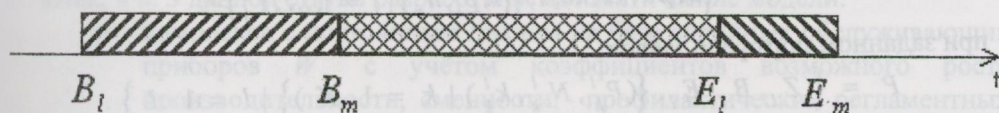


Рис. 3. Преобразование портфеля заявок.

необходимо преобразовать в тройку непересекающихся временных интервалов планирования $(B_1 = B_l, E_1 = B_m)$, $(B_2 = B_m, E_2 = E_l)$, $(B_3 = E_l, E_3 = E_m)$ и рассчитать для каждого из них количества k_1, k_2, k_3 требований d по принципу пропорциональности вхождения N_k^l пакетов P_k^l и N_k^m пакетов P_k^m в каждый из интервалов. Операцию повторять для всех строк из P со всеми ранее полученными непересекающимися интервалами.

2. В полученном множестве троек $\{(B_q, E_q, k_q) \mid q = 1..M\}$, где M - количество непересекающихся временных интервалов планирования, определить интервал r такой, что

$$\frac{k_r}{E_r - B_r} = \max_{q=1..M} \frac{k_q}{E_q - B_q};$$

3. Вычислить $k = kr \cdot (1 + mp + kz + kb)$ – количество требований d с учетом коэффициентов возможного развития производительности, заделов (запасных частей) и возможных сбоев обслуживания (брак);
4. $j = 1$;
5. Для операции j над требованием d вычислить фонд эффективного времени работы одной единицы прибора группы $i = i(j)$ по формуле:

$$T_i = k \cdot \left(t(j) \cdot (1 + p_i) + pd(j) + zd(j) + \frac{pi(j) + zi(j)}{ps(j)} + \frac{mt(j)}{pt(j)} \right)$$
6. Вычислить приближённое количество единиц приборов группы i по формуле:

$$n = \frac{3 \cdot T_i \cdot (1 + pu)}{s \cdot (E_r - B_r)};$$
7. $n_i = \lfloor n \rfloor + 1$;
8. $j = j + 1$;
9. Если $j \leq I$, то перейти к п. 5;
10. Конец алгоритма.

Таким образом, алгоритм рассчитывает структуру обслуживающего комплекса с постоянным циклом

$$W = \left(d, pu, kz, kb, s, \left\{ (i, p_i, n_i) \mid i = 1, \dots, I \right\} \right)$$

при заданном портфеле заявок

$$P = \left\{ (Z_l, B_l, E_l, \{ (P_k^l, N_k^l, k_k^l) \mid k = 1..K \}) \mid l = 1..L \right\}.$$

Небольшое замечание:

Рассмотрим разницу между дробной величиной n (см. п.6 алгоритма) и целой n_i (см. п.7 алгоритма), $\forall i = 1..I$, которая появляется из-за того, что например 0,3 единицы обслуживающего прибора, к сожалению, не бывает. Если $\exists i$ такие, что $n - n_i > 0.5$, то можно попытаться либо выбрать более производительную модель прибора, либо уменьшить коэффициент развития обслуживания пакетов, либо перенести профилактики приборов в нерабочую смену ($p_i = 0$) и т.п. После чего повторить алгоритм.

6. Такт обслуживающего комплекса с постоянным циклом

Тактом обслуживающего комплекса с постоянным циклом называется промежуток времени между запуском требований на первую операцию комплекса или их выпуском с последней операции.

$$\text{Величина } \tau_r = \frac{s \cdot (E_r - B_r)}{3 \cdot (1 + pu) \cdot k_r \cdot (1 + mp + kz + kb)} \quad \text{для } r\text{-того из}$$

непересекающихся временных интервалов планирования (B_r, E_r) , является минимальным тактом комплекса W .

Для $\forall \tau_q, q \neq r, q = 1..M$, таких, что

$$\tau_q = \frac{s \cdot (E_q - B_q)}{3 \cdot (1 + pu) \cdot k_q \cdot (1 + mp + kz + kb)},$$

τ_q является тактом комплекса W для интервалов планирования (B_q, E_q) , причём $\tau_q \geq \tau_r$.

Очевидно, что во всех интервалах (B_q, E_q) , $q \neq r, q = 1..M$, комплекс будет работать с пониженной производительностью, т.е. приборы комплекса будут недогружены требованиями d (простаивать).

Произойдёт это в том случае, когда портфель заявок P неуравновешен.

В п.2 алгоритма кроме $\frac{k_r}{E_r - B_r} = \max_{q=1..M} \frac{k_q}{E_q - B_q}$

можно вычислить ещё и $\frac{k_p}{E_p - B_p} = \min_{q=1..M} \frac{k_q}{E_q - B_q}$.

Тогда, если разница между $\max$ и $\min$ больше 5-10%, то такой портфель заявок называется неуравновешенным. В этом случае можно принять меры для уравнивания портфеля заявок P и повторить алгоритм синтеза структуры обслуживающего комплекса с постоянным циклом заново.

7. Вывод

Итак, в п. 3 данной статьи разработаны математические модели:

- комплекса с постоянным циклом на базе серийных обслуживающих приборов W с учётом коэффициентов возможного роста производительности, сменности, профилактических регламентных работ, возможных сбоев обслуживания, запасных частей;
- маршрута требования d с учётом подготовительно-заключительного времени, времени смены инструмента и размеров партий по его стойкости.
- портфеля заявок P , представляющего собой готовый или предлагаемый план выпуска конечных сборочных пакетов и потребность в обслуживании требования d для формирования каждого из них.

На основании этих моделей разработаны алгоритм синтеза структуры комплекса W п. 5 и расчёты тактов обслуживания таким комплексом требований d в любом из непересекающихся периодов планирования п. 6.

Конечно, для цифровых управляющих систем некоторые временные величины и коэффициенты приведенных математических моделей не имеют смысла, в отличие от управления дискретными производствами, но приравнивание к нулю этих величин никак не отражается на достоверности полученных результатов.

В работах [3-7] приводятся расчёты комплексов с постоянным циклом, но ни один из них не связывает их структуру и такт с портфелем заявок конечных (сборочных) пакетов и учитывают меньше особенностей маршрутов и конечных целей обслуживания требований.

ЛИТЕРАТУРА

1. Кошкин Л. Н., Густов А. А., Роторные машины для механической обработки, К., 1964;
2. Кошкин Л. Н., Комплексная автоматизация на базе роторных линий, М., 1965.
3. Организация поточного производства [Электронный ресурс] / et al. – 2002. Режим доступа: <http://www.nntu.sci-nnov.ru/RUS/fakyl/VECH/metod/orgprod1/part3.htm> – Последнее обращение: 12.07.2006. – Загл. с экрана.
4. ПОТОЧНЫЕ ЛИНИИ [Электронный ресурс] / et al. – 2004. Режим доступа: http://www.conveyor.ru/ru/articles/?cmd=full&nid=1598&tid=636&return_url=%3ftid%3d636%26st%3d0%26st1%3d%26tid1%3d – Последнее обращение: 12.07.2006. – Загл. с экрана.
5. Ребрин Ю.И., Основы экономики и управления производством, Гл.7, Таганрог: Изд-во ТРТУ, 2000.
6. Ruiz R., Maroto C., Alcaraz J. Two New Robust Genetic Algorithms for the Flowshop Scheduling Problem // OMEGA, 7The International Journal of Management Science. - № 34, 2006. - Pp. 461-476.
7. Pyke D. F., Powell S. G. Allocation of Buffers to Serial Production Lines with Bottlenecks. - IIE Transactions. - Vol. 28, 1996. - Pp. 18-29.

Разработка алгоритмов перехода между Е-сетями и вероятностно-временными графами

Амер Тахсин Саламех Абу-Джассар, Е. В. Дуравкин, Т. В. Семенова

Харьковский национальный университет им. В.Н. Каразина, Украина

In article the problem of complex using of analytical modeling and simulation technique is considered. E-net and probabilistic time graph are selected as basic tools of modeling. Translation algorithms between simulation model (E-net) and analytic model (probabilistic time graph) are developed.

1. Анализ проблемы.

Интенсивный рост объемов информации, циркулирующей в информационно-вычислительных системах, выдвигает все более жесткие требования к архитектуре и параметрам таких систем. С другой стороны, разработчикам отводится не так много времени для разработки и внедрения новых технологий передачи данных и управления информационными потоками, предназначенных для удовлетворения этих требований. Следовательно, в их арсенале должны быть средства, позволяющие за достаточно короткий срок получить информацию об исследуемой или разрабатываемой системе с заданной полнотой и достоверностью.

Такое положение вещей в данной области ставит свои задачи перед каждой из сторон, участвующих в разработке средств передачи данных, предоставлении услуг информационного обмена, потреблении таких услуг. Эти задачи заключаются в необходимости нахождения в максимально короткие сроки новых, наиболее экономичных решений, способных удовлетворить растущие запросы; уточнение влияния того или иного предлагаемого новшества на показатели качества уже существующих систем при их модификации; обоснованного выбора топологии, структуры, программного и информационного обеспечения средств информационного обмена для максимально качественного удовлетворения запросов потребителей.

Решение указанных задач невозможно без объективной предварительной оценки принятых решений, сравнения возможных вариантов. Исследование работы любой системы можно произвести либо испытаниями опытного образца, либо испытанием ее модели.

Специфика распределенных информационных систем указывает на значительные трудности, связанные с реализацией первого подхода. Следовательно, единственным разумным методом исследования разрабатываемых и модернизируемых систем данного класса является моделирование.

При моделировании перед разработчиками возникают задачи адекватного, формализованного описания анализируемого объекта, которое обеспечивало бы, с одной стороны, возможность изменения его структурных и функциональных

параметров для перехода от одного варианта к другому, а с другой стороны, обеспечивало возможность оценки каждого варианта по заданному критерию.

2. Постановка задачи и анализ публикаций.

В настоящее время наиболее распространенным методом анализа распределенных информационных систем является имитационное моделирование [1,2]. Для разработки имитационных моделей используется достаточно много математических аппаратов [2,3,4], однако, средством позволяющем наиболее полно отразить специфику функционирования систем в данной предметной области являются Е-сети [5,6].

В то же время Е-сети, впрочем, как и остальные средства имитационного моделирования, обладают рядом недостатков. Одним из основных является отсутствие возможности получения аналитических зависимостей между параметрами системы и анализируемыми показателями качества. Для получения таких зависимостей (наиболее часто вероятностно-временных) необходимо использование аналитических моделей, например построенных с использованием вероятностно-временных графов (ВВГ) [4]. Недостатком данного подхода является необходимость разработки аналитической модели с самого начала, что значительно увеличивает время исследования системы и затраты на разработку моделей.

Для устранения данной проблемы необходимо разработать метод позволяющий: на основе имитационной модели, представленной в виде Е-сети, построить аналитическую, с использованием ВВГ, а так же по аналитической модели построить имитационную.

Обратное преобразование необходимо в тех случаях, когда построенный вероятностно-временной граф не поддается упрощению и, следовательно, не позволяет получить требуемых зависимостей для анализируемых показателей качества. В этом случае использование имитационной модели позволит решить данную проблему.

3. Основная часть.

Формально Е-сеть задается следующим образом:

$E = (P, T, L, D, A, M_0)$, где

$P = \{p_1, p_2, \dots, p_n\}$ - конечное множество позиций Е-сети,

$T = \{t_1, t_2, \dots, t_m\}$ - конечное множество переходов Е-сети,

L - входная функция переходов, которая определяется как отображение
 $L: P \times T \rightarrow \{0, 1\}$,

D - выходная функция переходов, которая определяется как отображение
 $D: T \times P \rightarrow \{0, 1\}$,

A - временная функция переходов, которая определяется как отображение
 $A: T \rightarrow \square$, где $\square$ задает время перехода,

M_0 - вектор начальной разметки,

$Type: T \rightarrow \{t, F, F', J, J', MX, MY\}$ - функция, определяющая тип перехода.

