

НЕЙРОСЕТЕВАЯ МОДЕЛЬ ТЕМПЕРАТУРНОГО РЕЖИМА ПЛАВКИ МЕДНО-НИКЕЛЕВОГО СЫРЬЯ В ПЕЧИ ВАНЮКОВА

Суляев И.И.,

научный руководитель канд. техн. наук Писарев А.И.
Норильский индустриальный институт

Для процесса варки черновой меди необходим медный штейн оптимальной температуры, который должен обеспечивать процесс плавки в печи Ванюкова (ПВ). В соответствии со стандартом предприятия температура штейна в ПВ должна поддерживаться в диапазоне 1050-1150 °С, шлака 1200-1250 °С.

Температура шлака ниже 1200 °С приводит к низкой растворимости магнетита и его выделению в самостоятельную фазу, происходит обогащение шлака цветными металлами (рис. 1).

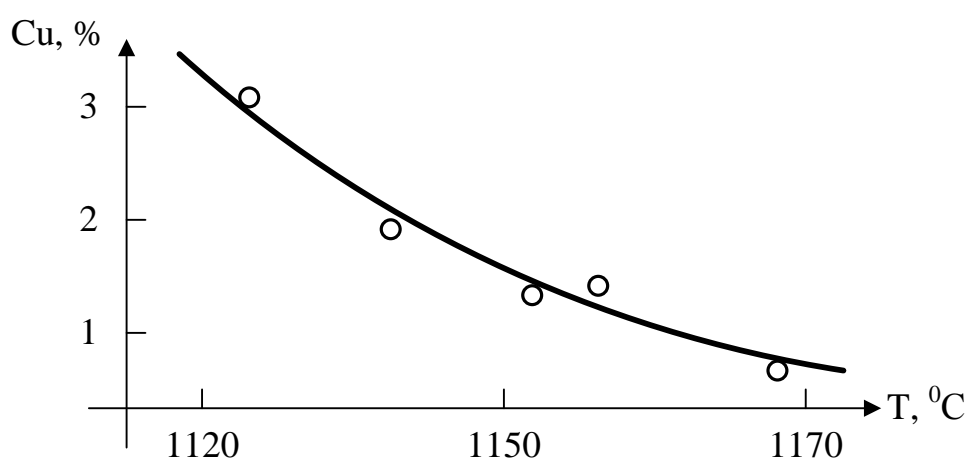


Рис. 1. График зависимости потерь меди со шлаком от температуры расплава в печи Ванюкова

Температура шлака выше 1250 °С вызывает ускоренный износ футеровки печи, перегрев отходящих газов. Таким образом, стабилизация температуры приведет к снижению потерь цветных металлов со шлаком.

В работе для построения нейросетевой модели температурного режима плавки используются статистические данные работы ПВ Медного завода ЗФ ОАО «ГМК «Норильский никель».

Созданию температурной модели расплава ПВ посвящено несколько работ, однако полученные математические модели не отличались высокой надежностью в условиях неполноты информации о технологическом процессе. Такие модели невозможно использовать в задаче оптимального управления печью.

Температура расплава в печи Ванюкова не измеряется в силу определенных причин (высокая температура, агрессивная среда и тем самым непродолжительный срок службы контрольно-измерительного прибора), поэтому фактором, косвенно характеризующим данный параметр, является температура воды на сливе с прикладных кессонов 1,2-го ряда горна печи, которая дает надежную информацию о тепловом режиме процесса, в частности, о температуре штейна.

В первом ряду кессонированного пояса печи расположено 22 водоохлаждаемых элемента, во втором – 4.

При разработке нейронной сети в первую очередь необходимо определить значимые входные факторы. В нашей модели только один выходной параметр – средняя температура воды на сливе с прикладных кессонов 1,2-го ряда правой стороны печи (рис. 3), косвенно характеризующая температуру штейна. Отметим, что разность давления охлаждающей воды на каждую сторону печи разное, поэтому целесообразно выбрать только одну из сторон. Для определения степени влияния входных факторов на выходную величину были проведены статистические исследования и выполнен корреляционный анализ.

График автокорреляционной функции $K_x(\tau)$ по температуре кессонов показал, что дискретность опроса температуры должна быть не более 5 минут. Следовательно, используемая нами частота опроса в 1 мин вполне приемлема.

