

ỨNG DỤNG MẠNG NƠON NHÂN TẠO ĐỂ DỰ BÁO PHỤ TẢI NGẮN HẠN TRONG HỆ THỐNG ĐIỆN CÓ XÉT ĐẾN ẢNH HƯỞNG CỦA THÔNG SỐ NHIỆT ĐỘ

APPLICATION OF ARTIFICIAL NEURAL NETWORK FOR SHORT-TERM LOAD FORECASTING IN POWER SYSTEMS CONSIDERING THE INFLUENCE OF TEMPERATURE PARAMETERS

Phạm Ngọc Hùng, Nguyễn Tùng Linh

Trường Đại học Điện lực

Tóm tắt: Công việc dự báo có ý nghĩa vô cùng quan trọng trong vận hành các hệ thống điện. Việc dự báo chính xác giúp đảm bảo tính ổn định của hệ thống điện (ổn định điện áp, ổn định tần số), bảo đảm sự cân bằng giữa điện năng sản xuất và điện năng tiêu thụ. Bài báo này đưa ra vấn đề ứng dụng thuật toán lan truyền ngược BPA (Back Propagation Algorithm) trong mạng Nơon nhân tạo để dự báo phụ tải ngắn hạn có xét đến ảnh hưởng của thông số nhiệt độ. Chương trình dự báo được viết trên ngôn ngữ Visual C.

Từ khóa: Dự báo, mạng nơon, nhiệt độ, ngắn hạn.

Abstract: Load forecasting is of great importance in power system operation. Accurate forecasting will help ensure the stability of power systems (voltage stability, frequency stability), balancing energy production and energy consumption. This paper addresses the application of Back Propagation Algorithm (BPA) in artificial neural network for short-term load forecast with consideration for the influence of temperature parameters. The forecast software is developed in Visual C.

Keywords: Forecast, neural network, temperature, short-term.

1. GIỚI THIỆU

Công tác dự báo luôn giữ vai trò quan

trọng trong nhiều lĩnh vực: quyết định kế hoạch sản xuất, hướng đầu tư phát triển trong tương lai, do đó có rất nhiều

các mô hình toán học áp dụng cho dự báo [1]. Cũng như các dự báo khác, dự báo phụ tải điện ngắn hạn cũng phải dựa vào số liệu thống kê, phân tích và áp dụng thuật toán để xác định mối quan hệ giữa phụ tải và các yếu tố ảnh hưởng, từ đó dự báo phụ tải dựa trên các yếu tố ảnh hưởng đó.

Qua nhiều công trình đã nghiên cứu [3, 5, 6] cho thấy rằng: những phương pháp dự báo truyền thống đã thể hiện nhiều nhược điểm: số liệu đầu vào nhiều, sai số lớn..., việc áp dụng mạng nơron nhân tạo cho dự báo phụ tải của hệ thống điện (HTĐ) dựa trên các yêu cầu: phương pháp đơn giản, có tính đến ảnh hưởng của biến nhiệt độ và đặc thù ngày, cho dự báo với sai số nhỏ là phương hướng của nghiên cứu này. Một trong những ưu điểm nổi bật của hệ thống nơron trong dự báo là phương pháp này không cần phải xác định những mối quan hệ giữa các biến số trước. Phương pháp này có thể xác định nhờ vào quá trình học hỏi về các mối quan hệ qua những thí dụ đã được đưa vào máy, không đòi hỏi bất kỳ giả định nào về các phân phối tổng thể và không giống những phương pháp dự báo truyền thống, nó có thể sử dụng mà không cần có đầy đủ số lượng các số liệu cần thiết. Chương trình hệ thống nơron có thể thay thế nhanh chóng mô hình hiện có, ví dụ như phân tích hồi quy, để đưa ra những dự báo chính xác mà không cần ngưng trệ các hoạt động đang diễn ra. Hệ thống nơron đặc biệt hữu ích khi số liệu đầu vào có tương quan cao hay có số lượng không đủ, hoặc khi hệ thống mang tính phi tuyến

cao. Phương pháp này cho kết quả dự báo có độ chính xác cao, dự báo được các sự kiện phụ thuộc thời gian.

2. TỔNG QUAN VỀ BÀI TOÁN DỰ TOÁN NGẮN HẠN

2.1. Khái quát

Người ta đã chứng tỏ rằng không có một phương pháp luận hoàn hảo trong tiếp cận các bài toán bằng cách sử dụng mạng nơron huấn luyện bởi thuật toán lan truyền ngược [2]. Ta có nhiều điều cần cân nhắc, lựa chọn để có thể thiết lập các tham số cho một mạng nơron, bao gồm:

- Số lớp ẩn, kích thước các lớp ẩn, hằng số học;
- Tham số bước đà (momentum);
- Khoảng, khuôn dạng dữ liệu đưa vào mạng;
- Dạng hàm nén (squashing);
- Khởi tạo ma trận trọng số;
- Tỷ lệ nhiễu ví dụ huấn luyện.

