

В данной работе была использована следующая архитектура: сеть с одним скрытым слоем, но переменным количеством нейронов (это количество задается пользователем), нейроны имеют сигмоидальную активационную функцию. Скорость обучения и количество эпох также доступно для изменения пользователем.

В итоге удалось добиться поставленной цели: временной ряд аппроксимируется довольно точно, но здесь стоит оговориться. Для точной аппроксимации необходимо подбирать определенные значения скорости обучения, количества нейронов в скрытом слое и количества эпох.

Результаты исследования подтвердили теоретические данные и показали, что количество нейронов в скрытом слое должно превышать количество наблюдений поступающих на вход нейросети приблизительно в полтора-два раза; оптимальная скорость обучения составляет 0,25–0,3 (меньшая скорость приводит к значительным временным издержкам), а большая приводит к большим ошибочным результатам; количество эпох стоит выбирать в диапазоне 1000–10 000, меньшие значения не дают приемлемого результата, а большие значительно увеличивают время обучения нейронной сети (в совокупности с низкой скоростью обучения для пятидесяти наблюдений время обучения может составить до нескольких дней).

На данный момент ведется разработка следующей версии программного продукта, в ходе которой поставлены следующие задачи:

- Повысить производительность нейронной сети и точность прогноза, за счет изменения архитектуры сети, принципа подачи данных и увеличения параметров, от которых зависит результат прогноза.
- Избавиться от чрезмерной зависимости от серверной базы данных MS SQL Server с введением возможности использования других баз данных и хранения информации в популярных файловых форматах, таких как csv и xml.
- Ввести возможность сохранения результатов работы с программным продуктом в виде проекта.

## **ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ИНФОРМАЦИОННО-МОДЕЛИРУЮЩЕЙ СИСТЕМЫ ШЛАКОВОГО РЕЖИМА ДОМЕННЫХ ПЕЧЕЙ ОАО «ММК»**

**© А.В. Костромин, В.В. Лавров, Н.А. Спирин, 2012**

*ФГАОУ ВПО «Уральский федеральный университет имени первого Президента России Б.Н. Ельцина», г. Екатеринбург*

Черная металлургия является одной из крупнейших отраслей промышленности. Объемы производства черных металлов колоссальны, как и область их применения – от изготовления столовых приборов до всевозможных инструментов, металлоконструкций и электрооборудования. Черную металлургию можно позиционировать как основополагающее производство, обладающее исключительной значимостью для экономики России.

Одной из основных задач черной металлургии в научном плане и практической деятельности специалистов на современном этапе является поиск методов повышения производительности металлургических агрегатов, качества продукции и сокращения издержек ресурсов. Эти задачи продиктованы прежде всего экономикой предприятий – требуется максимизировать прибыль при минимизации издержек (экономические ресурсы, вовлекаемые в производство, ограничены).

Шлаковый режим доменной плавки в существенной степени определяет основные ключевые показатели работы доменной печи – удельный расход кокса и производительность. Состав шлака, его количество и свойства оказывают воздействие на газодинамические,

тепловые и восстановительные процессы, протекающие в печи [1]. Одним из способов эффективного управления шлаковым режимом является моделирование его свойств на основе современных достижений в области математического моделирования доменного процесса. Моделирование шлакового режима выполняется в целях решения следующих технологических задач [2]:

- получение шлака, обладающего надлежащими свойствами в ходе всего процесса шлакообразования, начиная со стадии размягчения, последующей фильтрации первичного шлакового расплава в коксовой насадке и заканчивая получением конечного шлакового расплава;
- получение чугуна требуемого химического состава при допустимом содержании в нем серы;
- получение шлака, обеспечивающего достаточную газопроницаемость зоны шлакообразования, ровного схода шихты в печи, а также устойчивое тепловое состояние доменной плавки.

Вышеперечисленные задачи решает разработанная информационно-моделирующая система, которая используется инженерно-технологическим персоналом для оценки шлакового режима доменных печей ОАО «ММК» [2].

Система представляет собой комплекс программно-технических средств и базируется на архитектуре клиент-сервер. Серверная часть представляет собой хранилище данных под управлением операционной платформы Microsoft Windows и сервера баз данных Microsoft SQL Server 2005, расположенного в информационно-вычислительном центре доменного производства. В роли клиента выступает программное графическое Windows-приложение, работающее под управлением среды CLR платформы .NET Framework. Архитектура информационной системы представлена на рис. 1.