$\mu : P \rightarrow \{0,1\}$ - функция, определяющая наличие метки в заданной позиции.

С каждым переходом связана процедура перехода, определяющая как, куда и за какое время будет передана метка.

Вероятностная процедура для МХ перехода представляет собой набор утверждений типа:

$P(\mu(p_1)=1) = \rho / \mu(p_2)=1$ - вероятность перехода метки в позицию p_1 при условии ее наличия в позиции p_2 равна ρ .

$\text{Pr} : T \times P \times P \rightarrow B$, где $B = \{x \in \mathbb{R} : 0 \leq x \leq 1\}$ - функция, задающая вероятность перехода метки из одной позиции в другую в вероятностном МХ и МУ переходе.

ВВГ формально задается следующим образом:

$G = (V, P, T, M)$, где

$V = \{v_1, v_2, \dots, v_n\}$ - множество вершин,

P - функция, задающая вероятностную характеристику дуги, которая определяется как отображение $P : V \times V \rightarrow B$, где $B = \{x \in \mathbb{R} : 0 \leq x \leq 1\}$ - множество вероятностей (обозначим $P((v_i, v_j)) = P_{ij}$),

T - функция, задающая временную характеристику дуги, которая определяется как отображение $T : V \times V \rightarrow \mathbb{R}$ (обозначим $T((v_i, v_j)) = T_{ij}$),

F - множество функций вида $f(z) = p \cdot z'$

M - функция, задающая вероятностно-временную характеристику дуги в виде функции из F , которая определяется как отображение $M : V \times V \rightarrow F$,

т.е. паре (v_i, v_j) ставится в соответствие функция $P_{ij} \cdot z^{T_{ij}}$.

Преобразование ВВГ к Е-сети.

При разработке моделей с использованием ВВГ строится ориентированный граф, вершинами которого являются возможные состояния системы. При разработке моделей с использованием Е-сетей строится двудольный ориентированный граф, в котором один тип вершин соответствует состояниям анализируемой системы, а второй тип вершин - условиям перехода системы между состояниями.

Таким образом, набор состояний модели, представленной в виде ВВГ, может быть однозначно отображен в набор состояний модели, представленной в виде Е-сети. Далее необходимо разработать механизмы, позволяющие однозначно преобразовать функции дуг ВВГ в функции переходов соответствующих типов Е-сети.

В ВВГ возможны два основных варианта топологии (рис. 1), остальные варианты топологий являются частными случаями рассмотренных.

При преобразовании ВВГ к Е-сети в первом случае (рис. 1а) переход из одного состояния в другое является однозначным (вероятность перехода равна 1), а для его осуществления необходимо время, задаваемое функцией дуги. Такая дуга преобразуется в t-переход со временем срабатывания равным параметру функции дуги.

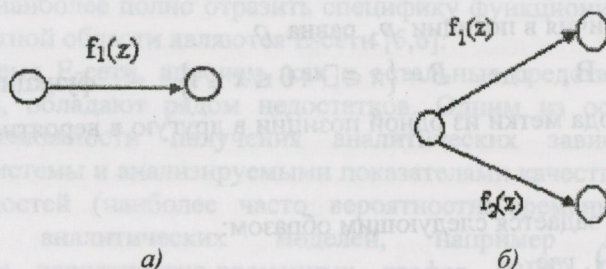


Рис. 1. Возможные топологии ВВГ

Во втором случае (рис. 1б) предполагается вероятностный процесс изменения состояния системы. При этом сумма вероятностей в функциях исходящих дуг должна быть равна 1, а временные характеристики дуг должны совпадать. Для описания данной ситуации средствами Е-сетей предлагается использовать МХ-переход. В таком переходе предикат решающего места будет определяться набором вероятностей перехода из вершины ВВГ в совокупность исходящих вершин, соответствующих местам Е-сети, являющихся исходящими для МХ-перехода.

Тогда алгоритм преобразования ВВГ в Е-сеть выглядит следующим образом.

На первом шаге по множеству вершин $V = \{v_1, v_2, \dots, v_n\}$ строится множество позиций $P = \{p_1, p_2, \dots, p_n\}$ Е-сети.

Далее для каждой вершины v_i из множества V выполняется следующий набор действий.

Если вершина имеет одну исходящую дугу, то создается t-переход.

Если вершина имеет несколько исходящих дуг, то создается МХ-переход.

Соответствующее условие проверяется с помощью функции $Out(v_i)$, возвращающей множество вершин, в которые переходят дуги, исходящие из v_i .

Таким образом, если мощность множества $Out(v_i) = 1$, то в множество переходов T Е-сети добавляется новый переход t_k и устанавливается тип нового перехода: $Type(t_k) = t$. Время срабатывания соответствующего перехода равно временной характеристике дуги, соединяющей вершину v_i и $\tilde{v}$, где $\tilde{v}$ — единственный элемент множества $Out(v_i)$. Определение перехода завершается указанием тех позиций Е-сети, с которыми он ассоциирован.

Поскольку вершине графа v_i соответствует позиция p_i Е-сети, а вершине $\tilde{v}$ - позиция $\tilde{p}$, то задание соответствующей топологии имеет вид:

$$L(p_i, t_k) = 1$$

$$D(t_k, \tilde{p}) = 1$$

Если мощность множества $Out(v_i)$ не равна 1, то во множество переходов T Е-сети добавляется новый переход t_k и устанавливается тип нового перехода: $Type(t_k) = MX$. Согласно требованию, временные характеристики всех дуг, исходящих из v_i , совпадают. Поэтому, время срабатывания соответствующего перехода равно временной характеристике дуги, соединяющей вершину v_i и произвольную вершину из множества $Out(v_i)$, например, $\tilde{v}$. Затем задаются значения входной и выходной функции перехода t_k . Входная функция задается как и ранее: $L(p_i, t_k) = 1$, а для задания значений выходной функции рассматриваются все вершины из множества $Out(v_i)$ и соответствующие им позиции. При этом сразу указываются вероятности перехода из позиции p_i в позиции, соответствующие вершинам из $Out(v_i)$. Задание соответствующих вероятностей и задает вероятностную процедуру созданного MX перехода.

При создании переходов различного типа ведется их сплошная нумерация с помощью счетчика k .

Ниже соответствующий алгоритм построения Е-сети по ВВГ представлен в псевдокоде. Считается, что формирование множества позиций $P = \{p_1, p_2, \dots, p_n\}$ по вершинам $V = \{v_1, v_2, \dots, v_n\}$ выполнено предварительно.

Используемые функции:

$Out(v)$ - функция, возвращающая множество вершин, в которые переходят дуги, исходящие из v .

Вход:

- вероятностно-временной граф $G = (V, P, T, M)$

- множество позиций $P = \{p_1, p_2, \dots, p_n\}$ Е-сети

Выход:

- Е-сеть $E = (P, T, L, D, A, M_0)$

Временные переменные:

- k - счетчик количества переходов

Псевдокод алгоритма:

begin


```

k = 0
for each  $v_i$  from  $V$  do begin
    if ( $|Out(v_i)| = 1$ )
        begin
            k = k + 1
             $T = T \cup \{t_k\}$ 
             $Type(t_k) = t$ 
             $A(t_k) = T(v_i, \tilde{v})$ 
             $L(p_i, t_k) = 1$ 
             $D(t_k, \tilde{p}) = 1$ 
        end
    else
        begin
            k = k + 1
             $T = T \cup \{t_k\}$ 
             $Type(t_k) = MX$ 
             $A(t_k) = T(v_i, \tilde{v})$  //  $\tilde{v}$  - произвольный элемент  $Out(v_i)$ 
             $L(p_i, t_k) = 1$ 
        end
    for each  $v_j$  from  $Out(v_i)$  do begin
         $D(t_k, p_j) = 1$ 
         $Pr(t_k, p_i, p_j) = P_{ij}$ 
    end
end
end for
end

```

Вектор атрибутов метки для разработанной Е-сети определяется согласно задаче исследования.

Преобразование Е-сети к ВВГ.

Основной особенностью Е-сетей, как средства анализа сложных систем является возможность моделирования параллельных взаимодействующих процессов. Для реализации данной возможности в формальном описании Е-сети присутствует набор переходов различных типов, определяющий различные комбинации условий, и вектор атрибутов метки, содержащий информацию о истории функционирования системы.

Очевидно, что при переходе от Е-сети к ВВГ будет потеряна часть информации, так например ВВГ не позволяют моделировать параллельные

процессы. В тоже время, получение характеристик анализируемой системы при больших ограничениях, это общий недостаток средств аналитического моделирования.

Использование ВВГ в данной ситуации позволит получить аналитическую модель системы, выделив необходимую часть из имитационной модели, представленной в виде Е-сети.

Как уже упоминалось выше, вершины ВВГ полностью соответствуют позициям Е-сети. Для определения топологии ВВГ и функций дуг необходимо описать процесс преобразования переходов различных типов.

Исходя из описания Е-сети, множество переходов содержит семь различных типов переходов определяющих логику функционирования модели, следовательно, на их основе необходимо разрабатывать топологию ВВГ и функции дуг. Преобразование переходов необходимо производить следующим образом.

Переход t -типа, имеющий входную позицию p_i и выходную p_j преобразовывается в дугу, соединяющую вершины v_i и v_j . Функция перехода будет иметь вид $1z^{A(t_k)}$, где $A(t_k)$ – время срабатывания преобразуемого t -перехода.

Преобразование $F(F')$ -переходов в ВВГ невозможно. Данный факт обусловлен тем, что переходы этого типа выполняют разделение моделируемых потоков (информационных, управляющих и т.п.), а аппарат ВВГ не позволяет описывать такие системы. Следовательно, при необходимости преобразования Е-сети в ВВГ на первом этапе исследователь должен выявить наличие $F(F')$ -переходов в модели. В случае, если такие переходы присутствуют в модели их можно заменить t -переходом, но при этом часть Е-сети, описывающую протекание второго процесса необходимо удалить из модели.

J -переходы описывают выполнение условия, при одновременном наступлении нескольких событий. При переходе к аналитической модели данную ситуацию также невозможно описать средствами ВВГ, следовательно в данном случае необходим предварительный анализ Е-сети на наличие J -переходов. В большинстве случаев такие переходы будут исключены из Е-сети при преобразовании $F(F')$ -переходов. Такая же ситуация будет наблюдаться и в случае MY -переходов.

J' -переходы, имеющие набор входных позиций $p_1, p_2, \dots, p_m$ и выходную позицию p_{m+1} , преобразовываются в набор вершин ВВГ $v_1, v_2, \dots, v_m, v_{m+1}$ и их соединяющие дуги, с функциями вида $1z^{A(t_k)}$, где $A(J_k)$ – время срабатывания преобразуемого J -перехода.

MX -переходы, описывающие управление моделируемыми потоками возможно преобразовать в ВВГ только в случае вероятностного задания предиката перехода. В этом случае входная позиция p_1 и выходные $p_2, \dots, p_m$ преобразуются в соответствующие вершины ВВГ $v_1, v_2, \dots, v_m$. Функции дуг, соединяющих вершину v_1 с вершинами $v_2, \dots, v_m$ будет иметь вид $p_i z^{A(MX)}$, где p_i – вероятность перехода в i -ю выходную позицию.

Исходя из вышесказанного алгоритм преобразования Е-сети в ВВГ заключается в следующем.

На первом шаге по множеству позиций $P = \{p_1, p_2, \dots, p_n\}$ Е-сети строится множество вершин $V = \{v_1, v_2, \dots, v_n\}$.

Далее последовательно просматриваются все переходы. Для каждого перехода t_k из множества T выполняется следующий набор действий.

Если переход t_k является t-переходом, то с помощью функций $In(t_k)$ и $Out(t_k)$ определяются входная (p_i) и выходная (p_j) позиции перехода. Поскольку речь идет о t-переходе, то вероятностная характеристика дуги, соединяющая соответствующие вершины графа v_i и v_j , будет равна единице: $P(v_i, v_j) = 1$. Временная характеристика дуги вычисляется по временной функции перехода: $T(v_i, v_j) = A(t_k)$.

