

THUẬT TOÁN LỌC SAI SỐ THÔ TRONG HỆ THỐNG ĐO ỨNG DỤNG CÔNG NGHỆ IOT GATEWAY

Đặng Văn Chí*, Nguyễn Thế Lực
Trường Đại học Mở - Địa chất

TÓM TẮT:

Bài báo giới thiệu một thuật toán lọc sai số thô để tự động loại bỏ các giá trị đo không phù hợp, đó là các trị đo bất thường bị phân tán do nhiều nguyên nhân gây ra. Bên cạnh đó thuật toán cũng xác định được kết quả đo trung bình với giả thiết rằng sai số của các trị đo là sai số ngẫu nhiên. Trong đó các trị đo được thu thập thông qua 1 hệ thống đo xa ứng dụng công nghệ IoT Gateway, các dữ liệu đo nhận được từ các sensor đo lường và được giám sát trên Web Server. Với kết quả nghiên cứu và chạy trên mô hình thực tế sẽ cho phép ứng dụng thuật toán này vào các hệ thống đo xa ứng dụng công nghệ IoT nhằm cải tiến và nâng cao độ chính xác của các phép đo.

Từ khóa: lọc sai số thô; sai số ngẫu nhiên; thuật toán lọc thông minh; IoT gateway

1. Mở đầu

Cuộc cách mạng công nghiệp 4.0 đang ngày càng phát triển ở Việt Nam và Thế Giới. Đồng hành với sự phát triển đó, các hệ thống đo xa đa kênh ứng dụng công nghệ IoT Gateway và Web Server đã và đang được khai thác và ứng dụng rất hiệu quả trong thực tế. Ở bất kỳ một hệ thống đo nào cũng không thể tránh được sự xuất hiện của các trị đo dị thường không mong muốn, đó là nguyên nhân gây ra sai số thô. Hầu hết các tín hiệu đo sẽ tác động tới đầu vào của các hệ thống nhận dạng, giám sát và điều khiển và nó sẽ làm sai lệch cho các quyết định. Mặc dù trong thực tế đã có các công trình nghiên cứu với nhiều các giải pháp khác nhau được áp dụng để lọc bỏ các nhiễu loạn này. Tuy nhiên xu thế hiện nay cho các hệ thống IoT là sử dụng công cụ lọc mềm bằng những thuật toán thông minh được nhúng vào thiết bị thu thập và xử lý dữ liệu hoặc có thể xử lý trên Server. Việc nghiên cứu áp dụng thành công thuật toán này sẽ góp phần nâng cao độ chính xác của các phép đo cũng như giảm thiểu các sai lệch đối với hệ thống đo xa ứng dụng IoT Gateway.

Ở Việt Nam, đặc biệt trong ngành đo đạc trắc địa, công tác xử lý các kết quả thông qua quan trắc là rất quan trọng. Trong một bài báo của 2 tác giả: “Trần Ngọc Đông, Nguyễn Hà(2014)” có đề xuất một phương pháp kiểm tra sai số vượt quá giới hạn trong công tác đo, quan trắc lún công trình. Giải pháp được nhóm tác giả đề xuất và đã thử nghiệm thành công khi lọc bỏ được các giá trị đo bất thường. Tuy nhiên công việc tính toán cũng vẫn mang tính chất thủ công và mất nhiều thời gian. Trong một nghiên cứu khác của nhóm tác giả “Trần Đình Trọng & nnk(2014)” có đề xuất phương pháp bình sai theo chuẩn L1 và phương pháp Dikin cũng với mục tiêu xử lý và phát hiện trị đo chứa sai số thô. Tuy nhiên các giải thuật đề xuất lại khá phức tạp và xử lý dữ liệu vẫn là off line.

Trên Thế Giới cũng đã có nhiều công trình, đề tài nghiên cứu đề xuất nhiều các giải pháp cùng các thuật toán đa dạng. Điển hình các tác giả Y. Yang(1999), Khodabandeh A. Amiri-Simkooei A. R(2011) đã đề xuất phương pháp bình sai với khả năng loại bỏ nhiễu sai số thô trong tập các trị đo. Tham khảo trong bài báo của tác giả Koch KR, Yang Y, (1998) đã đề xuất ý tưởng sử dụng bộ lọc Kalman bền vững (Robust Kalman Filter), đây là một trong những kỹ thuật lọc nhiễu rất hiệu quả đã được nhiều nhà Khoa học trên Thế Giới nghiên cứu ứng dụng trong nhiều lĩnh vực khác nhau. Tuy nhiên các thuật toán này lại khá phức tạp và rất khó triển khai khi áp dụng.