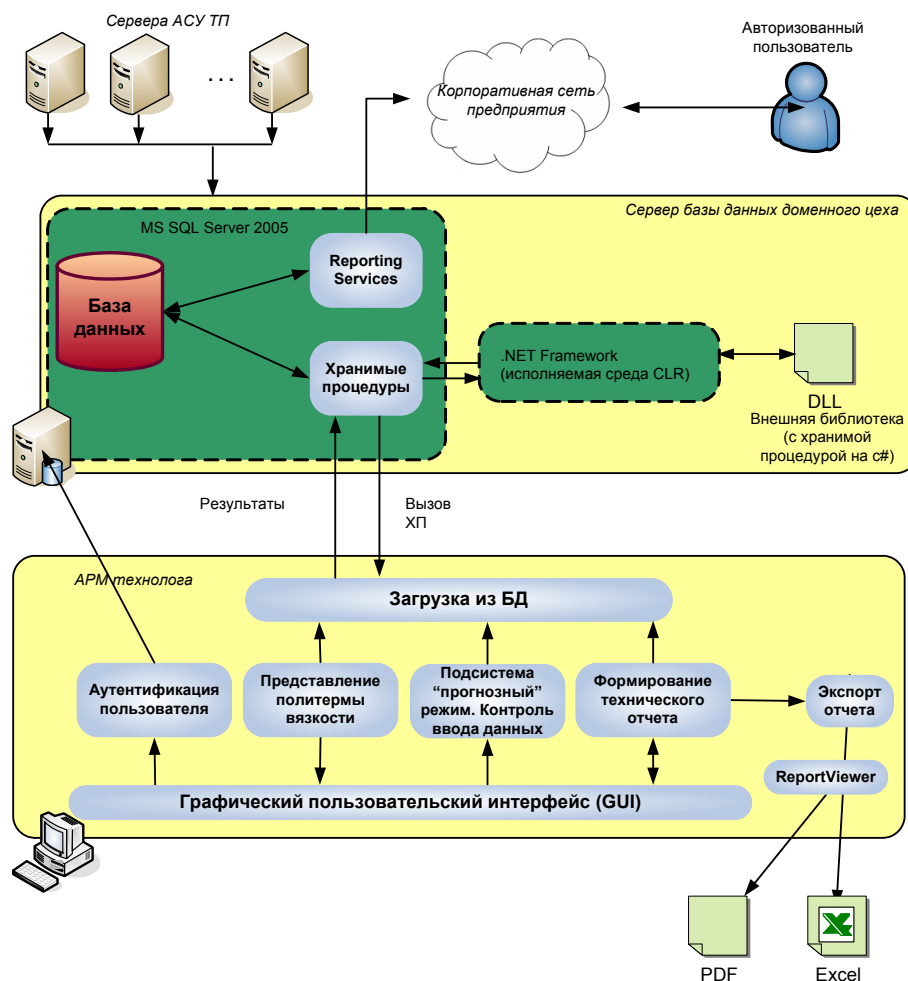


Рис. 1. Архитектура информационной системы

*Серверная часть* решает следующие задачи: безопасное и надежное хранение данных, разграничение прав доступа к данным, обеспечение целостности и непротиворечивости данных, высокоскоростной и бесперебойный доступ к данным. Дополнительно серверная часть информационной системы реализует подсистему автоматической обработки данных. Данная подсистема обеспечивает возможность обработки больших объемов данных в автоматическом режиме математическими моделями для получения показателей оценки шлакового режима доменной плавки. Реализация математических алгоритмов расчета выполнена в виде программных библиотек динамической компоновки (DLL-библиотек), за управлением которых следит Microsoft SQL Server. Использование DLL-библиотек как объектов базы данных возможно при построении последних на управляемом программном коде в связке с Microsoft SQL Server 2005 и выше [3]. Основные преимуществ интеграции Microsoft SQL 2005 и DLL на основе реализации объектов управляемого кода базы данных на платформе .NET Framework следующие:

- расширенная модель программирования;
- улучшенная стабильность и повышенная надежность;
- улучшенная безопасность;
- общая среда разработки систем;
- повышенная производительность.

Следует также отметить, что на серверах АСУ доменного цеха развернута служба Microsoft Reporting Service, которая обеспечивает доступ к отчетам по работе шлакового режима доменного цеха для корпоративных клиентов через Web-интерфейс.