Если переход t_k является J или J'-переходом, то, как и в случае t-перехода, с помощью функций $In(t_k)$ и $Out(t_k)$ определяется множество входных позиций ($SetIn$) и выходная (p_j) позиция перехода. Далее задаются характеристики дуг между вершинами графа, соответствующими позициям из $SetIn$, и вершиной, соответствующей позиции p_j . Вероятностная характеристика дуг устанавливается равной единице. Временная характеристика каждой из дуг вычисляется по временной функции перехода: $T(v_i, v_j) = A(t_k)$.

Если переход t_k является MX - переходом, то с помощью функций $In(t_k)$ и $Out(t_k)$ определяется входная позиция (p_i) и множество выходных позиций ($SetProcOut$) перехода. Далее задаются характеристики для каждой из дуг, которая соединяет вершину v_i , соответствующую позиции p_i , и вершину v_j , соответствующую позиции из множества $SetProcOut$. Временная характеристика дуги задается на основе времени срабатывания перехода: $T(v_i, v_j) = A(t_k)$, а вероятностная характеристика вычисляется на основе вероятностной процедуры MX - перехода.

Ниже соответствующий алгоритм построения ВВГ по Е-сети представлен в псевдокоде. Считается, что формирование множества вершин $V = \{v_1, v_2, \dots, v_n\}$ по позициям $P = \{p_1, p_2, \dots, p_n\}$ Е-сети выполнено предварительно.

Вход:

- Е-сеть $E = (P, T, L, D, A, M_0)$

- вершин $V = \{v_1, v_2, \dots, v_n\}$ ВВГ

Выход:

- вероятностно-временной граф $G = (V, P, T, M)$

Используемые функции:

$In(t_k)$ - функция, возвращающая множество позиций p_i Е-сети таких, что

$$L(p_i, t_k) = 1.$$

$Out(t_k)$ - функция, возвращающая множество позиций p_i Е-сети таких, что

$$D(t_k, p_i) = 1.$$

$ProcOut(t_k, p)$ - функция, возвращающая множество позиций p_i Е-сети таких, что $Pr(t_k, p, p_i) \neq 0$.

Псевдокод алгоритма:

begin

for each t_k from T do begin

if ($Type(t_k) = t$)

begin

$$p_i = In(t_k)$$

$$p_j = Out(t_k)$$

$$P(v_i, v_j) = 1$$

$$T(v_i, v_j) = A(t_k)$$

$$M(v_i, v_j) = 1 \cdot z^{A(t_k)}$$

end

if ($Type(t_k) = J$ or $Type(t_k) = J'$)

begin

$$SetIn = In(t_k)$$

$$p_j = Out(t_k)$$

for each p_i from $SetIn$ do begin

$$P(v_i, v_j) = 1$$

$$T(v_i, v_j) = A(t_k)$$

$$M(v_i, v_j) = 1 \cdot z^{A(t_k)}$$

end

if ($Type(t_k) = MX$)

begin


```

 $p_i = \ln(t_k)$ 
Set ProcOut = ProcOut( $t_k, p_i$ )
for each  $p_j$  from Set ProcOut do begin
   $P(v_i, v_j) = \Pr(t_k, p_i, p_j)$ 
   $T(v_i, v_j) = A(t_k)$ 
   $M(v_i, v_j) = \Pr(t_k, p_i, p_j) \cdot z^{A(t_k)}$ 
end
end for
end

```

4. Выводы. В статье разработаны алгоритмы позволяющие выполнять преобразование аналитических моделей, представленных в виде ВВГ в имитационные, представленные в виде Е-сетей, а так же и обратное преобразование от имитационных к аналитическим. Полученный результат позволяет упростить процесс анализа сложных систем, при котором требуется разработка как аналитических, так и имитационных моделей. Упрощение процесса анализа в данном случае обусловлено тем, что при имеющейся модели (ВВГ или Е-сети) ее можно использовать в качестве основы для разработки необходимой (аналитической или имитационной). В тоже время нельзя не отметить ряд ограничений данного метода. В первую очередь ограничения проявляются при переходе от Е-сети к ВВГ, что связано с невозможностью аналитического описания нескольких моделируемых потоков и фиксированных процедур управления ними. Так же при переходе от Е-сети к ВВГ теряется семантика вектора атрибутов метки и процедур переходов, выполняющих преобразования вектора атрибутов. Однако, несмотря на указанные ограничения, разработанные алгоритмы позволяют повысить полноту анализа распределенных информационных систем.

ЛИТЕРАТУРА

1. Кельтон В., Лоу А. Имитационное моделирование. Классика CS. 3-е изд. – СПб.: Питер; Киев: Издательская группа ВНУ. 2004. – 647 с.: ил.
2. Мурата Т. Сети Петри: Свойства, анализ, приложения// ТИИЭР. №4. 1989.
3. Модели и методы исследований в системах информатики: Сб. ст./ Ин-т проблем пер. информации / Отв. ред. Харкевич, В.А. Гармаш. – М.: Наука, 1988. – 156 с.
4. Лосев Ю.И. и др. Автоматизированное управление в сетях с коммутацией пакетов. – К.: Техника, 1994. – 312 с.
5. Алгоритм параллельного выполнения и синхронизации Е-сети / Литвинов В.В., Казимир В.В., Гавсиевич И.Б. // Математические машины и системы. – 2005. – № 4. – С. 73 – 84.
6. Применение микропроцессорных средств в системах передачи информации: Учеб. пособ. для вузов по сп-ти АСУ/ Я. Советов, О.И. Кутузов и др. – М.:Высш. шк., 1987. –256 с.:ил.

Дискретизация и численная идентификация дифференциально-игровых моделей макроэкономической динамики

А. А. Васильев, А. М. Назаренко

Сумской государственный университет, Украина

The article deals with mathematical modeling of macroeconomic processes using some approaches of game theory and profit maximization hypothesis. Prognostic and descriptive characteristics of discrete models, obtained by various methods from differential-game dynamic model, are compared. Algorithms to identify these models with the help of econometric methods are substantiated. It is possible to use them analyzing macroeconomics of the developed countries.

В последнее время в теории математического моделирования экономических процессов и систем нашли широкое применение методы эконометрики [1,2], которые позволяют провести глубокий качественный анализ изучаемых систем, представленных в виде линейных моделей. Однако большинство объектов, изучаемых экономической наукой, может быть охарактеризовано понятием сложная система, и адекватно описывать их в рамках линейных моделей невозможно [3]. Вопрос построения и идентификации нелинейных динамических математических моделей экономических процессов и систем малоизучен как в отечественной, так и в зарубежной литературе. Некоторые подходы для решения данной проблемы могут быть заимствованы у активно развивающейся в последнее время дифференциальной теории игр [4-6], где представление изучаемого процесса сводится к ситуации конфликта или кооперации, записанной в виде системы дифференциальных уравнений, и участвующие стороны осуществляют выбор стратегии во времени.

В [7] предложен концептуальный подход к построению эконометрико-игровых моделей. В нем принципы дифференциальной теории игр используются для теоретического обоснования конструируемых моделей, а с помощью методов эконометрики проводится их идентификация.

В данной работе рассматривается обобщение этого подхода на случай рынка с множеством участников, а также изучение последствий этого обобщения и дальнейшей дискретизации на дескриптивные и прогнозные свойства моделей. При этом ставилась цель обосновать применимость предлагаемого подхода к описанию крупных рынков, точнее, к описанию их текущего состояния и вычислению для них прогнозных значений.

Следуя [7], при построении моделей будем использовать принцип максимизации прибыли в экономическом анализе [8], согласно которому при моделировании макроэкономической деятельности можно рассматривать рынок, все участники которого имеют одну цель — максимизировать собственную

прибыль. Этих участников можно рассматривать как игроков, которые, с точки зрения их влияния на интегральную функцию цели, преследуют общие или противоположные цели. Для этого нужно выбрать для изучения некоторый основной показатель h и функцию G , описывающую его зависимость от влияющих на него факторов – рыночных ресурсов, которыми управляют соответствующие игроки. В зависимости от рассматриваемого показателя h все ресурсы можно разделить на две группы: способствующие и препятствующие росту h , а соответствующие факторы будут иметь положительную (факторы $p_1, p_2, \dots, p_n$) или отрицательную (факторы $q_1, q_2, \dots, q_m$) направленность влияния на функцию G . Направленность влияния можно определить на основании экономической теории или посредством проведения корреляционного анализа системы показатель-факторы.

В качестве функции G удобно выбрать производственную функцию типа Кобба-Дугласа

$$G = a_0 p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdot \dots \cdot p_n^{\alpha_n} \cdot q_1^{\beta_1} \cdot q_2^{\beta_2} \cdot \dots \cdot q_m^{\beta_m}. \quad (1)$$

Так, при моделировании макроэкономической деятельности страны в качестве основного показателя h может быть взят валовый внутренний продукт (ВВП), а факторами, влияющими на него, будут [8]: затраты основных фондов p_1 , численность рабочей силы p_2 , материальные затраты q_1 . Тогда, выбрав для изучения двухфакторный рынок, вместо (1) можно записать

$$G = a_0 p_1^{\alpha_1} \cdot p_2^{\alpha_2}, \quad (2)$$

а рассматривая трехфакторный:

$$G = a_0 p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdot q_1^{\beta_1}. \quad (3)$$

При этом сумма оценок коэффициентов α_1 и α_2 в первом случае и α_1, α_2 и β_1 во втором позволит оценить эффект от увеличения масштаба производства.

В (2) показатели α_1 и α_2 являются коэффициентами эластичности выпуска (ВВП) по затратам основного капитала (p_1) и труда (p_2). Если по отдельности они указывают на процентное увеличение (или уменьшение) ВВП при однопроцентных колебаниях величин затрат капитала p_1 и труда p_2 , то их сумма $\alpha_1 + \alpha_2$ отражает уже общую реакцию производства на указанные изменения показателей. При $\alpha_1 + \alpha_2 = 1$ (ВВП увеличивается в той же пропорции, что и p_1 , и p_2) наблюдается неизменность отдачи факторов при любых масштабах. Если $\alpha_1 + \alpha_2 > 1$, ВВП растет в большей пропорции и наблюдается возрастающий эффект от масштаба производства. Если же $\alpha_1 + \alpha_2 < 1$, налицо убывающий эффект от масштаба производства (ВВП растет в меньшей пропорции, чем p_1 и p_2), наращивание затрат ресурсов оборачивается снижением их продуктивности [11].

При игровой интерпретации функцию G можно рассматривать как результат разрешения конфликта, который возникает на изучаемом рынке между двумя указанными выше группами игроков. Так, игроки, управляющие факторами $p_1, p_2, \dots, p_n$, стремясь в процессе своей деятельности увеличить собственную прибыль, будут наращивать значение соответствующих факторов и, как следствие, увеличивать показатель h . Однако игроки, от которых зависят значения величин $q_1, q_2, \dots, q_m$, также, стремясь получить наибольшую прибыль, своими действиями будут уменьшать значение показателя h . Тогда по поведению функции G , включающей в себя результаты взаимодействия всех

рыночных участников, можно судить об успешности деятельности каждого из игроков в моменты времени $t = 0, 1, \dots, N$. Если рассматривать описанный рынок изменяющийся непрерывно во времени, то вместо дискретных значений изучаемых факторов в каждый момент времени необходимо ввести в рассмотрение соответствующие функции $p_1(t), p_2(t), \dots, p_n(t), q_1(t), q_2(t), \dots, q_m(t)$. В этом случае ситуация на рассматриваемом рынке может быть описана следующей системой дифференциальных уравнений градиентного типа [9]:

$$\begin{cases} \frac{dp_1}{dt} = u_1(t) \frac{\partial G(p_1, p_2, \dots, p_n, q_1, q_2, \dots, q_m)}{\partial p_1}, \\ \dots \\ \frac{dp_n}{dt} = u_n(t) \frac{\partial G(p_1, p_2, \dots, p_n, q_1, q_2, \dots, q_m)}{\partial p_n}, \\ \dots \\ \frac{dq_1}{dt} = -v_1(t) \frac{\partial G(p_1, p_2, \dots, p_n, q_1, q_2, \dots, q_m)}{\partial q_1}, \\ \dots \\ \frac{dq_m}{dt} = -v_m(t) \frac{\partial G(p_1, p_2, \dots, p_n, q_1, q_2, \dots, q_m)}{\partial q_m} \end{cases} \quad (4)$$

при начальных условиях $p_i(0) = p_i^*(0), q_j(0) = q_j^*(0)$ ($i = 1, 2, \dots, n; j = 1, 2, \dots, m$).

