

Полученная таким образом ОЭХ имеет вид таблицы, каждая строка которой выражает зависимость суммарного оптимального расхода теплоты  $Q_0$  на работающие агрегаты от

– заданных производственной  $Q_n$ , теплофикационной  $Q_m$  и электрической  $N$  нагрузок в случае работы турбины по электрическому графику;

– заданных производственной  $Q_n$  и теплофикационной  $Q_m$  нагрузок в случае работы турбины по тепловому графику.

При работе турбин по тепловому графику конкретными значениями производственного и теплофикационного отборов соответствует некоторая электрическая мощность, поэтому ОЭХ представлена в виде двух зависимостей:

– расхода теплоты на группу турбин от теплофикационного и производственного отборов;

– суммарной электрической мощности турбин от теплофикационного и производственного отборов.

– При работе турбин по электрическому графику ОЭХ представлена в виде зависимости расхода теплоты на группу турбин от суммарной электрической мощности турбин, теплофикационного и производственного отборов. Для получения аналитической зависимости ОЭХ необходимо:

– провести аппроксимацию ОЭХ для её дальнейшего использования в системной задаче или для оптимизации режимов работы оборудования ТЭЦ в целом будет весьма затруднительно, поскольку графики данных зависимостей не являются гладкими из-за смены состава работающего оборудования. Для выхода из этой ситуации используется кусочно-интервальная аппроксимация ОЭХ, для построения которой необходимо реализовать следующий алгоритм: Разбить ОЭХ на участки с постоянным составом работающего оборудования.

– выделить внутри каждого участка ОЭХ поверхности и ограничивающие их линии.

– Провести аппроксимацию поверхностей каждого участка ОЭХ для получения функциональных зависимостей  $Q_0 = f_1(Q_n, Q_m)$ ,  $N = f_2(Q_n, Q_m)$  для теплового режима и  $Q_0 = f_3(Q_n, Q_m, N)$  для электрического режима.

Для получения поверхностей и ограничивающих линий каждого участка ОЭХ используется метод наименьших квадратов.

## СЕРВЕР АВТОМАТИЗАЦИИ СТРУКТУР ПОДВЕСОК МОБИЛЬНЫХ МАШИН

*А.В. Кучерявенко*

Научный руководитель – к.т.н., доцент *Н.Н. Гурский*  
*Белорусский национальный технический университет*

Для обеспечения защиты водителя, пассажиров, перевозимых грузов и элементов конструкции машины от динамических нагрузок, возникающих в результате взаимодействия колес с неровностями дороги, применяются различные структуры систем поддрессоривания мобильных машин. Каждая из таких структур характеризуется набором необходимых параметров.

При проведении вычислительных экспериментов на компьютерных моделях, позволяющих оценивать динамические качества машин, необходимо иметь возможность оперативно менять структуры подвесок. Для реализации такой задачи целесообразно объединить существующие и перспективные структуры подвесок в виде отдельного законченного программного модуля.

Наиболее эффективное использование этого модуля в общем программном обеспечении, поддерживающем весь комплекс моделирования динамики машины, возможно только на платформе СОМ - технологии. Данная технология позволяет разработать независимый модуль, например, модуль структур подвесок и затем подключать его к другим программам, безотносительно к тому, на каком языке они написаны.

В настоящей работе рассматриваются новые подходы в разработке программного обеспечения компьютерного моделирования динамики мобильных машин, основанные на

концепции СОМ технологий, в частности, на одном из расширений этой технологии – сервере автоматизации применительно к заданию параметров выбранной структуры системы подрессоривания мобильной машины.

## АВТОМАТИЗИРОВАННОЕ ТЕСТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

*А.А. Землякова*

Научный руководитель – *Ю.Б. Попова*

*Белорусский национальный технический университет*

Тестирование является одним из важнейших этапов создания конкурентоспособного программного продукта и обычно сопровождает практически каждый этап разработки в соответствии с итеративной моделью жизненного цикла ПО. И сейчас, когда рынок программного обеспечения стремительно разрастается и появляется все больше и больше крупных и мелких кампаний, предлагающих свои услуги, производители, чтобы идти в ногу со временем, все чаще сталкиваются с необходимостью применения эффективных методов, как к разработке, так и к тестированию промежуточных и конечных версий программных продуктов [1].

Выделяют множество различных видов тестирования в зависимости от уровня задач и объектов на разных стадиях и этапах разработки, а именно: структурное, сборочное, функциональное, регрессионное, нагрузочное, автоматизированное, каждое из которых выполняется на определенном этапе и имеет свои собственные объекты и критерии тестирования.

Целью данной работы является:

- изучить и адаптировать под конкретные нужды предлагаемые в настоящее время методологии автоматизации тестирования;
- ознакомиться со средствами для автоматизации функционального тестирования;
- использовать выбранную методику для выполнения автоматизации тестирования, разработанного в тестовых целях веб-ориентированного приложения (изначально содержащего некоторые дефекты);
- сделать заключение об эффективности работы с данной методологией с помощью такого инструментального средства как Silk Test 6.0, чтобы в случае хороших результатов использовать их в дальнейшем на предприятии.

Для достижения данной цели необходимо решить следующие задачи:

- исследовать существующие методы и подходы к организации тестирования приложений;
- изучить специфику тестирования веб-ориентированных приложений;
- изучить существующие методологии, направленные на реализацию автоматизированного тестирования, и инструментарий для его осуществления;
- создать тестовое веб-приложение, работающее под IIS, предварительно изучив ASP-технологии;
- попытаться воспользоваться выбранной методологией автоматизации тестирования, чтобы автоматизировать тестирование созданного приложения и сделать вывод о целесообразности ее применения;
- также для осуществления указанной выше цели проанализировать возможности некоторых средств автоматизации функционального тестирования, изучив предоставляемые ими возможности, выбрать и изучить наиболее подходящий инструмент.

### Литература

1. Тестирование программного обеспечения, Сэм Канер, Джек Фолк, Енг Кек Нгуен, Киев, Издательство "ДиаСофт", 2000г.