Trong bài báo này, các tác giả đề xuất một giải pháp lọc mềm có thuật toán đơn giản, dễ lập trình code và triển khai mà vẫn đạt được mục tiêu cũng như các yêu cầu đặt ra. Trong đó biên độ nhiễu hay độ lớn sai số thô có thể được lọc bỏ thông qua một hệ số điều chỉnh của thuật toán. Kết quả nghiên cứu được nhóm tác giả chạy thử trên mô hình đo xa thông số độ ẩm ứng dụng công nghệ IoT Gateway. Với kết quả nghiên cứu bước đầu này cho phép ứng dụng vào thực tế cho các hệ thống đo xa ứng dụng công nghệ IoT và giám sát trên Server sẽ góp phần cải thiện và nâng cao độ chính xác cho các phép đo.

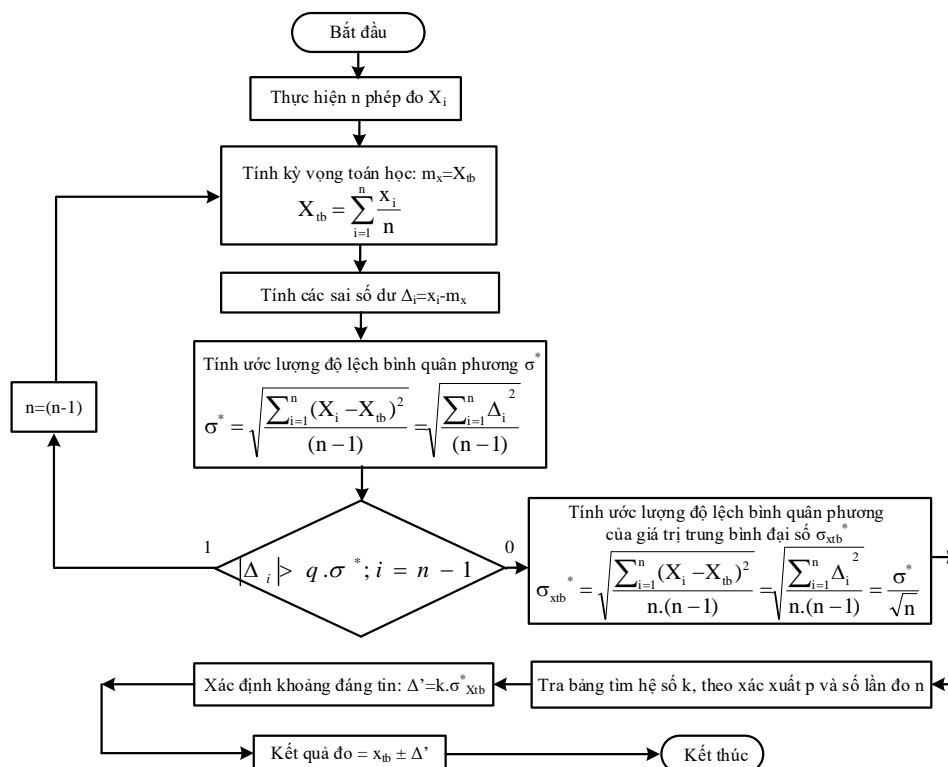
2. Thuật toán lọc sai số thô

Quá trình đo và thu thập tín hiệu đo lường sẽ không tránh khỏi các tín hiệu đo dị thường trộn lẫn vào các giá trị đo thực. Các trị đo không mong muốn này có thể xuất hiện một cách ngẫu nhiên và không theo một quy luật nào. Việc xác định các trị đo sai số ngẫu nhiên dựa trên giả thiết là sai số ngẫu nhiên của các phép đo thường tuân theo luật phân bố chuẩn (luật phân bố Gauss). Các sai số ngẫu nhiên có cùng giá trị

thì có cùng xác suất, có giá trị nhỏ thì xác suất xuất hiện lớn và giá trị lớn thì xác suất xuất hiện nhỏ. Nếu sai số ngẫu nhiên vượt quá một giá trị nào đó thì xác suất xuất hiện sẽ gần như bằng không, vì thế kết quả đo nào có sai số ngẫu nhiên như vậy sẽ bị loại bỏ. Cơ sở lý thuyết của thuật toán mô tả như dưới đây Đặng Văn Chí (2017).

Thực hiện trên gói dữ liệu có n phép đo với các kết quả x_1, x_2, \dots, x_n .

Thuật toán gia công kết quả đo và loại bỏ sai số thô được mô tả như lưu đồ hình 1.