Việc dự báo dữ liệu (DL) là một bài toán rất phức tạp, cả về số lượng dữ liệu cần quan tâm cũng như độ chính xác của dữ liệu dự báo. Do vậy, việc cân nhắc để có thể chọn được mô hình phù hợp cho việc dự báo dữ liệu là một việc rất khó khăn (chỉ có thể bằng phương pháp thử - sai). Thuật toán lan truyền ngược (BPA) là thuật toán được ứng dụng rất rộng rãi trong các lĩnh vực: phân lớp, dự báo đã được thực tế chứng tỏ là một công cụ tốt áp dụng cho các bài toán trong các lĩnh vực dự báo dữ liệu.

Do đặc trưng về độ phức tạp dữ liệu, các dữ liệu đầu ra thường là các con số (mảng các số) đầu phẩy động nên việc lựa chọn cấu trúc mạng phù hợp thường dùng phương pháp thử sai. Đồng thời cần phải chuẩn hoá các dữ liệu đầu vào và đầu ra để mạng có khả năng học tốt hơn từ các dữ liệu được cung cấp. Trong việc dự báo dữ liệu, nếu dữ liệu ở nhiều khoảng thời gian khác nhau được đưa vào mạng để huấn luyện thì việc dự báo chính xác là rất khó nếu như mục đích là dự báo chính xác 100% dữ liệu trong tương lai, chỉ có thể có được kết quả dự báo với một mức độ chính xác chấp nhận được.

2.2. Thu thập, phân tích và xử lý dữ liệu

Dữ liệu đóng một vai trò rất quan trọng trong các giải pháp sử dụng mạng nơron. Chất lượng, độ tin cậy, tính sẵn có và phù hợp của dữ liệu được sử dụng để phát triển hệ thống giúp cho các giải pháp thành công. Các mô hình đơn giản cũng có thể đạt được kết quả nhất định nếu như dữ liệu được xử lý tốt, bộc lộ được các thông tin quan trọng. Bên cạnh đó, các mô hình tốt có thể sẽ không cho ta các kết quả mong muốn nếu dữ liệu đưa vào quá phức tạp và rắc rối. Việc xử lý dữ liệu được bắt đầu bằng việc: *thu thập và phân tích dữ liệu, sau đó là bước tiền xử lý. Dữ liệu sau khi qua bước tiền xử lý được đưa vào mạng nơron. Cuối cùng, dữ liệu đầu ra của mạng nơron qua bước hậu xử lý, bước này sẽ thực hiện biến đổi kết quả trả về của mạng nơron sang dạng yêu cầu của bài toán.*

a. Thu thập dữ liệu

Bước thu thập các dữ liệu bao gồm 3 nhiệm vụ chính:

Xác định yêu cầu dữ liệu

Điều đầu tiên cần thực hiện khi lập kế hoạch thu thập dữ liệu là xác định xem các dữ liệu nào là cần thiết để có thể giải quyết bài toán. Về tổng thể có thể cần sự trợ giúp của các chuyên gia trong lĩnh vực của bài toán cần giải quyết. Ta phải biết:

- Các dữ liệu chắc chắn có liên quan đến bài toán;
- Các dữ liệu nào có thể liên quan;
- Các dữ liệu nào là phụ trợ.

Các dữ liệu có liên quan và có thể liên quan đến bài toán cần được xem là các đầu vào cho hệ thống.

Xác định nguồn dữ liệu

Bước kế tiếp là quyết định nơi sẽ lấy dữ liệu, điều này cho phép ta xác định được các ước lượng thực tế về những khó khăn và kinh phí cho việc thu thập dữ liệu. Nếu ứng dụng yêu cầu các dữ liệu thời gian thực, những ước lượng này cần tính đến khả năng chuyển đổi các dữ liệu tương tự thành dạng số. Trong một số trường hợp, ta có thể chọn lựa dữ liệu mô phỏng từ các tình huống thực tế. Tuy nhiên, cần phải quan tâm đến độ chính xác và khả năng thể hiện của dữ liệu đối với các trường hợp cụ thể.

Xác định lượng dữ liệu

Ta cần phải ước đoán số lượng dữ liệu cần thiết để có thể sử dụng trong việc xây dựng mạng. Nếu lấy quá ít dữ liệu thì những dữ liệu này sẽ không thể

phản ánh toàn bộ các thuộc tính mà mạng cần phải học và do đó mạng sẽ không có được phản ứng mong đợi đối với những dữ liệu mà nó chưa được huấn luyện.

Mặt khác, cũng không nên đưa vào huấn luyện cho mạng quá nhiều dữ liệu. Về tổng thể lượng dữ liệu cần thiết bị chi phối bởi số các trường hợp cần luyện cho mạng. Bản chất đa chiều của dữ liệu và cách giải quyết mong muốn là các nhân tố chính xác định số các trường hợp cần luyện cho mạng và kéo theo là lượng dữ liệu cần thiết. Việc định lượng gần đúng lượng dữ liệu cần đưa vào luyện mạng là rất cần thiết. Thông thường dữ liệu thường thiếu hoàn chỉnh, do đó nếu muốn mạng có khả năng thực hiện được những điều mà ta mong đợi thì nó cần phải được luyện với lượng dữ liệu lớn hơn. Đương nhiên, nếu có được độ chính xác và hoàn chỉnh của dữ liệu thì số các trường hợp cần thiết phải đưa vào mạng có thể giảm đi.

b. Phân tích dữ liệu

Có hai kỹ thuật cơ bản để có thể hiểu được dữ liệu:

Phân tích thống kê

Mạng nơron có thể được xem như là một mở rộng của phương pháp thống kê chuẩn. Các thử nghiệm có thể cho ta biết được khả năng mà mạng có thể thực hiện. Hơn nữa, phân tích có thể cho ta các đầu mối để xác định các đặc trưng, ví dụ nếu dữ liệu được chia thành các lớp, các thử nghiệm thống kê có thể xác định được khả năng phân biệt các lớp trong dữ liệu thô hoặc dữ liệu đã qua tiền xử lý.