Содержательный смысл функций $u_i(t)$ и $v_j(t)$ в (4) состоит в характеристике скорости, с которой соответствующий игрок инвестировал средства на развитие сопоставленного ему фактора. Рассматривая систему (4) с экономической точки зрения, можно заметить, что каждое из уравнений представляет скорость роста рассматриваемого фактора. Например, dp_1/dt есть произведение скорости вложений $u_1(t)$ в p_1 на эффективность их вложения $\partial G / \partial p_1$.

Задача нахождения функций управления $u_i(t)$ и $v_j(t)$ теоретически может решаться несколькими способами [10], но известные процедуры их численной реализации требуют настолько много вычислений, что проведение полной идентификации системы (4) даже небольшой размерности затруднительно. Кроме того, существующие подходы к решению обратных задач, к которым относится задача идентификации (4), разработаны для детерминированных моделей, а применение последних, в силу особенностей экономического моделирования, может привести к неадекватному описанию реальных систем. Поэтому здесь предложен другой подход к решению поставленной задачи, который базируется на эконометрическом аппарате и позволяет не только провести регрессионный анализ моделей, построенных на базе (4), но и оценить их качество приближения с помощью коэффициента детерминации R^2 [11].

В данной работе функции управления аппроксимируются многочленами

$$\begin{aligned} u_i(t) &= b_{i0} + b_{i1}t + b_{i2}t^2 + \dots + b_{ik_i}t^{k_i}, \quad i = 1, 2, \dots, n; \\ v_j(t) &= c_{j0} + c_{j1}t + c_{j2}t^2 + \dots + c_{jl_j}t^{l_j}, \quad j = 1, 2, \dots, m, \end{aligned} \quad (5)$$

причём степени многочленов устанавливаются апостериорно – экспериментально на основе критерия, который будет описан ниже.

Решение задачи идентификации модели сводится к нахождению оценок неизвестных коэффициентов в (5). Для этого в работе предлагается два подхода. Согласно первому необходимо записать (4) в разностной форме, заменив неизвестные функции $u_i(t)$ и $v_j(t)$ предложенным в (5) аналогом. Получим следующую модель:

$$\left\{ \begin{array}{l} p_1^{t+1} = p_1^t + (b_{10} + b_{11}t + b_{12}t^2 + \dots + b_{1k_1}t^{k_1}) \frac{\partial G}{\partial p_1} + u_{k_1}^t, \\ \dots \dots \dots \\ p_n^{t+1} = p_n^t + (b_{n0} + b_{n1}t + b_{n2}t^2 + \dots + b_{1k_n}t^{k_n}) \frac{\partial G}{\partial p_n} + u_{k_n}^t, \\ q_1^{t+1} = q_1^t - (c_{10} + c_{11}t + c_{12}t^2 + \dots + c_{1l_1}t^{l_1}) \frac{\partial G}{\partial q_1} + v_{l_1}^t, \\ \dots \dots \dots \\ q_m^{t+1} = q_m^t - (c_{m0} + c_{m1}t + c_{m2}t^2 + \dots + c_{ml_m}t^{l_m}) \frac{\partial G}{\partial q_m} + v_{l_m}^t, \end{array} \right. \quad (6)$$

где $t = 0, 1, \dots, N-1$; p_i^t ($i = 1, 2, \dots, n$), q_j^t ($j = 1, 2, \dots, m$) – известные на основании статистических данных значения параметров системы в момент времени t ; $u_{k_i}^t$ ($i = 1, 2, \dots, n$), $v_{l_j}^t$ ($j = 1, 2, \dots, m$) – некоррелирующие, нормально распределенные случайные отклонения модели с нулевым математическим ожиданием и соответствующими дисперсиями: $u_{k_i}^t \sim N(0, \sigma_{u_{k_i}}^2)$, $v_{l_j}^t \sim N(0, \sigma_{v_{l_j}}^2)$.

Второй подход основан на записи системы (4) в интегральной форме:

$$\left\{ \begin{array}{l} p_1(t) = p_1^* + \int_0^t u_1(t) \frac{\partial G(p_1, p_2, \dots, p_n, q_1, q_2, \dots, q_m)}{\partial p_1} dt, \\ \dots \dots \dots \\ p_n(t) = p_n^* + \int_0^t u_n(t) \frac{\partial G(p_1, p_2, \dots, p_n, q_1, q_2, \dots, q_m)}{\partial p_n} dt, \\ q_1(t) = q_1^* - \int_0^t v_1(t) \frac{\partial G(p_1, p_2, \dots, p_n, q_1, q_2, \dots, q_m)}{\partial q_1} dt, \\ \dots \dots \dots \\ q_m(t) = q_m^* - \int_0^t v_m(t) \frac{\partial G(p_1, p_2, \dots, p_n, q_1, q_2, \dots, q_m)}{\partial q_m} dt. \end{array} \right. \quad (7)$$

Привлекательность интегральной формы (7) связана с предположением о том, что она позволит сгладить случайные выбросы в исходных статистических данных, вследствие введения временного лага, и избежать процедуры их предварительной фильтрации, которая, как правило, вносит в них дополнительные неточности.

Следуя (7), в работе предлагается следующая приближенная модель:

$$\begin{cases} p_1^{t+1} = P_1 + b_{10} \sum_{i=0}^t \frac{\partial G}{\partial p_1} + b_{11} \sum_{i=0}^t i \frac{\partial G}{\partial p_1} + b_{12} \sum_{i=0}^t i^2 \frac{\partial G}{\partial p_1} + \dots + b_{1k_1} \sum_{i=0}^t i^{k_1} \frac{\partial G}{\partial p_1} + u_{k_1}^t, \\ \dots \\ p_n^{t+1} = P_n + b_{n0} \sum_{i=0}^t \frac{\partial G}{\partial p_n} + b_{n1} \sum_{i=0}^t i \frac{\partial G}{\partial p_n} + b_{n2} \sum_{i=0}^t i^2 \frac{\partial G}{\partial p_n} + \dots + b_{1k_n} \sum_{i=0}^t i^{k_n} \frac{\partial G}{\partial p_n} + u_{k_n}^t, \\ q_1^{t+1} = Q_1 - c_{10} \sum_{i=0}^t \frac{\partial G}{\partial q_1} - c_{11} \sum_{i=0}^t i \frac{\partial G}{\partial q_1} - c_{12} \sum_{i=0}^t i^2 \frac{\partial G}{\partial q_1} - \dots - c_{1l_1} \sum_{i=0}^t i^{l_1} \frac{\partial G}{\partial q_1} + v_{l_1}^t, \\ \dots \\ q_m^{t+1} = Q_m - c_{m0} \sum_{i=0}^t \frac{\partial G}{\partial q_m} - c_{m1} \sum_{i=0}^t i \frac{\partial G}{\partial q_m} - c_{m2} \sum_{i=0}^t i^2 \frac{\partial G}{\partial q_m} - \dots - c_{ml_m} \sum_{i=0}^t i^{l_m} \frac{\partial G}{\partial q_m} + v_{l_m}^t. \end{cases} \quad (8)$$

Оценки коэффициентов в (6) и (8) можно вычислить, рассматривая каждое уравнение соответствующей системы для моментов времени $t = 0, 1, \dots, N-1$ и используя метод наименьших квадратов. Для иллюстрации сказанного запишем первое уравнение системы (8) в матричном виде:

$$\begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & \frac{\partial G}{\partial p_1} \Big|_{t=0} & 0 & \dots & 0 \\ 1 & \sum_{i=0}^1 \frac{\partial G}{\partial p_1} \Big|_{t=i} & \sum_{i=0}^1 i \frac{\partial G}{\partial p_1} \Big|_{t=i} & \dots & \sum_{i=0}^1 i^{k_1} \frac{\partial G}{\partial p_1} \Big|_{t=i} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & \sum_{i=0}^{N-1} \frac{\partial G}{\partial p_1} \Big|_{t=i} & \sum_{i=0}^{N-1} i \frac{\partial G}{\partial p_1} \Big|_{t=i} & \dots & \sum_{i=0}^{N-1} i^{k_1} \frac{\partial G}{\partial p_1} \Big|_{t=i} \end{pmatrix} \begin{pmatrix} P_1 \\ b_{10} \\ b_{11} \\ b_{12} \\ \dots \\ b_{1k_1} \end{pmatrix} + \begin{pmatrix} u_{k_1}^0 \\ u_{k_1}^1 \\ u_{k_1}^2 \\ \dots \\ u_{k_1}^N \end{pmatrix} = \begin{pmatrix} P_1^0 \\ P_1^1 \\ P_1^2 \\ \dots \\ P_1^N \end{pmatrix} \quad (9)$$

или сокращенно

$$\bar{A} \bar{b}_1^* + \bar{u}_{k_1} = \bar{y}.$$

Тогда оценки неизвестных коэффициентов будут вычисляться по формуле [11]

$$\bar{b}_1^* = (\bar{A}' \bar{A})^{-1} \bar{A}' \bar{y}, \quad (10)$$

где $\bar{A}'$ — транспонированная матрица.

Заметим, что применение моделей (6), (8) и разработанной методики их идентификации позволяет не только оценить функции $u_i(t)$ и $v_j(t)$, что дает возможность изучить состояние системы в рассматриваемый период, но и, в силу независимости значений величин p_i и q_j в момент времени t от их значений в предыдущие моменты времени, дать краткосрочный прогноз динамики развития изучаемых величин.

Степени полиномов (5) представляется разумным выбирать из условия нахождения максимально точных прогнозных значений исследуемых величин. Для определения оптимальных степеней в (5) в работе предлагается использовать критерий, базирующийся на величине доверительного интервала для прогнозных значений изучаемых величин [11]:

$$y_{N+1} = \hat{y}_{N+1} \pm \delta t_{\alpha}, \quad \delta = \sqrt{\hat{\sigma}_u^2 (1 + \bar{a}_{N+1} (\bar{A}' \bar{A})^{-1} \bar{a}_{N+1})}. \quad (11)$$

Здесь δ – среднеквадратическая ошибка прогноза, t_{α} – двусторонний квантиль распределения Стьюдента с $N-l-1$ степенями свободы (l – степень соответствующего полинома), $\hat{\sigma}_u^2$ – оценка дисперсии остаточного члена, $\bar{a}_{N+1}$ – значение вектора объясняющих переменных в момент времени $N+1$, матрица $\bar{A}$ описана выше.

Соответственно, чем ниже значение среднеквадратической ошибки прогноза, тем уже доверительный интервал для прогнозных значений и, следовательно, выше прогнозные свойства модели. Тогда оптимальной степенью соответствующего полинома в (5) можно считать такую степень, при которой среднеквадратическая ошибка прогноза в (11) будет минимальной.

Обоснуем описанный выше подход фактом успешного рассмотрения реальных макроэкономических систем небольшой размерности для определённых рынков. Рассматриваемые рынки включают трех обобщенных участников: первый – это производители товаров и услуг, которые выпускают конечный продукт, а в качестве фактора, которым он управляет возьмем потребление основных фондов p . Вторым участником выберем домохозяйства и будем характеризовать его численностью рабочей силы q . В качестве третьего участника будем рассматривать условного рыночного агента, следящего за материальными затратами и характеризуемого уровнем потребления r . Соответственно каждый из указанных участников в процессе своей деятельности оказывал влияние на величину ВВП h , однако направленность влияния каждого фактора на h априорно определить невозможно, так как она будет зависеть от рассматриваемого рынка.