Таким образом, входными факторами модели являются:

- $F_{\text{квс}}$ – расход кислородо-воздушной смеси (КВС) после смесителя;
- $P_{\text{квс}}$ – давление КВС после смесителя;
- Q_{O_2} – содержание кислорода в КВС;
- $F_{\text{пг}}$ – расход природного газа к зоне плавления;
- $F_{\text{ш}}$ – расход шихты;
- $T_{\text{в}}$ – температура воды до охлаждения;
- $P_{\text{в}}$ – давление воды на охлаждение кессонов правой стороны печи;
- $V_1 \dots V_8$ – скорости питателей под расходными бункерами.

Скорости питателей под бункерами характеризуют объемную производительность по каждому компоненту шихты. Связь между производительностью F и скоростью ленты V ленточного транспортера можно выразить формулой:

$$F_i = k_i \cdot V_i$$

где k_i – эмпирический коэффициент i -го питателя ($i = 1 \div 8$).

Как известно, для нейронных сетей весьма верно эмпирическое правило «мусор на входе – мусор на выходе», поэтому очень важна предварительная обработка (препроцессирование) исходных данных для обучения сети. Обучающий массив данных обозначим X_{ij} , где i – номер параметра, j – момент времени, x_{1j} – $F_{\text{квс}}$, x_{2j} – $P_{\text{квс}}$, x_{3j} – Q_{O_2} , x_{4j} – $F_{\text{пг}}$, x_{5j} – $F_{\text{ш}}$, x_{6j} – $T_{\text{в}}$, x_{7j} – $P_{\text{в}}$, x_{8j} – V_1 , x_{9j} – V_2 , x_{10j} – V_3 , x_{11j} – V_4 , x_{12j} – V_5 , x_{13j} – V_6 , x_{14j} – V_7 , x_{15j} – V_8 , x_{16j} – $T_{\text{к}}$, x_{17j} – $T_{\text{к-1}}$.

Массив разбиваем на три подмножества, из которых исключаем периоды, когда ПВ простаивала или показания контрольно-измерительных приборов были недостоверными (об этом сигнализирует бит состояния параметра в контроллере). Заполнение пропущенных значений осуществляется методом линейной интерполяции. Неизвестное значение x_j в момент времени t_j вычисляется следующим образом:

$$x_j = \frac{t_j(x_{j-1} - x_{j+1}) + (t_{j-1}x_{j-1} - t_{j+1}x_{j+1})}{t_{j-1} - t_{j+1}}$$

Размер обучающего набора соответствует одной неделе работы ПВ (10000 значений). Для снижения «нагрузки» на сеть данные подвергаем нормированию по формуле (1). Их приводим к интервалу $[0;1]$, который соответствует диапазону изменения логистической функции активации каждого нейрона сети. Нормирование исходных данных снижает нагрузку на сеть примерно на 20%, что позволяет улучшить соотношение «ошибка/время обучения».

$$\hat{x}_{ij} = \frac{x_{ij} - x_{i \min}}{x_{i \max} - x_{i \min}}, \quad (1)$$

где \hat{x}_{ij} – нормированное значение i -го параметра;

$x_{i \min} = \min_j \{x_{ij}\}$ – минимальное значение i -го параметра;

$x_{i\max} = \max_j \{x_{ij}\}$ - максимальное значение i -го параметра;

Далее каждый параметр подвергается фильтрации посредством соотношения (2). Задача фильтрации – исключение резких скачков параметра, выделение «полезного» сигнала для обучения нейросети.

$$y_{ij} = y_{i,j-1} \cdot K_n + \hat{x}_{ij} \cdot K_d, \quad (2)$$

где y_{ij} – отфильтрованное значение сигнала;

$y_{i,j-1}$ – отфильтрованное значение сигнала на предыдущем шаге;

\hat{x}_{ij} – фильтруемое значение сигнала;

K_n, K_d – коэффициенты ($K_n = 0,97, K_d = 0,03$).

Количество данных первого набора (обучающего) выбираем в пределах 50-70% от исходного. На его основании происходит подстройка синаптических весов нейронной сети. Второй набор – контрольный, на котором при поиске оптимальной модели выбирается наилучшая модель. Третий набор – тестовый. Итоговая модель проверяется на данных из этого множества, чтобы убедиться, что результаты, достигнутые на обучающем и контрольном множествах, реальны, а не являются артефактами процесса обучения.

Отметим, что алгоритма выбора структуры сети нет, поэтому подбор количества слоев сети и нейронов в них осуществлялся экспериментально. Основным критерием выбора сети было оптимальное соотношение между временем и ошибкой обучения на обучающем множестве (рис. 2).