Trực quan hoá dữ liệu

Trực quan hoá dữ liệu bằng cách vẽ biểu đồ trên các dữ liệu theo một dạng thích hợp sẽ cho ta thấy được các đặc trưng phân biệt của dữ liệu, chẳng hạn như: các điểm lệch hay các điểm đỉnh. Điều này nếu thực hiện được có thể áp dụng thêm các thao tác tiền xử lý để tăng cường các đặc trưng đó. Thông thường, phân tích dữ liệu bao gồm cả các kiểm tra thống kê và trực quan hoá. Các kiểm tra này sẽ được lặp đi lặp lại. Trực quan hoá cho ta sự đánh giá về dữ liệu và các khái niệm sơ khởi về các mẫu nằm sau dữ liệu. Trong khi các phương pháp thống kê cho phép ta kiểm thử những khái niệm này.

c. Xử lý dữ liệu

Dẫn nhập về xử lý dữ liệu

Khi những dữ liệu thô đã được thu thập, chúng cần phải được chuyển đổi sang các khuôn dạng phù hợp để có thể đưa vào luyện mạng. Ở bước này ta cần thực hiện các công việc sau:

- Kiểm tra tính hợp lệ dữ liệu (Data validity checks);
- Việc kiểm tra tính hợp lệ sẽ phát hiện ra các dữ liệu không thể chấp nhận được mà nếu sử dụng chúng thì sẽ cho ra các kết quả không tốt. Chẳng hạn, có thể kiểm tra khoảng hợp lệ của dữ liệu về nhiệt độ không khí của một vùng nhiệt đới; ta mong muốn các giá trị trong khoảng từ 8°C đến 40°C, các giá trị nằm ngoài khoảng này sẽ không chấp nhận.

Nếu có một mẫu cho một phân bố sai của dữ liệu thì ta cần xem xét nguyên nhân của nó. Dựa trên bản chất của

nguyên nhân dẫn đến sai lầm, ta có thể hoặc phải loại bỏ các dữ liệu này, hoặc cho phép những thiếu sót đó. Nếu có các thành phần quyết định không mong muốn như là các xu hướng hay các biến thiên có tính chất mùa vụ, chúng cần được loại bỏ ngay.

Phân hoạch dữ liệu (partitioning data)

Phân hoạch là quá trình chia tập dữ liệu thành các tập kiểm định, huấn luyện và kiểm tra. Tập kiểm định dùng để xác định kiến trúc của mạng, tập huấn luyện dùng để điều chỉnh trọng số của mạng, tập kiểm tra dùng để kiểm tra hiệu năng của mạng sau khi huấn luyện. Ta cần phải đảm bảo rằng:

- Tập ví dụ huấn luyện chứa đủ dữ liệu, các dữ liệu đó phân bố phù hợp sao cho có thể biểu diễn các thuộc tính mà ta muốn mạng sẽ học được.
- Không có dữ liệu trùng nhau hay tương tự nhau của các dữ liệu trong các tập dữ liệu khác nhau.

Tiền xử lý

Về mặt lý thuyết, một mạng nơron có thể dùng để ánh xạ các dữ liệu thô đầu vào trực tiếp thành các dữ liệu đầu ra. Nhưng trong thực tế, việc sử dụng quá trình tiền xử lý cho dữ liệu thường mang lại những hiệu quả nhất định trước khi những dữ liệu này được đưa vào mạng. Có rất nhiều kỹ thuật liên quan đến tiền xử lý dữ liệu. Tiền xử lý dữ liệu có thể là thực hiện lọc dữ liệu hay các phương pháp phức tạp hơn như: các phương pháp kết xuất, trích chọn các đặc trưng từ dữ liệu ảnh tĩnh. Bởi lẽ việc chọn thuật toán dùng trong tiền xử lý dữ liệu là phụ thuộc vào ứng

dụng và bản chất của dữ liệu nên các khả năng lựa chọn là rất lớn. Tuy nhiên, mục đích của các thuật toán tiền xử lý dữ liệu thường tương tự nhau [6]:

- Chuyển đổi dữ liệu về khuôn dạng phù hợp đối với đầu vào mạng nơron, điều này thường đơn giản hoá quá trình xử lý của mạng phải thực hiện trong thời gian ngắn hơn. Các chuyển đổi này có thể gồm:

- Áp dụng một hàm toán học cho đầu vào. Mã hoá các dữ liệu văn bản trong cơ sở dữ liệu;

- Chuyển đổi dữ liệu sao cho nó có giá trị nằm trong khoảng $[0,1]$;

- Lựa chọn các dữ liệu xác đáng nhất, việc lựa chọn này có thể bao gồm các thao tác đơn giản như lọc hay lấy tổ hợp của các đầu vào để tối ưu hoá nội dung của dữ liệu. Điều này đặc biệt quan trọng khi mà dữ liệu có nhiều hoặc chứa các thông tin thừa. Việc lựa chọn cẩn thận các dữ liệu phù hợp sẽ làm cho mạng dễ xây dựng và tăng cường hiệu năng của chúng đối với các dữ liệu nhiễu;

- Tối thiểu hoá số các đầu vào mạng, giảm số chiều của dữ liệu đầu vào và tối thiểu số các mẫu đưa vào có thể đơn giản hoá được bài toán. Trong một số trường hợp ta không thể nào đưa tất cả các dữ liệu vào mạng.