Конкретнее рассмотрим динамику развития Дании, Финляндии, Франции, Италии, Японии, Нидерландов и США в период 1990-2005 гг. [12], используя статистические данные о потреблении основных фондов p , количестве рабочих q , уровне материальных затрат r и ВВП h . Построение моделей осуществим на данных 1990-2003 гг., а на 2004 г. и 2005 г. будем вычислять прогнозные значения изучаемых величин. Так как в [12] приведены прогнозные экспертные оценки их значений на 2006 г., то интересным является сравнение последних с рассчитанными на основании моделей значениями. Иллюстрацию проведенных вычислений проведем на примере Дании (табл. 1), а результаты расчетов для остальных стран приведем в сводной таблице.

Аппроксимируем зависимость величины валового внутреннего продукта от потребления основного капитала, численности рабочих страны и материальных затрат функцией вида (3). После оценивания неизвестных коэффициентов линеаризованная форма (3) принимает вид (в скобках указаны стандартные ошибки оценок, u – случайное отклонение модели):

$$\ln G = -1.714 + 0.450 \ln p + 0.434 \ln q + 0.487 \ln r + u, \quad R^2 = 0.99858,$$

(с.о.) (1.538) (0.162) (0.183) (0.177)

откуда получаем

$$G(p, q, r) = 0.180 \cdot p^{0.450} \cdot q^{0.434} \cdot r^{0.487} \cdot e^u. \quad (12)$$

Таблица 1. Макроэкономические показатели Дании в период 1990-2005 гг.

t	Год	Затраты основного капитала p (mio eur ¹)	Численность рабочих q (тыс. чел.)	Материальные затраты r (mio eur ¹)	ВВП h (mio eur ¹)
0	1990	16176.4	2645	80745.5	107000.0
1	1991	17161.1	2629	83910.9	110558.7
2	1992	17925.4	2600	88099.5	116092.4
3	1993	19100.2	2562	92535.6	120076.2
4	1994	20152.8	2604	99888.1	129511.9
5	1995	21566.5	2629	106301.4	139129.2
6	1996	22700.7	2655	110526.4	145323.9
7	1997	23641.8	2687	113588.9	150414.1
8	1998	24597.2	2727	118464.5	155163.1
9	1999	26233.9	2753	122588.0	163199.9
10	2000	27425.9	2764	126349.1	173597.9
11	2001	28767.1	2785	130828.4	179226.1
12	2002	30011.3	2783	136258.4	184743.6
13	2003	30676.1	2742	140867.2	189640.5
14	2004	31909.0	2741	147704.1	197221.7
15	2005	33247.5	2761	155172.4	208546.1
16	2006	35098.9	2773	162104.0	221849.8

¹ mio_eur: Миллионов евро (с 01.01.1999)/Миллионов ECU (European Currency Unit) (до 31.12.1998).

Отметим, что все коэффициенты полученной модели значимы при уровне значимости $\alpha = 0.05$, и, следовательно, она может быть выбрана для моделирования макроэкономического рынка Дании. Использование в качестве функции выигрыша в рассматриваемой дифференциальной игре именно производственной функции типа Кобба-Дугласа позволяет избежать проведения корреляционного анализа системы показатель-факторы, так как направленность воздействия и значимость влияния конкретного фактора на изучаемый показатель будет определяться соответственно знаком и значимостью соответствующего оцененного коэффициента.

Анализ (12) показывает, что факторы p , q и r имеют положительную направленность влияния на величину h , и в обозначениях систем (6) и (8) им будут соответствовать p_1 , p_2 и p_3 . Учитывая это, аналогом (6) для изучаемого макроэкономического рынка Дании будет:

$$\begin{aligned}
 p^{t+1} &= p^t + (b_{10} + b_{11}t + b_{12}t^2 + \dots + b_{1k_1}t^{k_1}) \cdot 0.0810 p^{-0.550} q^{0.434} r^{0.487} + u_{k_1}^t, \\
 q^{t+1} &= q^t + (b_{20} + b_{21}t + b_{22}t^2 + \dots + b_{2k_2}t^{k_2}) \cdot 0.0782 p^{0.450} q^{-0.566} r^{0.487} + u_{k_2}^t, \\
 r^{t+1} &= r^t + (b_{30} + b_{31}t + b_{32}t^2 + \dots + b_{3k_3}t^{k_3}) \cdot 0.0878 p^{0.450} q^{0.434} r^{-0.513} + u_{k_3}^t.
 \end{aligned} \quad (13)$$

Матричная форма первого уравнения (13), расписанного для моментов времени $t = 0, 1, \dots, N-1$, принимает вид:

$$\begin{pmatrix} \frac{\partial G}{\partial p} \Big|_{t=0} & 0 & \dots & 0 \\ \frac{\partial G}{\partial p} \Big|_{t=1} & \frac{\partial G}{\partial p} \Big|_{t=1} & \dots & \frac{\partial G}{\partial p} \Big|_{t=1} \\ \frac{\partial G}{\partial p} \Big|_{t=2} & 2 \frac{\partial G}{\partial p} \Big|_{t=2} & \dots & 2^{k_1} \frac{\partial G}{\partial p} \Big|_{t=2} \\ \dots & \dots & \dots & \dots \\ \frac{\partial G}{\partial p} \Big|_{t=N-1} & (N-1) \frac{\partial G}{\partial p} \Big|_{t=N-1} & \dots & (N-1)^{k_1} \frac{\partial G}{\partial p} \Big|_{t=N-1} \end{pmatrix} \begin{pmatrix} b_{10} \\ b_{11} \\ b_{12} \\ \dots \\ b_{1k_1} \end{pmatrix} + \begin{pmatrix} u_{k_1}^1 \\ u_{k_1}^2 \\ u_{k_1}^3 \\ \dots \\ u_{k_1}^{N-1} \end{pmatrix} = \begin{pmatrix} p^1 - p^0 \\ p^2 - p^1 \\ p^3 - p^2 \\ \dots \\ p^N - p^{N-1} \end{pmatrix}.$$

Отсюда можно найти оценки $\hat{b}_{1i}$ неизвестных коэффициентов b_{1i} методом наименьших квадратов и оценить функцию управления $u_1(t)$, соответствующую фактору p . Аналогично могут быть оценены функции управления $u_2(t)$ для q и $u_3(t)$ для r . Расчеты дают:

$$\begin{aligned} u_1(t) &= 389.526, & u_2(t) &= 0.466 - 0.0180t, \\ & \text{(c.o.)} & & \text{(c.o.)} \quad (0.201) \quad (0.00820) \\ u_3(t) &= 4937.955 + 2504.817t - 503.835t^2 + 26.465t^3. \\ & \text{(c.o.)} & & (1471.323) \quad (1107.170) \quad (218.916) \quad (11.831) \end{aligned}$$

Здесь все коэффициенты в $u_1(t)$, $u_2(t)$ и $u_3(t)$ значимы, что дает основание предполагать адекватность построенных моделей. При найденных функциях управления величины p , q и r приближаются с высокой точностью: для них коэффициенты детерминации равны 0.99414, 0.91348 и 0.99841 соответственно.

Чтобы подтвердить адекватность модели (12), (13) для макроэкономического рынка Дании, необходимо проверить условия, накладываемые на остатки u , u_{k_1} ,

u_{k_2} и u_{k_3} , эконометрическим аналогом которых есть условия Гаусса-Маркова [13]. Расчеты показывают, что все присутствующие в модели случайные отклонения можно считать нормально распределенными случайными величинами. Из условий Гаусса-Маркова не удовлетворяется только третье условие (отсутствие автокорреляции) для (13). Присутствие автокорреляции в модели (13) может быть объяснено тем, что аппроксимация функций управления проводится многочленами, зависящими от времени, а в этом случае наличие зависимости между значением случайного отклонения в текущий момент времени от его значений в предыдущие моменты считается допустимым [13].

Таким образом, модель (12), (13) адекватно описывает ситуацию на макроэкономическом рынке Дании, и может быть использована, например, для вычисления краткосрочного прогноза с помощью (13).

Так, прогнозные значения для величины потребления основных фондов, численности рабочих и материальных затрат на 2004 год составили 31757.1, 2749 и 147740.5 соответственно (относительные ошибки прогнозов -0.478%, 0.289% и 0.0247%). Доверительные интервалы для прогнозных значений p , q и r равны 31757.1 ± 594.2 , 2749 ± 53 и 147740.5 ± 3670.1 (величины доверительных интервалов в процентах составляют 1.87%, 1.93% и 2.48%). Они оказались

достаточно узкими, что говорит о хороших прогнозных свойствах построенной модели. Отметим, что прогнозное значение величины ВВП на 2004 год, вычисленное с помощью функции $G(p, q, r)$ (12), при найденных значениях p, q и r , равно 196930.9 (относительная ошибка прогноза равна -0.148%).

Проверим прогнозные свойства модели (12), (13) на два шага вперед. Используя в (13) в качестве исходных данных вычисленные прогнозные значения на 2004 год, находим величины потребления основных фондов, численности рабочих и материальных затрат на 2005 год. Они равны 32843.7, 2756 и 156755.7 соответственно (относительные ошибки прогнозов -1.23%, 0.19% и 1.01%), а доверительные интервалы для них 32843.7 ± 1291.1 , 2756 ± 87 , 156755.7 ± 6599.4 (величины доверительных интервалов в процентах составляют 3.93%, 3.15% и 4.21%). В данном случае доверительные интервалы несколько расширились, так как при их расчете приходится учитывать случайность вектора объясняющих переменных. Однако качество построенных прогнозов, как и в предыдущем случае, оказалось высоким. Прогнозное значение для величины h на 2005 год, вычисленное с помощью (12) оказалось равным 206007.1 (относительная ошибка прогноза равна -1.23%).

Проведя аналогичные расчеты, но уже на данных 1990-2005 гг., можно вычислить прогнозные значения величин потребления основных фондов, численности рабочих и материальных затрат на 2006 год и сравнить их с экспертными оценками, приведенными в последней строке табл. 1. Так, прогнозные значения величин p, q, r и h получились равными 34519.2, 2770, 165483.7 и 218097.2 (ошибки найденных прогнозов относительно экспертной оценки -1.68%, 0.12%, 2.04% и -1.72%), что говорит о высокой степени соответствия найденных по модели (12), (13) и экспертных прогнозов.

Рассмотрим случай использования в качестве модели, задающей связь между факторами изучаемой макроэкономической системы, модель (8). Для экономики Дании (табл. 1) функция выигрыша равна (12). Аналог (8) в данном случае:

$$\begin{aligned} p^{t+1} &= P + b_{10} \sum_{i=0}^t \frac{\partial G}{\partial p} \bigg|_{t=i} + b_{11} \sum_{i=0}^t i \frac{\partial G}{\partial p} \bigg|_{t=i} + b_{12} \sum_{i=0}^t i^2 \frac{\partial G}{\partial p} \bigg|_{t=i} + \dots + b_{1k_1} \sum_{i=0}^t i^{k_1} \frac{\partial G}{\partial p} \bigg|_{t=i} + u_{k_1}^t, \\ q^{t+1} &= Q + b_{20} \sum_{i=0}^t \frac{\partial G}{\partial q} \bigg|_{t=i} + b_{21} \sum_{i=0}^t i \frac{\partial G}{\partial q} \bigg|_{t=i} + b_{22} \sum_{i=0}^t i^2 \frac{\partial G}{\partial q} \bigg|_{t=i} + \dots + b_{2k_2} \sum_{i=0}^t i^{k_2} \frac{\partial G}{\partial q} \bigg|_{t=i} + u_{k_2}^t, \quad (14) \\ r^{t+1} &= R + b_{30} \sum_{i=0}^t \frac{\partial G}{\partial r} \bigg|_{t=i} + b_{31} \sum_{i=0}^t i \frac{\partial G}{\partial r} \bigg|_{t=i} + b_{32} \sum_{i=0}^t i^2 \frac{\partial G}{\partial r} \bigg|_{t=i} + \dots + b_{3k_3} \sum_{i=0}^t i^{k_3} \frac{\partial G}{\partial r} \bigg|_{t=i} + u_{k_3}^t. \end{aligned}$$

Используя матричную форму (9) и ее аналоги для второго и третьего уравнений в (14), можно найти оценки неизвестных коэффициентов в функциях управления и констант P, Q и R . Получим:

$$u_1(t) = 355.147 + 8.169t, \quad P = 16025.695;$$

(с.о.) (17.379) (2.796)

$$u_2(t) = -2.0694 + 1.104t - 0.0859t^2, \quad Q = 2650.333;$$

(с.о.) (0.288) (0.107) (0.00834)

$$u_3(t) = 7266.0158, \quad R = 80450.324.$$

(с.о.) (146.178)

Здесь все коэффициенты построенных моделей значимы (при уровне значимости $\alpha = 0.05$). При указанных функциях управления факторы p , q и r приближаются к коэффициентам детерминации равными 0.99832, 0.98068 и 0.99517 соответственно. Проверка условий Гаусса-Маркова для остаточных членов в (14) дает результат, аналогичный модели (13).