*Клиентская часть* представляет собой Windows приложение, разработанное с помощью интегрированной среды разработки Microsoft Visual Studio 2008 на платформе .NET Framework, язык С#. Клиентское приложение позволяет пользователю посмотреть и рассчитать фактические показатели работы шлакового режима (вязкость, основность и т.д.), прогнозировать показатели, а также работать с показателями в сравнительном периоде. Каждый из режимов работы приложения оснащен удобным графическим интерфейсом, позволяет построить графические зависимости и отчет по результатам работы. Основную функцию клиентского приложения выполняет подключаемая математическая модель, реализованная в виде DLL-библиотеки. Фактически математическая библиотека представляет собой набор классов: класс для расчета шлакового режима доменной плавки (класс «Solver») и классы – вспомогательные структуры для хранения исходных данных (например, класс «Component»). В классе «Solver» реализуется математическая модель шлакового режима. Класс «Solver» содержит в себе набор подклассов, которые представляют собой объекты реального мира, такие как кокс, шлак и т.д. Класс «Solver» является своего рода корневым классом, объединившим в себе все необходимые для расчета подклассы. Таким образом, классы математической библиотеки можно представить в виде следующей диаграммы (рис. 2).

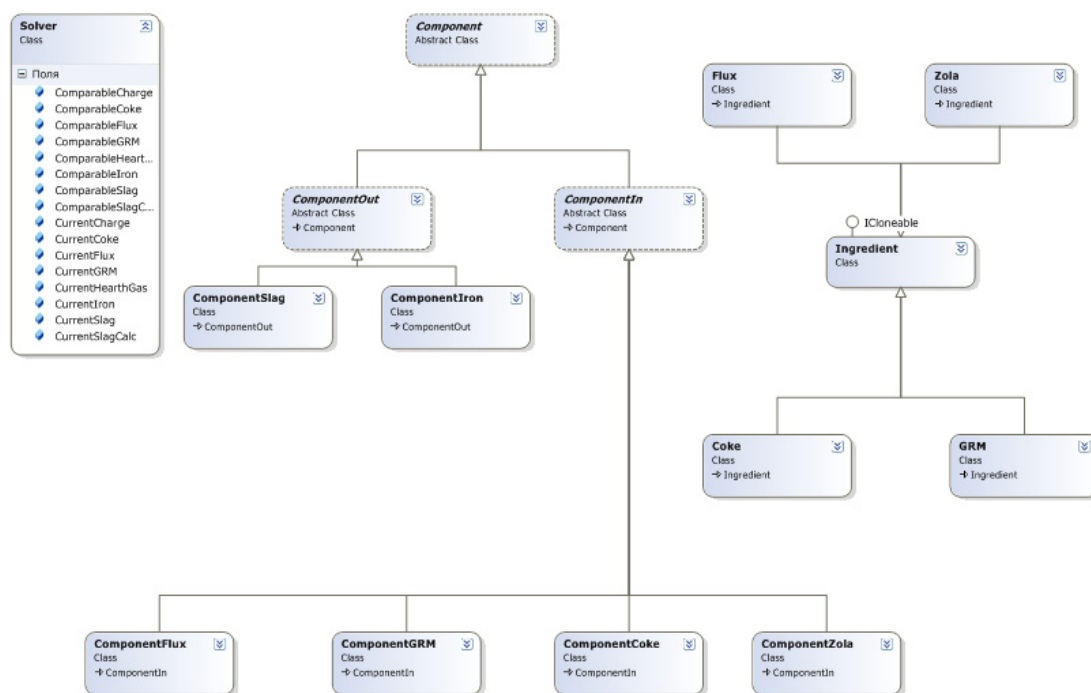


Рис. 2. Классы математической библиотеки

Реализация классов использует все основные принципы объектно-ориентированного программирования такие, как инкапсуляция, полиморфизм, наследование. В конечном итоге математическая библиотека была протестирована с помощью средств автоматизированного тестирования NUnit.

Клиентское приложение позволяет работать в прогнозном и сравнительном периодах, то есть рассчитывать показатели шлакового режима по исходным данным, введенным пользователем программы, и сравнивать результаты. Это позволяет моделировать «поведение» доменного шлака и наглядно сравнивать результаты.

В основе реализации прогнозного и сравнительного режима лежит набор классов и библиотек, относящихся к сборкам Reflection. Набор этих библиотек – это одна из ключевых возможностей .NET Framework, которая позволяет динамически получать информацию об объектах среды, а также использовать их [4]. Сборки Reflection служат для динамического формирования списка исходных данных. Например, механизмы Reflection используются для динамического построения списка исходных данных для сопоставления двух режимов. Класс математической модели представляется в виде набора свойств и соответствующим им атрибутам, что позволяет в дальнейшем построить таблицу с исходными данными.