Hậu xử lý

Hậu xử lý bao gồm các xử lý áp dụng cho đầu ra của mạng. Cũng như đối với tiền xử lý, hậu xử lý hoàn toàn phụ thuộc vào các ứng dụng cụ thể và có thể bao gồm cả việc phát hiện các tham số có giá trị vượt quá khoảng cho phép

hoặc sử dụng đầu ra của mạng như một đầu vào của một hệ khác, chẳng hạn như một bộ xử lý dựa trên luật. Đôi khi hậu xử lý chỉ đơn giản là quá trình ngược lại đối với quá trình tiền xử lý.

3. BÀI TOÁN DỰ BÁO ĐỈNH VÀ ĐÁY BIỂU ĐỒ PHỤ TẢI

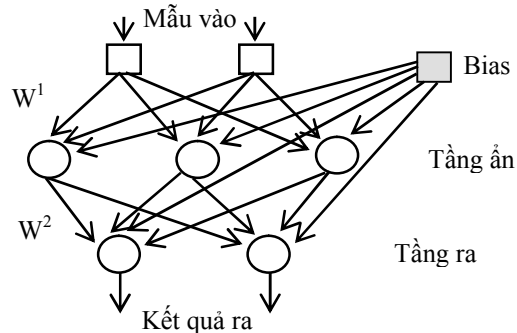
Phần này mô tả thuật toán học sử dụng để điều chỉnh hiệu năng của mạng sao cho mạng có khả năng sinh ra được các kết quả mong muốn. Các mạng nơron truyền thẳng nhiều lớp được huấn luyện bằng phương pháp học có giáo viên hướng dẫn. Phương pháp này căn bản dựa trên việc yêu cầu mạng thực hiện chức năng của nó và sau đó trả lại kết quả, kết hợp kết quả này với các đầu ra mong muốn để điều chỉnh các tham số của mạng. Về cơ bản, BPA là dạng tổng quát của thuật toán trung bình bình phương tối thiểu (Least Means Square-LMS) bởi lẽ nó cũng sử dụng kỹ thuật giảm theo hướng vector gradient nhưng với độ phức tạp của hàm lỗi lớn hơn. Thuật toán này thuộc dạng thuật toán xấp xỉ để tìm các điểm mà tại đó hiệu năng của mạng là tối ưu. Chỉ số tối ưu thường được xác định bởi một hàm số của ma trận trọng số và các đầu vào nào đó bất kỳ.

3.1. Mô tả thuật toán

Ta sẽ sử dụng dạng tổng quát của mạng nơron truyền thẳng nhiều lớp như ở hình 1. Khi đó đầu ra của một lớp trở thành đầu vào của lớp kế tiếp [3]:

$$a^{m+1} = f^{m+1}(W^{m+1}a^m + b^{m+1}); m = 0, 1, \dots, M-1 \quad (1)$$

ở đây M là số lớp trong mạng, các nơron lớp thứ nhất nhận tín hiệu từ bên ngoài: $a^0 = p$; đầu ra của lớp cuối cùng là đầu ra của mạng: $a = a^m$.



Hình 1. Kiến trúc mạng

Chỉ số hiệu năng (performance index)

Thuật toán BPA sử dụng chỉ số hiệu năng là trung bình bình phương lỗi của đầu ra so với giá trị đích. Đầu vào của mạng là tập ví dụ huấn luyện:

$$\{(p_1, t_1), (p_2, t_2), \dots, (p_q, t_q)\} \quad (2)$$

ở đây p_i là đầu vào và t_i là đầu ra đích tương ứng (với $i = 1, 2, \dots, q$). Mỗi đầu vào đưa vào mạng, sau khi tính toán cho đầu ra, đầu ra này được đem so sánh với đầu ra mong muốn. Thuật toán sẽ điều chỉnh các tham số của mạng để tối thiểu hoá trung bình bình phương lỗi [4]:

$$F(x) = E|e^2| = E|(t - a)^2| \quad (3)$$

ở đây x là biến được tạo thành bởi các trọng số và độ lệch, E là ký hiệu kỳ vọng toán học. Nếu như mạng có nhiều đầu ra, ta có thể viết lại phương trình trên ở dạng ma trận như sau:

$$F(x) = E|e^T e| = E|(t - a)^T (t - a)| \quad (4)$$

Ký hiệu $\hat{F}(x)$ là giá trị xấp xỉ của $F(x)$ thì ta có xấp xỉ của trung bình bình phương lỗi như sau:

$$\hat{F}(x) = (t(k) - a(k))^T (t(k) - a(k)) = e^T(k)e(k) \quad (5)$$

trong đó kỳ vọng toán học của bình phương lỗi được thay bởi bình phương lỗi tại bước thứ k .