Следовательно, модель (12), (14) адекватно описывает динамику выбранных макроэкономических показателей для Дании, и пригодна для целей прогноза.

Прогнозные значения для потребления основного капитала, численности рабочих и материальных затрат на 2004 год, вычисленные с помощью (14), составили 32327.9, 2675 и 146293.8 соответственно (относительные ошибки прогнозов 1.30%, -2.45% и -0.96%). Доверительные интервалы для них равны 32327.9 ± 934.7 , 2675 ± 99 и 146293.8 ± 4581.1 (величины доверительных интервалов в процентах составляют 2.89%, 3.72% и 3.13%). Прогноз величины h на 2004 год есть 195253.4 (относительная ошибка прогноза -1.01%).

Соответствующие расчеты позволяют найти прогнозные значения изучаемых величин на 2005 год. Они равны: 33603.6 для p , 2681 для q , 151021.2 для r и 201969.7 для h (относительные ошибки прогнозов 1.06%, -2.98%, -2.75% и -3.26%), а соответствующие доверительные интервалы для p , q и r составили 33603.6 ± 1519.3 , 2681 ± 158 и 151021.2 ± 9531.4 (величины доверительных интервалов в процентах 4.52, 5.89% и 6.31%).

Аналогичные расчеты на данных 1990-2005 гг. позволяют сравнить прогнозные значения, найденные с помощью (12), (14), и соответствующие экспертные оценки значений изучаемых величин на 2006 год. В результате вычислений с помощью (14) было получено, что прогнозные значения величины потребления основного капитала, численности рабочих, материальных затрат и ВВП на 2006 год, равны 34745.4, 2752, 156827.7 и 212442.5 (ошибки найденных прогнозов относительно экспертной оценки -1.02%, -0.77%, -3.36% и 7.16%). Как видим, в этом случае также наблюдается высокая степень соответствия модельных и экспертных прогнозных значений.

Сравнивая модели (12), (13) и (12), (14), следует отметить, что у первой, вследствие отсутствия временного лага, значительно выше точность прогноза и уже доверительный интервал построенных прогнозных значений, что говорит о ее хороших прогнозных свойствах, однако у второй лучше дескриптивные свойства. Отсюда можно сделать вывод, что для изучения поведения макроэкономической системы в будущих периодах, необходимо использовать модель (12), (13), а для ретроспективного анализа системы – модель (12), (14).

Результаты аналогичного моделирования для Финляндии, Франции, Италии, Японии, Нидерландов и США приведены в табл. 2. В ней указаны прогнозные значения моделируемых величин (верхнее число), а также относительные ошибки прогноза и величина доверительного интервала (нижние числа) в процентах от прогнозного значения. В качестве функции-результата разрешения конфликта выбиралась трехфакторная модель типа (3). Если она содержала незначимые коэффициенты, то после их удаления она вырождалась в двухфакторную (2). Анализ табл. 2 подтверждает сказанное выше, что модель (12), (13) обладает лучшими прогнозными, а (12), (14) – лучшими дескриптивными свойствами.

Таблица 2. Результаты моделирования для Финляндии, Франции, Италии, Японии, Нидерландов и США

Страна	Потребление основных фондов p			Численность рабочих q			Материальные затраты r		
	R^2	2004	2005	R^2	2004	2005	R^2	2004	2005
Для модели (1), (6)									
Финляндия	0.932	22237.0 -0.73; 1.78	23734.9 -1.85; 3.61	0.864	2398 -1.38; 2.03	2390.8 0.26; 4.21	—	—	—
Франция	0.960	209874.3 1.30; 2.31	216500.7 1.85; 4.01	0.889	25418 -1.79; 2.05	25922 -3.45; 4.11	0.987	1339665.2 -0.58; 1.53	1407865.4 -1.86; 3.21
Италия	—	—	—	0.940	24583 -1.43; 2.35	24753 -1.91; 4.52	0.985	1079854.7 0.93; 2.22	1107214.2 1.30; 3.88
Япония	0.930	910333.3 -3.04; 3.12	918687.9 -2.99; 5.52	—	—	—	0.902	2709948.7 3.76; 4.39	2649460.9 4.50; 6.85
Нидерланды	0.945	74377.4 -1.26; 2.92	76185.5 -1.03; 3.67	—	—	—	0.965	364853.0 -1.26; 2.12	363714.9 1.42; 3.41
США	0.937	1036859.5 -1.89; 2.74	1057566.9 2.62; 3.95	0.924	138080 1.41; 2.68	139724 1.81; 4.47	0.937	8323649.7 2.86; 4.18	8371147.1 2.82; 5.94
Для модели (1), (8)									
Финляндия	0.981	22567.1 2.19; 3.52	24102.4 3.35; 5.47	0.946	2419.1 2.23; 3.65	2489.3 3.71; 6.03	—	—	—
Франция	0.987	211839.9 0.36; 2.44	220996.7 -0.22; 4.36	0.927	25009.5 -0.19; 2.21	25260.1 -0.92; 5.30	0.994	1310524.7 1.63; 2.03	1355943.5 1.90; 4.05
Италия	—	—	—	0.963	23495.9 3.13; 3.74	23627.0 2.77; 5.46	0.991	1059082.1 2.90; 3.45	1086423.7 3.24; 6.13
Япония	0.953	921095.2 -4.17; 4.55	928585.2 -4.02; 7.16	—	—	—	0.923	2917384.3 -4.60; 5.21	2912519.4 -4.94; 7.73
Нидерланды	0.969	72299.1 1.57; 3.43	75060.1 0.45; 5.05	—	—	—	0.982	360944.2 -0.19; 2.63	375094.1 -1.65; 4.74
США	0.962	1043251.9 -2.49; 3.38	1122873.5 -3.35; 5.33	0.966	137283.8 2.00; 3.03	146650.8 3.00; 5.72	0.970	8356617.2 -3.24; 4.67	8824546.2 -2.46; 6.83

Выводы. Разработана и исследована методика построения динамических моделей в дискретном времени для макроэкономических систем, обосновывается их адекватность. Разработанный метод идентификации неизвестных параметров моделей, построенных на основании дифференциальной системы уравнений градиентного типа, позволяет оценить с высокой точностью функции управления, применяемые участниками рынка.

Показано, что модели (1), (6) и (1), (8) обеспечивают высокое согласование со статистическими данными по реальным системам. Анализ результатов моделирования макроэкономики семи европейских стран выявил, что модель (1), (6) обладает хорошими прогнозными свойствами и может использоваться для построения высокоточных прогнозов. Другая модель (1), (8) обладает лучшими дескриптивными свойствами и может применяться для изучения связей между элементами системы и для анализа ее свойств. В дальнейшем представляет интерес исследование реальных рынков с большим числом участников.

ЛИТЕРАТУРА

1. Шніпко О.С. Моделювання ВВП та впливу на нього показників господарської діяльності підприємств окремих галузей промисловості України//Формування ринкових відносин в Україні.– 2006. – №1.– С.11-19
2. Муха О.В. Эконометричний аналіз і прогнозування інфляції на сучасному етапі в Республіці Білорусь//Статистика України. – 2005. – №4. – С. 20-27.
3. Гранберг А.Г. Моделирование социалистической экономики. – М.: Экономика, 1988. – 487 с.
4. Чикрий А.А. Конфликтно управляемые процессы. – Киев: Наук. думка, 1992. – 384 с.
5. Weidlich W. Sociodynamics: A Systematic Approach to Mathematical Modelling in the Social Sciences. – Harwood: Academic Publishers, 2000.
6. Antipin A. Gradient approach of computing fixed points of equilibrium problems. // Journal of Global Optimization, 2001. – P. 1-25.
7. Назаренко А. М. Об эконометрико-игровом методе построения и идентификации математических моделей макроэкономических процессов // Механизм регулирования экономики. – Сумы: ИТД «Университетская книга», 2006. – №1. – С. 105-114.
8. Макконелл К.Р., Брю С.Л. Экономикс: принципы, проблемы, политика: Пер. с 13-го англ. изд. – М.: ИНФРА-М, 1999. – XXXIV, 974 с.
9. Васильев Ф.П. Численные методы решения экстремальных задач: Учеб. пособие для вузов. – 2-е изд., перераб. и доп. – М.: Наука, 1988. – 552 с.
10. Альбрехт Э.Г., Быстрай Г.П. О динамических моделях эволюции некоторых макроэкономических процессов // Исследование федерализма в России: междисциплинарный подход. – Екатеринбург: Институт философии и права УрО РАН, 1999. – С. 214-232.
11. Назаренко О.М. Основи економетрики: Вид. 2-ге, перероб.: Підручник. – Київ: „Центр навчальної літератури”, 2005. – 392 с.
12. <http://epp.eurostat.ec.europa.eu/>
13. Доугерти К. Введение в эконометрику: Пер. с англ. – М.: ИНФРА-М, 2001. – XIV, 402 с.

Вісник Харківського національного університету
Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»

УДК 004.822

№ 733, 2006, с.47-57

Определение ограничений к системе интеграции информации с использованием онтологий и пример реального приложения

В.Н. Владимиров

Запорожский национальный университет, Украина

In this article we consider existing ontology-based approaches to the integration of autonomous, distributed and heterogeneous information sources. We identify similar pitfalls for this domain and consider existing solutions for them. Basing on these solutions we propose possible system implementation restrictions to make possible development of framework of intelligent ontology-driven information retrieval from distributed, heterogeneous, legally and physically autonomous IR in the frame of the organizational network of the National Higher Education System.

1. Введение в проблематику и постановка задачи

На данный момент в различных сферах промышленности и хозяйства широко используются компьютерные информационных систем. Методы сбора, обработки информации в них являются хорошо разработанными. Современные задачи требуют полного доступа к информации хранящейся в этих информационных системах, являющихся распределенными и несогласованными между собой. Поэтому разработка способов получения информации от распределенных, гетерогенных, физически автономных информационных ресурсов (ИР) является важным направлением научных исследований.

Эти исследования относятся к сфере распределенного интеллектуального извлечения информации, или в более широком смысле – интеллектуальной интеграции информации. В этой сфере последние 10 лет ведутся интенсивные исследования. Например, в Information Society Technologies Key Action Line of the EU FP6 и подобных ей национальных и международных сетях. Другими примерами исследовательских проектов разрабатывающих формальные, алгоритмические, архитектурные инфраструктуры, прототипы программного обеспечения для распределенного интеллектуального извлечения информации из распределенных, разнородных ИР и интеллектуальной интеграции информации являются: BUSTER [1], DOME ([2], [3]), InfoSleuth [4], KRAFT [5], MOMIS [6], OBSERVER [7], Ontobroker [8], PICSEL [30], SIMS [9], TSIMMIS [10], и другие.

Непосредственно в данной работе нас интересуют вопросы, возникающие на этапах анализа требований и проектирования интегрированных гетерогенных, физически автономных систем обмена информацией.

В связи с этим отметим высокую актуальность реальной задачи создания современной системы обмена информацией UnIT-NET IEDI[†].

[†] UnIt-NET: IT in University Management Network. <http://www.unit-net.org.ua/>. Описание в [18]

Поэтому целью нашей работы является разработка концепции, позволяющей систематизировать и применять в конкретных приложениях теорию и опыт использования онтологий и проверить эту концепцию в задаче анализа требований для UnIT-NET IEDI.