При работе в сравнительном периоде существует возможность редактировать химические свойства исходных шихтовых материалов, а также добавлять новые материалы. Для решения этой задачи был создан настраиваемый элемент управления, что позволило автоматизировать и ускорить процесс разработки и избежать дополнительных ошибок.

Разработанная система полностью интегрирована в информационную структуру предприятия в процессе эксплуатации взаимодействует с другими программно-аппаратными информационными системами комбината. В этой связи обеспечение отдельных функций системы производится на основе автоматического выполнения задач по расписанию, например, функции сбора производственных данных и подготовки отчетных документов за истекший отчетный период.

Опыт эксплуатации системы показал, что ее применение в доменном производстве имеет несомненные преимущества и приносит положительный эффект от ее использования. Это обусловлено повышенными рисками от принятия необоснованных, неправильных управленческих решений с учетом относительно больших масштабов потребления сырьевых

и топливно-энергетических ресурсов, а также выпуска продукции доменного цеха. Инженерно-технологический персонал доменного цеха с помощью программного комплекса имеет возможность оперативно производить анализ работы шлакового режима, оценивать состояние хода технологического процесса, производить изучение и анализ наметившихся отклонений ключевых показателей, выявлять причины, повлекшие эти отклонения и разрабатывать мероприятия по повышению эффективности металлургического производства.

#### **Список использованных источников**

1. Компьютерные методы моделирования доменного процесса / О.П. Онорин, Н.А. Спирин, В.Л. Терентьев, Л.Ю. Гилева, В.Ю. Рыболовлев, И.Е. Косаченко, В.В. Лавров, А.В. Терентьев ; под ред. Н.А. Спирина. Екатеринбург: УГТУ-УПИ, 2005. 301 с.
2. Модельные системы поддержки принятия решений в АСУ ТП доменной плавки / Н.А. Спирин, В.В. Лавров, В.Ю. Рыболовлев, А.В. Краснобаев, О.П. Онорин, И.Е. Косаченко ; под ред. Н.А. Спирина. Екатеринбург: УрФУ, 2011. 462 с.
3. *Нильсен П.* SQL Server 2005. Библия пользователя. М.: Вильямс, 2008. 1232 с.
4. *Балена Ф. Димауро Дж.* Современная практика программирования на Microsoft Visual Basic и Visual C# / пер. с англ. М.: Русская редакция, 2006. 640 с.

### **БАЛАНСИРОВКА ВЫЧИСЛИТЕЛЬНОЙ НАГРУЗКИ В СОВРЕМЕННЫХ WEB-ПРИЛОЖЕНИЯХ**

© **К.А. Криницын, В.Ю. Носков, 2012**

*ФГАОУ ВПО «Уральский федеральный университет  
имени первого Президента России Б.Н. Ельцина», г. Екатеринбург*

#### **1. Необходимость балансировки**

Одним из важнейших характеристик веб-приложений является их доступность и скорость работы. По статистике Jupiter Communications (Юпитер Комуникэйшенс) 46 % пользователей web-систем прекращают посещение web-сервера из-за плохого времени отклика. BusinessWeek сообщает, что средний пользователь ожидает загрузки web-страницы на своем экране только 8 секунд, после чего покидает сервер в поисках более быстрого ресурса Amazon утверждает, что 100ms задержки стоят им 1 % продаж; Google подсчитал, что каждые дополнительные 0,5 с задержки приводят к падению трафика на 20 %. При этом простое наращивание мощности одной аппаратной платформы не приносит ожидаемого эффекта в связи с тем, что ресурсы любой аппаратной платформы являются ограниченными. Кроме этого, использование единой аппаратной платформы существенно снижает возможности по обеспечению отказоустойчивости и доступности решения. Построение же многоузловых систем требует решения задачи распределения нагрузки между отдельными узлами созданной системы. Простое равномерное нагрузки по узлам не может являться эффективным, поскольку требует применения одинаковых аппаратных платформ с одинаковыми наборами сервисов, что существенно усложняет масштабирование платформы, а также не учитывает текущую загрузку ресурсов и вычислительные возможности аппаратных платформ.

Проблему равномерного распределения нагрузки можно решить с помощью балансировки сетевой нагрузки, которая состоит из мониторинга состояния элементов сети и эффективное использование информации о состоянии каждого из элементов.

#### **2. Классы решений**

Классы решений, используемые при построении многоузловых систем.