Thuật toán giảm theo hướng cho trung bình bình phương lỗi xấp xỉ (với α là hệ số học) là [5]:

$$\begin{cases} w_{ji}^m(k+1) = w_{ji}^m(k) - \alpha \frac{\partial \hat{F}}{\partial w_{ji}^m} \\ b_j^m(k+1) = b_j^m(k) - \alpha \frac{\partial \hat{F}}{\partial b_j^m} \end{cases} \quad (6)$$

Luật xích (Chain Rule)

Đối với mạng nơron truyền thẳng nhiều lớp, lỗi không chỉ là một hàm của chỉ các trọng số trong lớp ẩn, do vậy việc tính các đạo hàm từng phần này là không đơn giản. Chính vì lý do đó mà ta phải sử dụng luật xích để tính. Luật này được mô tả như sau: Giả sử ta có f là hàm của biến n , ta muốn đạo hàm của f có liên quan đến một biến w khác, luật xích này như sau [6]:

$$\frac{df(n(w))}{dw} = \frac{df(n)}{dn} \cdot \frac{dn(w)}{dw} \quad (7)$$

Vậy đạo hàm trong (6) sẽ là:

$$\begin{cases} \frac{\partial \hat{F}}{\partial w_{ji}^m} = \frac{\partial \hat{F}}{\partial n_j^m} \cdot \frac{\partial n_j^m}{\partial w_{ji}^m} \\ \frac{\partial \hat{F}}{\partial b_j^m} = \frac{\partial \hat{F}}{\partial n_j^m} \cdot \frac{\partial n_j^m}{\partial b_j^m} \end{cases} \quad (8)$$

trong đó hạng thức thứ hai ở vế phải của phương trình (8) được tính một cách dễ dàng, bởi vì đầu vào của lớp m của mạng là một hàm của trọng số và độ lệch trong tầng đó:

$$n_j^m = \sum_{i=1}^{s^{m-1}} w_{ji}^m a_i^{m-1} + b_j^m \quad (9)$$

do vậy $\frac{\partial n_j^m}{\partial w_{ji}^m} = a_i^{m-1}$; $\frac{\partial n_j^m}{\partial b_j^m} = 1$

Nếu gọi $s_j^m = \frac{\partial \hat{F}}{\partial n_j^m}$ là độ nhạy cảm

của \hat{F} đối với các thay đổi của phần tử thứ j của đầu vào của mạng tại lớp thứ m . Khi đó ta có:

$$\begin{cases} \frac{\partial \hat{F}}{\partial w_{ji}^m} = \frac{\partial \hat{F}}{\partial n_j^m} \cdot \frac{\partial n_j^m}{\partial w_{ji}^m} = s_j^m a_i^{m-1} \\ \frac{\partial \hat{F}}{\partial b_j^m} = \frac{\partial \hat{F}}{\partial n_j^m} \cdot \frac{\partial n_j^m}{\partial b_j^m} = s_j^m \end{cases} \quad (10)$$

Thuật toán giảm theo hướng biểu diễn như sau:

$$\begin{cases} w_{ji}^m(k+1) = w_{ji}^m(k) - \alpha s_j^m a_i^{m-1} \\ b_j^m(k+1) = b_j^m(k) - \alpha s_j^m \end{cases} \quad (11)$$

Lan truyền ngược độ nhạy cảm

Bây giờ ta cần tính toán nốt ma trận độ nhạy cảm s^m . Để thực hiện điều này cần sử dụng một áp dụng khác của luật xích. Quá trình này cho ta khái niệm về sự “Lan truyền ngược” bởi vì nó mô tả mối quan hệ hồi qui trong đó độ nhạy cảm s^m được tính qua độ nhạy cảm s^{m+1} của lớp $m+1$. Để dẫn đến quan hệ đó, ta sử dụng ma trận Jacobian:

$$\frac{\partial n^{m+1}}{\partial n^m} = \begin{bmatrix} \frac{\partial n_1^{m+1}}{\partial n_1^m} & \frac{\partial n_1^{m+1}}{\partial n_2^m} & \dots & \frac{\partial n_1^{m+1}}{\partial n_{s^m}^m} \\ \frac{\partial n_2^{m+1}}{\partial n_1^m} & \frac{\partial n_2^{m+1}}{\partial n_2^m} & \dots & \frac{\partial n_2^{m+1}}{\partial n_{s^m}^m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial n_{s^{m+1}}^{m+1}}{\partial n_1^m} & \frac{\partial n_{s^{m+1}}^{m+1}}{\partial n_2^m} & \dots & \frac{\partial n_{s^{m+1}}^{m+1}}{\partial n_{s^m}^m} \end{bmatrix} \quad (12)$$

Xét thành phần thứ j, i của ma trận trên:

$$\begin{aligned} \frac{\partial n_j^{m+1}}{\partial n_i^m} &= \frac{\partial \left(\sum_{j=1}^{s^m} w_{ji}^{m+1} a_j^m + b_j^{m+1} \right)}{\partial n_i^m} \\ &= w_{ji}^{m+1} \frac{\partial f^m(n_i^m)}{\partial n_i^m} \\ &= w_{ji}^{m+1} f^m(n_i^m) = w_{ji}^{m+1} \cdot \frac{\partial f^m(n_i^m)}{\partial n_i^m} \end{aligned} \quad (13)$$