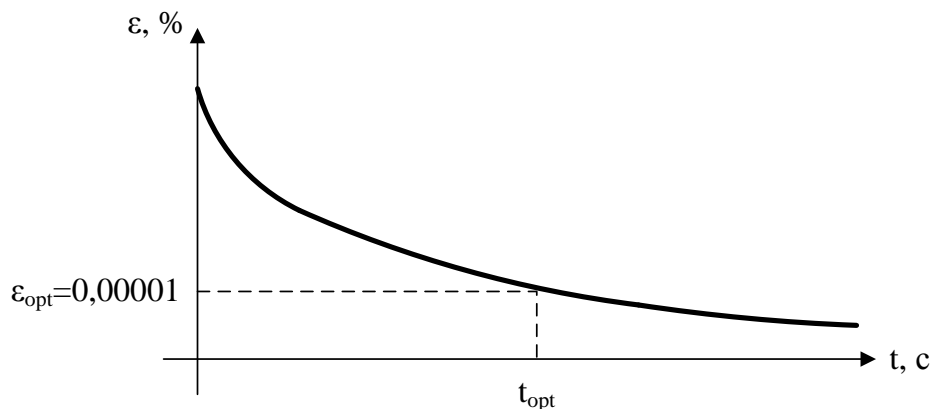


Рис. 2. График зависимости среднеквадратичной ошибки от скорости обучения нейронной сети

Как видно из графика, при дальнейшем увеличении числа итераций, среднеквадратичная ошибка уменьшается незначительно, а вероятность возникновения явления переобучения сети возрастает.

В результате поиска оптимальной модели лучшее соотношение показал перцептрон со следующей структурой: 17 входов, 17 нейронов в скрытом слое, 1 нейрон в выходном слое. Активационная функция всех нейронов сети – логистическая (сигмоидальная), описываемая формулой:

$$f(x) = \frac{1}{1 + e^{-\alpha \cdot x}},$$

где α – параметр наклона логистической функции ($\alpha = 1$).

В качестве алгоритма обучения сети выбран метод Левенберга-Марквардта, сутью которого является минимизация среднеквадратичной ошибки на обучающем множестве посредством подбора весовых коэффициентов и смещений у нейронов. Метод Левенберга-Марквардта (метод доверительных интервалов) – самый быстрый алгоритм обучения, но имеет ряд важных ограничений. Он применим только для сетей с одним выходным элементом, работает только со среднеквадратичной функцией ошибок и

требует памяти порядка W^2 (где W – количество синаптических весов у сети, поэтому для обучения больших сетей он неэффективен). Формула вычисления среднеквадратичной ошибки имеет вид:

$$\varepsilon = \frac{1}{N} \sum_{i=1}^N (T_{\text{э}i} - T_i)^2 \rightarrow \min$$

где $T_{\text{э}i}$ – экспериментальное значение температуры на i -м шаге;

T_i – рассчитанное значение температуры на i -м шаге;

N – объем обучающей выборки (10000 значений).

Нейронная сеть реализует функцию регрессии от 17 переменных:

$$T_{k+1} = f(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}).$$

Время обучения сети составило 20с при 43 эпохах и среднеквадратичной ошибке 10^{-5} . Результаты обучения сети отображены на рис. 3. Отметим совпадение теоретических и экспериментальных данных с заданной ошибкой в широком диапазоне изменения температуры (от 34°C до 40°C) прикладных кессонов 1,2 ряда горна печи Ванюкова-3.



Рис. 3. Результат обучения персептрона структуры 17-17-1

Структура нейромодели отображена на рис. 4. Математическая модель выполнена на базе архитектуры многослойного персептрона Розенблатта с одним скрытым слоем, в котором количество нейронов равно количеству входных факторов. Нейроны входного слоя сети не участвуют в вычислениях, а являются лишь повторителями сигнала. Весовые коэффициенты таких нейронов равны единице, а смещения нулю. Выходной слой содержит один нейрон, отвечающий за формирование выходного сигнала модели. Нейроны последующего слоя сети связаны с нейронами предыдущего по принципу «каждый с каждым». С целью «отслеживания» динамики изменения температуры на вход модели подается средняя температура кессонов на текущем и предыдущем шаге (T_k и T_{k-1}).

Для проверки степени адекватности модели на входы персептрона подаем нормированный массив данных за один месяц 2010 г. работы ПВ-3. Экспериментальные $T_{\text{э}i}$ и теоретические данные T_{pi} показаны на рис. 5. Рассчитанный коэффициент корреляции

$$R = \sqrt{\frac{\sum_{i=1}^N (T_{Pi} - \bar{T}_{\Theta})^2}{\sum_{i=1}^N (T_{\Theta i} - \bar{T}_{\Theta})^2}} \cdot 100\% \approx 99\%$$

показывает хорошее совпадение экспериментальных и теоретических данных.