Нашу задачу поставим так: на базе сложившейся к настоящему времени модели объединения распределенных первоначально несогласованных компьютерных систем [14-16] разработать концептуальную точку зрения, позволяющую согласовать до степени прикладной применимости результаты текущих исследований [11] и проверить эффективность такого подхода путем его приложения к выработке общих требований для проекта UnIT-NET IEDI.

2. Проблема взаимодействия

Проблема объединения несогласованных и распределенных компьютерных систем (КС) известна также как проблема взаимодействия.

[14] предлагает следующую классификацию проблем взаимодействия:

- *системные*: Они возникают в случае интеграции КС с различным аппаратным и программным обеспечением (например, операционными системами);
- *синтаксические*: возникают в случае, когда в интегрируемых КС используются различные языки и представления данных;
- *структурные*: связаны с использованием различных моделей данных;
- *семантические*: связаны со значениями терминов используемых при обмене информацией между интегрируемыми КС.

Группа проблем семантического взаимодействия также известна как проблемы семантической несогласованности [12].

Для достижения семантической согласованности в несогласованной информационной системе, должно быть достигнуто общее для всей системы понимание о смысле и значении передаваемой информации. Семантические конфликты возникают всякий раз, когда два контекста используют различные интерпретации информации.

Выделяют три основных случая семантической неоднородности [15]:

- *конфликты смешения смысла* – возникают в случаях, когда единицы информации имеют одинаковое название в разных случаях, а их значения – различаются. Например, значение выражения «последняя цена на торгах» зависит от времени торгов, указываемом в контексте;

- *конфликты систем измерения* – возникают, когда для измерения значений используются различные справочные системы измерения. Пример – различные валюты;

- *конфликты наименования* – возникают в случаях, когда используются различные схемы информации. Одним из признаков такого конфликта является наличие синонимов и омонимов.

Использование онтологий для уточнения подразумеваемых значений информации – возможный подход к решению проблемы семантической неоднородности. Достижение взаимодействия является ключевой сферой применения для онтологий [16]. На данный момент разработано множество подходов к интеграции информации на основе онтологий.

Делаем вывод, что для решения задачи интеграции можно использовать техники, подходы и программные парадигмы, которые выделяют похожие сложности в предметной области:

1) Сложности связанные со способом, которым решается проблема семантической несогласованности при интеграции информации с помощью онтологий;

2) Сложностей связанные с вопросами предоставления автономности и динамической сущности элементов открытой системы;

3) Задачи формулирования запросов, эффективной декомпозиции запросов без потери информации, уточнение и объединение результатов запросов.

Применительно к системе UnIT-NET IEDI это позволяет определить вопросы, требующие решения – проблема семантической неоднородности с помощью использования онтологий, обеспечение автономности и динамической сущности элементов системы, формулирование запросов и их эффективная декомпозиция.

3. Подходы к решению проблемы семантической несогласованности

Первая группа сложностей связана со способом, которым решается проблема семантической несогласованности при интеграции информации с помощью онтологий. Как выделено в [2], первая группа включает вопросы разработки (подходы снизу-вверх и сверху вниз) и использования онтологий, построения отображений между онтологиями, и установления связей между онтологиями и информационными ресурсами, выступающими в роли поставщиков информации.

3.1 Роль онтологий при интеграции информации

Изначально онтологии представляются как эксплицитная спецификация концептуализации [17]. Поэтому онтологии могут быть использованы при решении задачи интеграции для описания семантики информационных ресурсов (ИР). Онтологии могут быть использованы для идентификации и установления соответствия между семантически сходными компонентами.

Почти во всех подходах к интеграции информации на основе онтологий последние используются для явного описания семантики информационных источников. Большинство проектов применяют один из следующих подходов использования онтологий ([11]): одна онтология (SIMS), несколько онтологий (OBSERVER), гибридный подход (BUSTER, DOME).

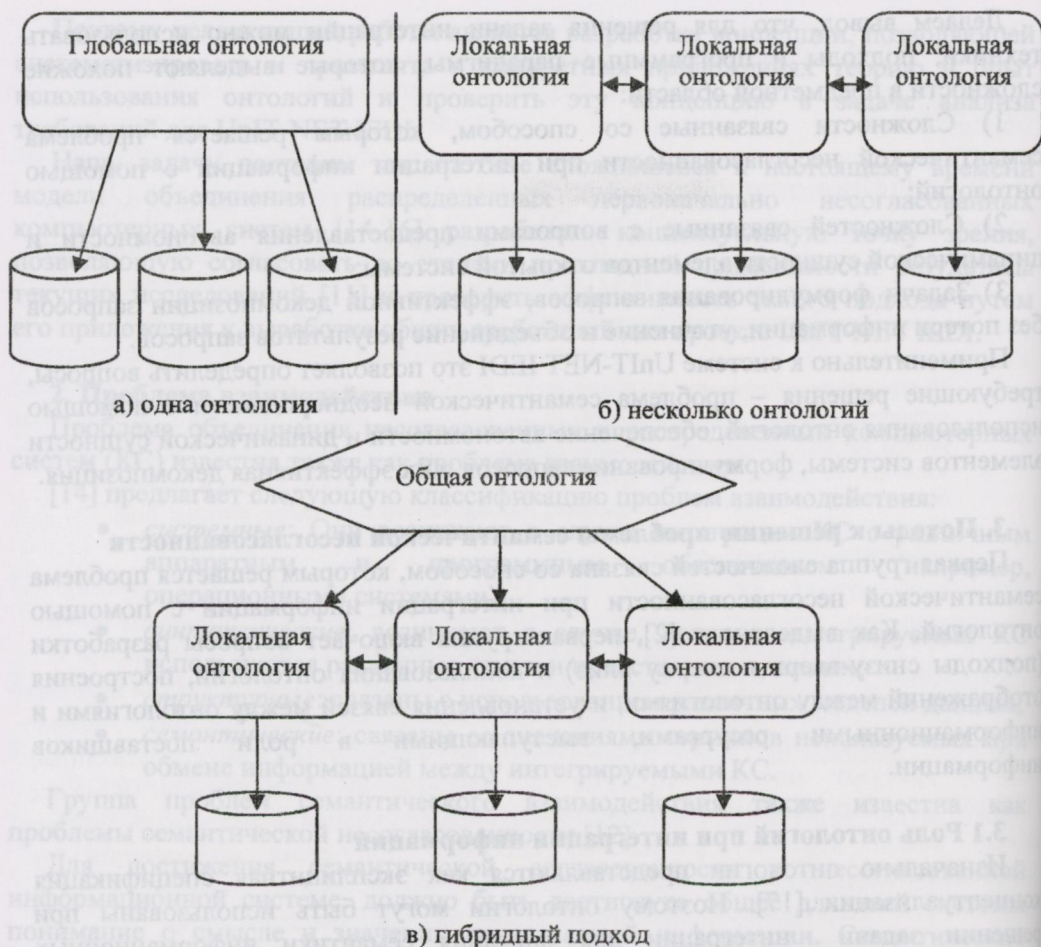


Рисунок 1. Подходы к использованию онтологий

Таблица 1. Сравнение различных подходов к интеграции ИР с помощью онтологий

	Одна онтология	Множество онтологий	Гибридный
Усилия на внедрения и разработку	Небольшие	Большие	разумные
Степень поддерживаемой семантической неоднородности между информационными ресурсами	Низкая (похожие представления о предметной области)	Поддерживает неоднородные представления	Поддерживает неоднородные представления
Действия необходимые для добавления/удаления ИР	Требует изменений в глобальной онтологии	Требует 1) построение и предоставление новой онтологии ИР;	построение и предоставление новой онтологии ИР

		2) установление связей между новой онтологией ИР и уже существующих онтологий ИР	
Сравнение нескольких онтологий	Не требуется	Является сложной задачей из-за отсутствия общего словаря	Является простой задачей, так как онтологии ИР используют общий словарь
Примеры реализации подходов	SIMS, ONTOLINGUA	OBSERVER	COIN, MECOTA, BUSTER

Возвращаясь к системе UnIT-NET IEDI можно сказать, что для решения задачи интеграции информации на основе онтологий в данном случае наиболее подходит гибридный подход к использованию онтологий.

3.2 Отображения между онтологиями и ИР

Связи между онтологиями и схемами информационных ресурсов устанавливаются с помощью отображений между элементами онтологий и элементами схем данных. Как указано в [2], причинами для построения этих отображений являются:

- Определения схем данных не всегда являются подходящим источником знаний о предметной области для людей запрашивающих информацию в системе, обычно эти схемы выполняют только техническую роль;
- Запросы, сформулированные к системе, выражаются на онтолого-ориентированном языке запросов без использования схемы данных. Поэтому построение и использование отображений между элементами онтологий и элементами схем данных позволяет автоматически выполнять пользовательские запросы в системе.

Как правило, информация в ИР храниться в базе данных. Онтология ИР может быть связана как со схемой данных, так и с отдельными терминами из базы данных ИР.

Выделяют [11] следующие подходы к построению отображений между онтологиями и ИР:

- *Использование структурного сходства.* Онтология строится путем точного копирования структуры базы данных и представления ее с помощью языка поддерживающего автоматический вывод. В этом случае выполняется интеграция с копией модели данных ИР. Этот подход реализован в системе TSIMMIS [20] и медиаторе системы SIMS [19];
- *Определение терминов.* Онтология используется для описания семантики терминов из базы данных, определяя соответствующие им

концепты. В этом случае онтология ИР уже не зависит от структуры базы данных ИР. Этот подход реализован в системе BUSTER [21];

- *Расширение структуры.* Этот подход является наиболее распространенным. В нем комбинируются два предыдущих подхода. Логическая модель строится подобной структуре информационного ресурса, а также содержит дополнительные определения концептов. Этот подход использован для интеграции информации в системах OBSERVER [22], KRAFT [23], PICSEL [24] и DWQ[25];
- *Мета-аннотация.* Подход заключается в использовании мета-аннотаций, которые добавляют информацию о семантике в информационный ресурс. Этот подход является типичным для интеграции информации представленной в World Wide Web, где аннотирование является исходным и общепринятым способом для добавления семантики. Этот подход реализован в системах Ontobroker [26] и SHOE [27].

В конкретном случае, учитывая разнородную структуру баз данных ИР входящих в систему UnIT-NET IEDI, использование расширения структуры в качестве способа построения отображений является наиболее подходящим.

3.3 Отображения между онтологиями

В случаях, когда архитектура онтологий системы включает в себя несколько онтологий, объединенных “горизонтально” (как в подходе, использующем несколько онтологий) или “вертикально” (в гибридном подходе), необходимо построить отображения между онтологиями.

Отображения онтологий в системе предоставляют связи между элементами эквивалентными или родственными элементами онтологий, таким образом, обеспечивается повторное использование онтологий.

Выделяют [11] следующие подходы к построению отображений между онтологиями используемые в системах для интеграции информации:

- *Предопределенные отображения.* Этот подход реализован в системе KRAFT [28], где трансляция между онтологиями выполняется специальными программными агентами-медиаторами, использующими предварительно заданные правила трансформации. При реализации этого подхода используются различные виды отображений – от простых отображений один-к-одному между классами и значениями в онтологиях до отображений между областями онтологий, заданных составными выражениями. Этот подход является гибким, но не позволяет проверять сохранность семантической информации – пользователь имеет возможность задать такие правила отображения, которые не будут иметь смысла либо порождать конфликты;
- *Лексические связи.* Этот подход заключается в расширении общей модели описательной логики добавлением связями между онтологиями заимствованными из лингвистики. В системе

OBSERVER [22], в которой реализован этот подход, используются связи типа синоним, ономим, пересечение, включение и отсечение;

- *Связь с онтологией верхнего уровня.* Реализуется путем наследования концептов из общей онтологии верхнего уровня локальными онтологиями. Таким образом, устанавливаются связи между локальными онтологиями и онтологией верхнего уровня, что позволяет разрешать конфликты и неоднозначности, т.е. сравнивать онтологии между собой. Но не позволяет устанавливать прямые связи между концептами локальных онтологий. Пример реализации – система DWQ [25]
- *Семантические соответствия.* Заключается в определении семантических соответствий заданных с определенной системой. Например, используя [29] общий словарь для определения концептов различных онтологий. Позволяет избежать построения произвольных отображений между концептами и избежать проблем связанных с неоднозначностью, возникающей при построении не прямых отображений с помощью онтологии верхнего уровня, выполняемых при использовании предыдущего подхода.