Như vậy ma trận Jacobian có thể viết lại như sau:

$$\frac{\partial n^{m+1}}{\partial n^m} = W^{m+1} \dot{F}^m(n^m) \quad (14)$$

ở đây:

$$\dot{F}^m(n^m) = \begin{bmatrix} f^m(n_1^m) & 0 & \dots & 0 \\ 0 & f^m(n_2^m) & & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & & f^m(n_{s^m}^m) \end{bmatrix}$$

Bây giờ ta viết lại quan hệ hồi qui cho độ nhạy cảm dưới dạng ma trận:

$$\begin{aligned} s^m &= \frac{\partial \hat{F}}{\partial n^m} = \left(\frac{\partial n^{m+1}}{\partial n^m} \right)^T \frac{\partial \hat{F}}{\partial n^{m+1}} \\ &= \dot{F}^m(n^m) (W^{m+1})^T s^{m+1} \end{aligned} \quad (15)$$

Bây giờ ta có thể thấy độ nhạy cảm được lan truyền ngược qua mạng từ lớp cuối cùng về lớp đầu tiên:

$$s^M \rightarrow s^{M-1} \rightarrow \dots \rightarrow s^2 \rightarrow s^1$$

Cần nhấn mạnh rằng ở đây thuật toán BPA sử dụng cùng một kỹ thuật giảm theo hướng. Sự phức tạp duy nhất là ở chỗ để tính gradient ta cần phải lan truyền ngược độ nhạy cảm từ các lớp về các lớp trước. Bây giờ ta cần biết điểm bắt đầu lan truyền ngược, xét độ nhạy cảm s^m tại lớp cuối cùng:

$$\begin{aligned} s_j^m &= \frac{\partial \hat{F}}{\partial n_j^m} = \frac{\partial (t-a)^T (t-a)}{\partial n_j^m} \\ &= \frac{\partial \sum_{i=1}^{s^M} (t_i - a_i)^2}{\partial n_j^M} = -2(t_j - a_j) \frac{\partial a_j}{\partial n_j^M} \end{aligned} \quad (16)$$

$$\text{do } \frac{\partial a_j}{\partial n_j^M} = \frac{\partial a_j^M}{\partial n_j^M} = \frac{\partial f^M(n_j^M)}{\partial n_j^M} = f^M(n_j^M)$$

nên ta có thể viết dưới dạng ma trận như sau:

$$s^M = -2 \dot{F}^M(n^M) (t-a) \quad (17)$$

Tóm lại, thuật toán lan truyền ngược có thể biểu diễn như sau:

Bước 1: Lan truyền xuôi đầu vào qua mạng

$$\begin{aligned} a^0 &= p \\ a^{m+1} &= f^{m+1}(W^{m+1} a^m + b^{m+1}) \\ a &= a^M \end{aligned}$$

với $m = 0, 1, \dots, M-1$.

Bước 2: Lan truyền độ nhạy cảm (lỗi) ngược lại qua mạng:

$$s^M = -2 \dot{F}^M (n^M)(t-a)$$

$$s^m = \dot{F}^m (n^m)(W^{m+1})^T s^{m+1}$$

với $m = M - 1, M - 2, \dots, 1$

Bước 3: Các trọng số và độ lệch được cập nhật bởi công thức:

$$W^m(k+1) = W^m(k) - \alpha s^m (a^{m-1})^T$$

$$b^m(k+1) = b^m(k) - \alpha s^m$$

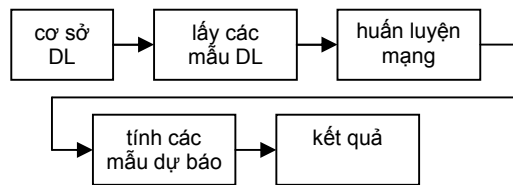
3.2. Cấu trúc mạng nơon dự báo đỉnh/đáy biểu đồ phụ tải HTĐ

Có các số liệu về nhiệt độ cao nhất T_{max}/T_{min} và phụ tải cao nhất P_{max}/P_{min} tương ứng trong ngày của các ngày làm việc trong tháng, ta có thể xây dựng tập hợp ví dụ huấn luyện dùng để huấn luyện và kiểm tra khả năng dự báo của mạng. Một ví dụ huấn luyện cụ thể trong tập ví dụ huấn luyện nói trên được xây dựng từ các giá trị vào:

- T_{max}/T_{min} của ngày cần dự báo;
- T_{max}/T_{min} của ngày trước ngày cần dự báo;
- T_{max}/T_{min} của ba ngày có cùng kiểu ngày trước đó;
- P_{max}/P_{min} của ba ngày có cùng kiểu ngày trước đó và một giá trị đầu ra, P_{max}/P_{min} của ngày cần dự báo.

Số liệu đầu vào và ra của các ngày khác được xây dựng theo phương pháp tương tự. Như vậy có dữ liệu về nhiệt độ cao nhất/thấp nhất trong ngày và phụ tải cao nhất/thấp nhất tương ứng của ngày đó có thể xây dựng tập ví dụ

huấn luyện cho bất kỳ giai đoạn nào. Sơ đồ các bước thực hiện cho ở hình 2.