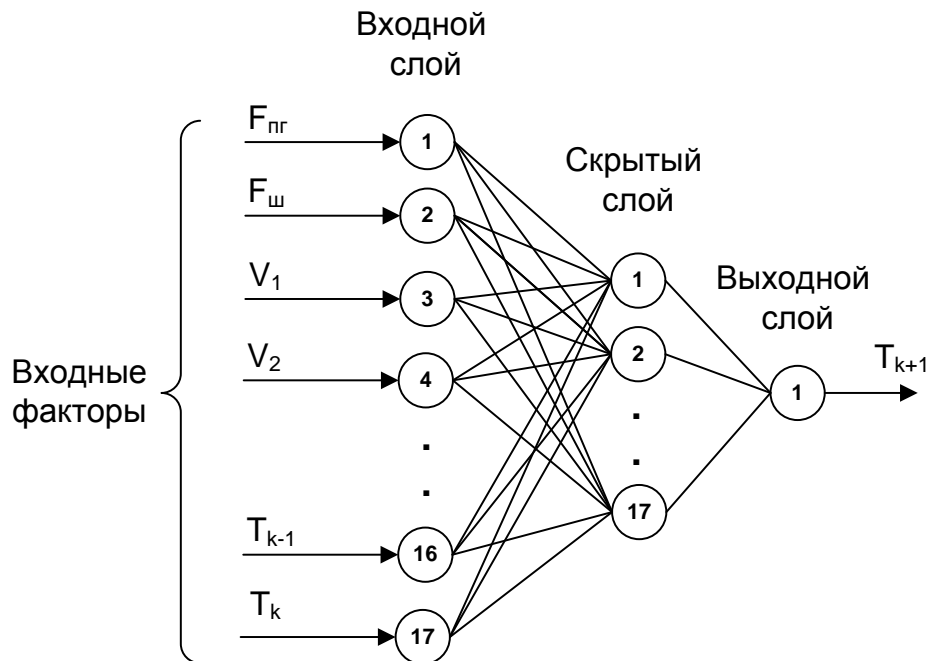


Рис. 4. Внутренняя структура нейромодели

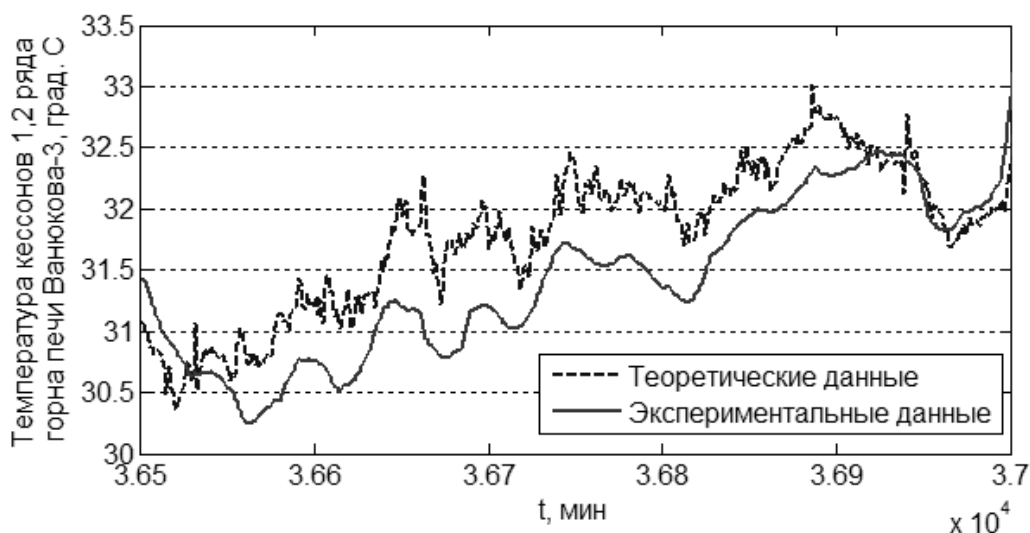


Рис. 5. Экспериментальные и теоретические данные средней температуры воды на сливе с прикладных кессонов 1,2-го ряда горна ПВ-3

Однако, для построения долгосрочного прогноза температуры (к примеру, на 30 мин вперед) необходимо учитывать динамику изменения каждого параметра в отдельности. При этом структура сети значительно усложняется, а время ее обучения увеличивается.