В применении к системе UnIT-NET IEDI, в рамках которого предполагается использовать гибридный подход к использованию онтологий, мы можем сделать вывод, что для построения отображений между онтологиями ИР следует использовать подход, заключающийся в установлении связей с онтологией верхнего уровня.

4. Требования автономности информационных ресурсов и открытости системы

Требования автономности информационных ресурсов и открытости системы в целом являются другими причинами для использования отображений между элементами онтологий и схемами данных информационных ресурсов.

Вторая группа сложностей рассматривает вопросы предоставления автономности и динамической сущности элементов открытой системы. В качестве решений, в этой области предлагается использовать одну из медиаторных архитектур: централизованную или децентрализованную. Централизованная медиаторная архитектура подразумевает единственный центр, который содержит всю информацию об онтологиях, информационных ресурсах, их отображения, а также контролирует формулирование и выполнение запросов. Такой подход реализован в системе TSIMMIS.

Децентрализованная медиаторная архитектура подразумевает наличие отдельного агента/враппера для каждого информационного ресурса, который хранит отображения глобальной/общей онтологии (-иях) и обслуживает информационные ресурсы (InfoSleuth, SIMS, KRAFT).

Следовательно, в отношении системы UnIT-NET IEDI использование централизованной медиаторной архитектуры с центральным медиатором позволяет обеспечить целостность системы с одновременным предоставлением автономности входящим в систему ИР.

5. Задачи формулирования запросов, эффективной декомпозиции запросов

Третьей группой подзадач подходов к интеграции информации вызывающих возможные сложности являются задачи формулирования запросов, эффективной декомпозиции запросов без потери информации, уточнение и объединение результатов запросов.

Анализируя [19, 22, 24], можно выделить сложившиеся к настоящему времени подходы к решению этих задач:

- использование знаний содержащихся в онтологиях (отношения *hypernym/hyponym*) для переформулирования запросов, содержащих термины которые не определены в онтологии (-иях), для конструирования запросов без потери информации (OBSERVER);
- использование некоторых техник переписывания и отображения для получения запросов к информационным ресурсам, которые почти полностью соответствуют исходному запросу (PICSEL);
- использование глобальной онтологии при формулировании запросов в качестве глобальной модели для построения запросов. В этом случае пользователь формулирует запрос в терминах глобальной онтологии, а затем запрос трансформируется в подзапросы к ИП в терминах онтологий ИП (SIMS).

По отношению к системе UnIT-NET IEDI использование глобальной онтологии при формулировании запросов и использование техники переписывания запросов с использованием отображений для формулирования, обработки и выполнения запросов, является наиболее эффективной.

6. Выводы

В данной работе представлена авторская концепция системного подхода к использованию онтологий для решения задачи интеграции распределенных, гетерогенных, физически автономных информационных ресурсов, позволившая выяснить фактическую применимость этого подхода при разработке требований и проектных решений в рамках создания современной системы обмена информацией UnIT-NET IEDI.

Это позволило обнаружить, что некоторые из проанализированных проблем имеют только частичные решения, например, проблема семантического взаимодействия, как правило, частично решается путем поручения некоторого рода соглашений узлам-участникам, тем самым, предоставляя инфраструктуру для семантических представлений. Эти частичные решения, очевидно, ограничивают сферу приложения и функциональность реализуемых программных прототипов выполняющих задачу распределенного интеллектуального извлечения информации.

Прикладное значение настоящей работы демонстрируется тем, что по результатам исследования были определены ограничения, накладываемые в случае реализации на систему UnIT-NET IEDI - программную инфраструктуру, позволяющую обмениваться электронными данными между университетами и государственными учреждениями Украины. Сложность задачи связана с тем, что субъекты обмена физически удалены между собой, принадлежат

различным юридическим владельцам, а также их информационные ресурсы являются семантически разнородными [18]. Найденные ограничения можно свести к следующим требованиям

- IEDI строится на принципах медиаторной архитектуры с центральным медиатором
- IEDI использует гибридный подход [11] для представления знаний
- IEDI использует регистрацию информационных ресурсов перед тем как ресурс станет доступным для запросов
- IEDI не предоставляет полную автоматизацию для процессов построения отображений и выравнивания онтологий
- Компоненты IEDI используют технику переписывания с использованием отображений для формулирования, обработки и выполнения запросов

В рамках этих ограничений разработка формальных методов для поддержки получения информации из гетерогенных, распределенных, автономных источников, а именно механизма декомпозиции и выполнения запросов к распределенным автономным семантически гетерогенным информационным источникам является новой задачей, не имеющей готовых решений в данном контексте. Разработка адекватных решений может служить направлением дальнейших исследований.

ЛИТЕРАТУРА

1. Stuckenschmidt H., Wache H., Voegele T., Visser U.: Enabling technologies for interoperability. In: (Visser, U., Pundt H. Eds.) Workshop on the 14th International Symposium of Computer Science for Environmental Protection, Bonn, Germany, 2000, pp. 35-46.
2. Cui, Z., Jones, D., O'Brien, P.: Issues in Ontology-based Information Integration. In: (A. Gomez-Perez, M. Gruninger, H. Stuckenschmidt, M. Uschold) Proceedings of the IJCAI-01 Workshop on Ontologies and Information Sharing, Seattle, USA, August 4-5, 2001, pp. 141-146.
3. Cui, Z., Jones, D., O'Brien, P.: Semantic B2B Integration: Issues in Ontology-based Applications. SIGMOD Record, Vol. 31, No. 1, March 2002. Pp. 43-48
4. Bayardo et al.: InfoSleuth: Semantic Integration of Information in Open and Dynamic Environment. In Proceedings of the 1997 ACM International Conference on the Management of Data (SIGMOD), Tucson, Arizona, May 1997.
5. Gray, P., Preece A., Fiddian N., Gray W., Bench-Capon T., Shave M., Azarmi N., Wiegand M.: KRAFT: Knowledge Fusion From Distributed Databases and Knowledge Bases. In: Proc. 8th Intl. Workshop on Database and Expert System Applications (DEXA-97), IEEE Press, pp. 682-691.
6. Bergamaschi, S., Castano, S., De Capitani di Vimercati, S., Montanari, S. Vincini, M.: An Intelligent Approach to Information Integration. In: Proc. Of Formal Ontology in Information Systems (FOIS-98), June, 1998.
7. Kashyap V., Sheth A.: Information Brokering across Heterogeneous Digital Data: A Metadata-based Approach. Kluwer Academic Publishers, 2000

8. Decker, S., Erdmann, M., Fensel, D., and Studer, R.: Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In R. Meersman et al. (eds.): *Semantic Issues in Multimedia Systems. Proceedings of DS-8*. Kluwer Academic Publisher, Boston, 1999, 351-369.
9. Arens, Y., Knoblock, C.A., Shen, W.: Query Reformulation for Dynamic Information Integration. *Journal of Intelligent Information Systems*. 1996.
10. Garcia-Molino, H. et. al.: The TSIMMIS Approach to Mediation: Data Models and Languages. In *Proceedings of the NGITS (Next Generation Information Technologies and Systems)*, June 1995.
11. Wache, H., Vogeles, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H., and Hubner, S.: Ontology-Based Integration of Information - A Survey of Existing Approaches. In: (A. Gomez-Perez, M. Gruninger, H. Stuckenschmidt, M. Uschold) *Proceedings of the IJCAI-01 Workshop on Ontologies and Information Sharing*.
12. Won Kim and Jungyun Seo. Classifying schematic and data heterogeneity in multidatabase systems. *IEEE Computer*, 24(12):12-18, 1991. problem classification of semantic heterogeneity.
13. V. Kashyap and A. Sheth. Schematic and semantic similarities between database objects: A context-based approach. *The International Journal on Very Large Data Bases*, 5(4):276-304, 1996.
14. A.P. Sheth. Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics. M. F. Goodchild, M. J. Egenhofer, R. Fegeas, and C. A. Kottman (eds.) *Interoperating Geographic Information Systems*, Kluwer.
15. Cheng Hian Goh. Representing and Reasoning about Semantic Conflicts in Heterogeneous Information Sources. Phd, MIT, 1997.
16. Mike Uschold and Michael Gruninger. Ontologies: Principles, methods and applications. *Knowledge Engineering Review*, 11(2):93-155, 1996.
17. Tom Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199-220, 1993.
18. Bulat, A., Ermolayev, V., Gray, E., Keberle, N., Plaksin, S., Shapar, V., Vladimirov, V., Zholtkevich, G. (2004) *The Infrastructure for Electronic Data Interchange. Reference Architecture Specification. Version 1.0. UNIT-NET Deliverable No D2.2.D.1*, Feb. 2004.
19. Yigal Arens, Chun-Nan Hsu, and Craig A. Knoblock. Query processing in the sims information mediator. In *Advanced Planning Technology*. AAAI Press, California, USA, 1996.
20. S. Chawathe, H. Garcia-Molina, J. Hammer, K. Ireland, Y. Papakonstantinou, J. Ullman, and J. Widom. The tsimmis project: Integration of heterogeneous information sources. In *Conference of the Information Processing Society Japan*, pages 7-18, 1994.
21. Heiner Stuckenschmidt and Holger Wache. Context modelling and transformation for semantic interoperability. In *Knowledge Representation Meets Databases (KRDB 2000)*. 2000.
22. E. Mena, V. Kashyap, A. Sheth, and A. Illarramendi. Observer: An approach for query processing in global information systems based on interoperability between

- pre-existing ontologies. In Proceedings 1st IFCIS International Conference on Cooperative Information Systems (CoopIS '96). Brussels, 1996.
23. A.D. Preece, K.-J. Hui, W.A. Gray, P. Marti, T.J.M. Bench-Capon, D.M. Jones, and Z. Cui. The kraft architecture for knowledge fusion and transformation. In Proceedings of the 19th SGES International Conference on Knowledge-Based Systems and Applied Artificial Intelligence (ES'99). Springer, 1999.
24. Francois Goasdoue, V.eronique Lattes, and Marie-Christine Rousset. The use of carin language and algorithms for information integration: The picsele project. International Journal of Cooperative Information Systems (IJCIS), 9(4):383 – 401, 1999.
25. Diego Calvanese, Giuseppe De Giacomo, and Maurizio Lenzerini. Description logics for information integration. In Computational Logic: From Logic Programming into the Future (In honour of Bob Kowalski), Lecture Notes in Computer Science. Springer- Verlag, 2001
26. D. Fensel, I. Horrocks, F. Van Harmelen, S. Decker, M. Erdmann, and M. Klein. Oil in a nutshell. In 12th International Conference on Knowledge Engineering and Knowledge Management EKAW2000, Juanles- Pins, France, 2000.
27. Jeff Heflin and James Hendler. Semantic interoperability on the web. In Extreme Markup Languages 2000, 2000.
28. A.D. Preece, K.-J. Hui, W.A. Gray, . Marti, T.J.M. Bench-Capon, D.M. Jones, and Z. Cui. The kraft architecture for knowledge fusion and transformation. In Proceedings of the 19th SGES International Conference on Knowledge-Based Systems and Applied Artificial Intelligence (ES'99). Springer, 1999.
29. Holger Wache. Towards rule-based context transformation in mediators. In S. Conrad, W. Hasselbring, and G. Saake, editors, International Workshop on Engineering Federated Information Systems (EFIS 99), Kuhlungsborn, Germany, 1999. Infix-Verlag.
30. Lattes V., Rousset M.-C.: The Use of CARIN Language and Algorithms for Information Integration: The PICSEL System. International Journal of Cooperative Information Systems, Vol.9, No.4, 2000, pp.383-401.