Hình 2. Sơ đồ khối của chương trình

4. KẾT QUẢ XÂY DỰNG CHƯƠNG TRÌNH DỰ BÁO ĐỒ THỊ PHỤ TẢI NGẮN HẠN TRONG HTĐ

4.1. Dữ liệu dự báo phụ tải

Các thông số phụ tải mẫu dùng để huấn luyện mạng nơon là P_{max} , P_{min} và nhiệt độ T_{max} , T_{min} tương ứng cùng ngày trong vài tuần trước (hình 3). Tốc độ huấn luyện được chọn giá trị ban đầu trong khoảng 0.3 đến 0.5, hệ số quan tính được chọn giá trị ban đầu là 0.5. Trong nghiên cứu này lựa chọn số liệu phụ tải P_{max} và nhiệt độ T_{max} trong ngày của các ngày làm việc tháng 6/2003 của HTĐ Miền Bắc [7] làm các ví dụ để học, dự báo và kiểm chứng sai số.

DuLieuPhuTai	
Ngày	
PhuTai_Max	
NhietDo_Max	
PhuTai_Min	
NhietDo_Min	

Hình 3. Biểu đồ cơ sở dữ liệu

Ta sẽ chọn số liệu phụ tải P_{max} và nhiệt độ T_{max} trong ngày của các ngày làm việc tháng 6/2009 làm các ví dụ để học và dự báo. Bảng 1 là tập hợp các dữ liệu phụ tải cao điểm, phụ tải thấp

điểm, nhiệt độ cao và nhiệt độ thấp trong ngày của tháng 6/2009 khu vực miền Bắc nước ta [8].

Bảng 1. Dữ liệu phụ tải và nhiệt độ

Thứ	Ngày	Dữ liệu dự báo đỉnh		Dữ liệu dự báo đáy	
		P_{cao} điểm (MW)	T_{max} ($^{\circ}C$)	$P_{thấp}$ điểm (MW)	T_{min} ($^{\circ}C$)
2	9/6/2003	6588.5	36	3997.5	26
3	10/6/2003	6581.5	36	4316.5	26
4	11/6/2003	6454.9	35	4124.5	26
5	12/6/2003	6450.2	32	3837.5	25
6	13/6/2003	6587.1	33	3858.5	25
7	14/6/2003	6245.5	35	3658.5	26
CN	15/6/2003	5896.5	34	3842	26
2	16/6/2003	6412.9	35	3760.5	25
3	17/6/2003	6431.4	35	3890	25
4	18/6/2003	6506.5	35	3969	26
5	19/6/2003	6547.8	35	4149.5	27
6	20/6/2003	6592.2	33	4063.5	25
7	21/6/2003	6237.3	33	3850	26
CN	22/6/2003	5743.5	34	3618	26
2	23/6/2003	6396.7	33	3452	26
3	24/6/2003	6404.8	32	3829.5	25
4	25/6/2003	6391.6	35	3814	25
5	26/6/2003	6483.1	35	3956.5	26
6	27/6/2003	6284.8	35	3995.5	27
7	28/6/2003	5958.3	36	3781	27
CN	29/6/2003	5551.3	36	3634	26
2	30/6/2003	6050.8	35	3289	25

4.2. Kết quả

Dựa vào các số liệu về nhiệt độ cao nhất/thấp nhất trong ngày và P_{max}/P_{min} tương ứng của các ngày làm việc trong tháng, ta xây dựng tập hợp mẫu dùng để huấn luyện và kiểm tra khả năng dự báo của mạng. Như vậy có dữ liệu về nhiệt độ cao nhất trong ngày và phụ tải cao nhất tương ứng của ngày đó ta có thể xây dựng tập ví dụ huấn luyện cho bất kỳ giai đoạn nào trong năm. Kết

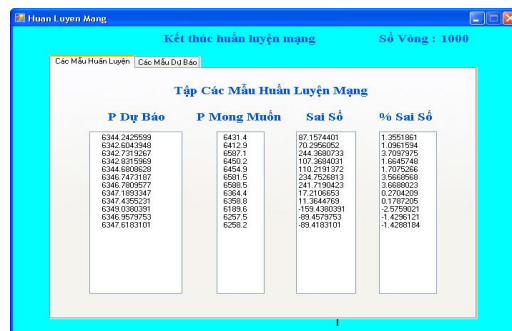
quả huấn luyện mạng và tính toán dự báo đồ thị phụ tải cho ngày 23/06/2009 được cho trong hình 4, 5, 6.



Hình 4. Giao diện form dự báo phụ tải ngày 23/06/2009

Chức năng của form này là:

- Lấy các mẫu dữ liệu để huấn luyện;
- Chọn ngày cần dự báo;
- Chọn số vòng và hệ số lỗi khi huấn luyện;
- Dự báo các đỉnh và đáy của phụ tải.



Hình 5. Giao diện form kết quả huấn luyện mạng

Khối trong hình 5 có chức năng khởi tạo mạng neuron nhân tạo, tiến hành quá trình huấn luyện mạng, đưa ra các kết quả mẫu huấn luyện, điều chỉnh trọng số của tập huấn luyện sao cho mẫu huấn luyện có sai số và % sai số là nhỏ nhất. Khi kết thúc huấn luyện sẽ đưa ra

kết quả của mẫu dự báo, sai số và % sai số.

Tập Các Mẫu Dự Báo			
P Dự Báo	P Mong Muốn	Sai Số	% Sai Số
6439.0656959	6391.6	47.4656959	0.7426256

Hình 6. Giao diện form kết thúc huấn luyện mạng

Từ kết quả nhận được như trong các hình 5 và hình 6 cho thấy rằng: phương pháp mạng nơron cho kết quả tốt hơn so với các phương pháp truyền thống. Cụ thể đối với tập số liệu này sai số của dự báo bằng mạng nơron là 0,74%, trong khi đó sai số của hàm đa thức bậc 2 và hàm mũ là 11,5% [8].

4.3. Nhận xét

Hiện có nhiều bài toán dự báo phụ tải điện ứng dụng thuật toán lan truyền ngược BPA. Tuy nhiên, so với phương pháp truyền thống (tức xử lý báo động bằng hệ chuyên gia hoặc do chính nhân viên vận hành thực hiện) thì phương pháp này cho phép tiết kiệm thời gian rất nhiều, bởi vì xử lý kiến thức do con người hoặc bằng hệ thống chuyên gia đều chậm hơn phân loại mẫu bằng mạng nơron truyền thẳng. So với hệ chuyên gia, mạng nơron có khả năng làm việc tốt hơn với các mẫu tín hiệu không đầy đủ, biến dạng hoặc gián đoạn.

Ngoài ra, mạng nơron trình bày trong bài báo này đã xét đến ảnh hưởng của

hiệt độ và yếu tố mùa trong năm (bảng 1). Chính vì vậy mà mô hình, cấu trúc mạng nơron đề xuất đã thể hiện được tính ưu việt của nó về độ chính xác. Phương pháp này không yêu cầu quá nhiều lượng thông tin đầu vào so với các mạng nơron truyền thẳng cho dự báo phụ tải đã được công bố [5, 9, 10], hoàn toàn phù hợp với công tác điều độ, vận hành hệ thống điện thực ở Việt Nam.

5. KẾT LUẬN

Việc dự báo dữ liệu là một bài toán rất phức tạp, cả về số lượng dữ liệu cần quan tâm cũng như độ chính xác của dữ liệu dự báo. Do vậy, việc cân nhắc để có thể chọn được mô hình phù hợp cho việc dự báo dữ liệu là một việc rất khó khăn (chỉ có thể bằng phương pháp thử - sai). Thuật toán lan truyền ngược ứng dụng trong dự báo đã được thực tế chứng tỏ là một công cụ tốt áp dụng cho các bài toán trong các lĩnh vực dự báo dữ liệu.

Có các số liệu về nhiệt độ cao nhất T_{max} và phụ tải cao nhất P_{max} tương ứng trong ngày của các ngày làm việc trong tháng, ta có thể xây dựng tập hợp ví dụ huấn luyện dùng để huấn luyện và kiểm tra khả năng dự báo của mạng.

Phương pháp BPA học rất nhanh, mạng học mối quan hệ giữa công suất phụ tải, thời gian ngày - tuần, và điều kiện nhiệt độ thông qua dữ liệu vectơ tham số học. Vì vậy quy trình dự báo bằng mạng nơron sử dụng thuật toán lan truyền ngược sai số hứa hẹn mang lại nhiều kết quả khả quan trong vấn đề dự báo phụ tải ngắn hạn.

TÀI LIỆU THAM KHẢO

- [1] Đặng Ngọc Dinh, Trần Bách, Trịnh Hùng Thám; Hệ thống điện; Tập I, II. Nhà xuất bản Đại học và Trung học chuyên nghiệp, 1976.
- [2] Lê Minh Trung; Mạng nơron nhân tạo; Nhà xuất bản Thống kê, 1999.
- [3] Juntakan Taweekun; Load forecasting in Thailand using neural networks; Prince of Songkla University, Hatyai, Songkla, Thailand, 2005.
- [4] Trần Lộc Hùng; Lý thuyết xác suất và thống kê toán học; Nhà xuất bản Giáo dục, 1998.
- [5] Jyh-Ming Kuo; Nonlinear Dynamic Modeling with artificial neural networks; The Degree of Doctor of Philosophy, 1993.
- [6] Frank Scrimgeour; Modelling and Forecasting the Demand for Electricity in New Zealand: A Comparison of Alternative Approaches; The Energy Journal, Volume 24, no.1, 2003.
- [7] Tập đoàn Điện lực Việt Nam (EVN); Tổng kết vận hành hệ thống điện Việt Nam; Hà Nội 2009.
- [8] M. Becvali, M. Cellura, V. Lo Brano, A. Marvuglia; Forecasting daily urban electric load using artificial neural networks; Energy Conversion and management; Vol.45 (2008); pp 2879-2900.
- [9] Mehdi Khashei; An Artificial neural networks (p, d, q) model for timeseries forecasting; Expert system with application; Vol.37 (2010), pp 479-499.

Giới thiệu tác giả:



Tác giả Phạm Ngọc Hùng sinh năm 1976 tại Hải Dương, tốt nghiệp Trường Đại học Bách Khoa Hà Nội và nhận bằng thạc sỹ kỹ thuật điện năm 2006. Tác giả hiện là giảng viên, giảng dạy tại Bộ môn Nhà máy điện và Trạm biến áp, Khoa Hệ thống điện - Trường Đại học Điện lực. Các lĩnh vực nghiên cứu chính: nhà máy điện, quá trình quá độ và năng lượng mới.