Hình 1. Thuật toán lọc sai số thô và xác định kết quả đo

Giá trị tin cậy nhất đại diện cho x chính là giá trị trung bình của n lần đo:

$$x_{tb} = \frac{\sum_{i=1}^n x_i}{n}$$

Sai số tuyệt đối của một phép đo so với giá trị trung bình

$$\Delta_i = X_i - X_{tb}$$

Theo định nghĩa sai số ngẫu nhiên Δ_i của lần đo thứ i có thể coi là hiệu giữa kết quả đo X_i và kỳ vọng toán học m_x của nó:

$$\Delta_i = X_i - m_x$$

Trong đó:

m_x là kỳ vọng toán học, $m_x = X_{tb}$ khi số lần đo là tương đối lớn

Xác định ước lượng độ lệch bình quân phương σ^*

Thực hiện việc so sánh nếu $|\Delta_i| > q \cdot \sigma^*$ thì giá trị đó sẽ bị loại bỏ, thuật toán quay lại và tiếp tục thực hiện từ đầu tiên.

Trong đó:

q là hệ số hiệu chỉnh về biên độ hoặc độ lớn của trị đo không mong muốn cần loại bỏ, thực tế sẽ tùy thuộc vào đối tượng đo, yêu cầu về độ chính xác phép đo để hiệu chỉnh q cho phù hợp.

Vì giá trị trung bình đại số còn có một sai số ngẫu nhiên nào đó, nên ta đưa ra khái niệm là ước lượng độ lệch bình quân phương của giá trị đại số:

$$\sigma_{x_{tb}}^* = \sqrt{\frac{\sum_{i=1}^n (x_i - x_{tb})^2}{n(n-1)}} = \sqrt{\frac{\sum \Delta_i^2}{n(n-1)}} = \frac{\Delta_i}{n(n-1)}$$

Sai số ngẫu nhiên được tính theo:

$$\Delta' = k \cdot \sigma_{x_{tb}}^*$$

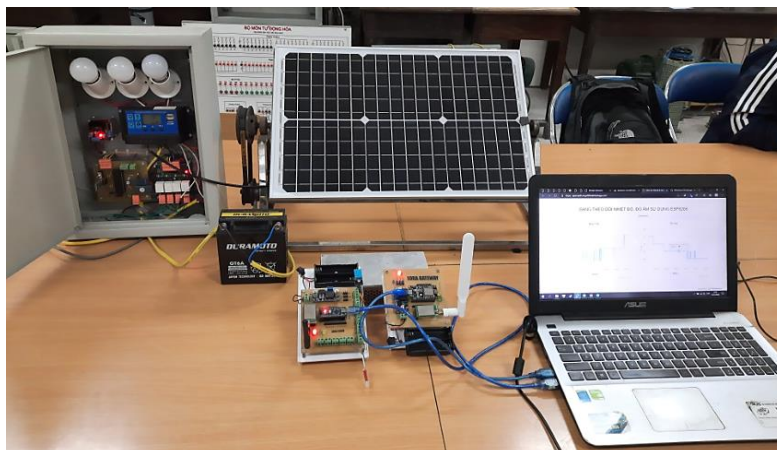
Trong đó:

k: được gọi là hệ số phân bố student, nó phụ thuộc vào phân bố xác suất của sai số ngẫu nhiên, của loại dụng cụ đo và số lượng phép đo n, xác định hệ số k thông qua tra bảng.

Khi đó kết quả đo: $x = x_{tb} \pm \Delta'$

3. Xây dựng mô hình ứng dụng IoT Gateway để đo xa nhiệt độ - độ ẩm trên Server

Nhóm tác giả đã xây dựng và phát triển một hệ thống đo xa ứng dụng công nghệ IoT Gateway để thu thập thông số nhiệt độ và độ ẩm, hình 2, Đặng Văn Chí (2021).



Hình 2. Mô hình đo xa nhiệt độ và độ ẩm IoT Gateway

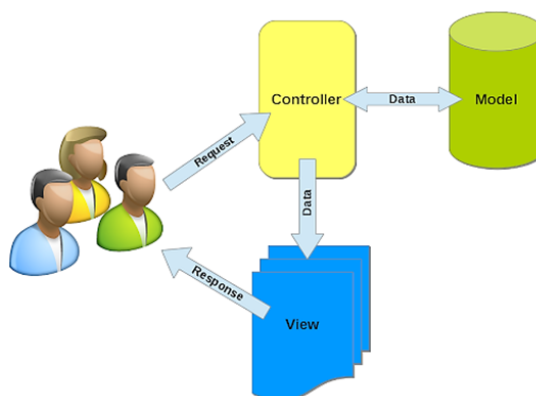
Mô hình gồm có:

- Lora Gateway module: là một thiết bị trung gian nhận thông tin nhiệt độ và độ ẩm từ các Sensor node tại hiện trường bằng sóng Lora và gửi dữ liệu đo được lên Server (Wifi/Ethenet/3G)
- Sensor node module: tích hợp cảm biến đo nhiệt và độ ẩm và truyền về Gateway bằng sóng LoRa
- Giao diện giám sát trên PC: được phát triển trên dịch vụ web hosting: 000webhost.com
- Các thiết bị phụ trợ khác: hệ thống pin năng lượng mặt trời tích hợp bộ điều khiển sạc.

4. Phát triển phần mềm xử lý dữ liệu ứng dụng thuật toán lọc sai số thô

4.1. Mô hình cấu trúc xử lý dữ liệu

Mô hình hệ thống đo xa ứng dụng công nghệ IOT Gateway đã được phát triển như hình 2, trong đó giao diện giám sát trên server được áp dụng dịch vụ web hosting: 000webhost.com. Hệ thống này được sử dụng để xây dựng mô hình cấu trúc xử lý dữ liệu, nền tảng mô hình cơ sở MVC (Model View Controller) để xuất như hình 3.



Hình 3. Mô hình cấu trúc xử lý dữ liệu MVC

4.2. Phát triển phần mềm xử lý dữ liệu

Phần mềm xử lý dữ liệu trên Server được xây dựng trên nền tảng .Net Core theo mô hình MVC. Chương trình tiếp nhận dữ liệu gửi tới dưới định dạng json, sau đó dữ liệu sẽ được xử lý theo thuật toán có lưu đồ như hình 1 để loại bỏ những giá trị bất thường. Dữ liệu sau khi được xử lý sẽ có độ tin cậy và chính xác

cao hơn với sai số nằm trong một giới hạn cho phép.

Trong mô hình MVC, dữ liệu được gửi đến Controller thông qua cấu trúc dữ liệu được định nghĩa bởi Model. Controller sẽ tiến hành xử lý dữ liệu và trả kết quả về phía người dùng thông qua View, trích dẫn một đoạn code thực hiện chương trình được cho như dưới đây:

Trong đoạn code hàm DataFilter() (trong DataController) sẽ xử lý theo các gói dữ liệu đầu vào, và xuất ra kết quả.

```
0 references
public class DataController : Controller
{
    0 references
    public IActionResult Index()
    {
        List<Data> model = DataFilter();
        return View(model);
    }
    1 reference
    private List<Data> DataFilter()
    {
        List<Data> inputData;
        List<Data> errorData=null;
        using (StreamReader r=new StreamReader("DataUpload\\datalog.json"))
        {
            string dataString = r.ReadToEnd();
            inputData = JsonConvert.DeserializeObject<List<Data>>(dataString);
        }
    }
}
```

Trích dẫn một đoạn code thực hiện quá trình xử lý dữ liệu trên Server được mô tả ở dưới:

```
private List<Data> DataFilter()
{
    List<Data> inputData;
    List<Data> errorData=null;
    using (StreamReader r=new StreamReader("DataUpload\\datalog.json"))
    {
        string dataString = r.ReadToEnd();
        inputData = JsonConvert.DeserializeObject<List<Data>>(dataString);
    }
    if (inputData != null)
    {
        double sumTemperature = 0, avgTemperature=0;
        double sumHumidity = 0, avgHumidity=0;
        foreach(Data dataItem in inputData)
        {
            sumTemperature += dataItem.temperature;
            sumHumidity += dataItem.humidity;
        }
        int n = inputData.Count;
        avgTemperature = sumTemperature / n;
        avgHumidity = sumHumidity / n;
        errorData = new List<Data>();
        for(int i = 0; i < n; i++)
        {
            Data dataObj = inputData[i];
            Data errorDataObj = new Data();
            errorDataObj.temperature = dataObj.temperature - avgTemperature;
            errorDataObj.humidity = dataObj.humidity - avgHumidity;
            errorDataObj.dataId = dataObj.dataId;
            errorDataObj.time_logging = dataObj.time_logging;
            errorData.Add(errorDataObj);
        }
    }
}
```

Các dữ liệu được lưu trữ dưới dạng cấu trúc dữ liệu json: trích đoạn code mô tả như dưới:

```
{
    "dataId": 1236,
    "temperature": 31,
    "humidity": 81,
    "time_logging": "2021-05-20 08:12:00"
},
```

4.3. Kết quả nghiên cứu và thảo luận

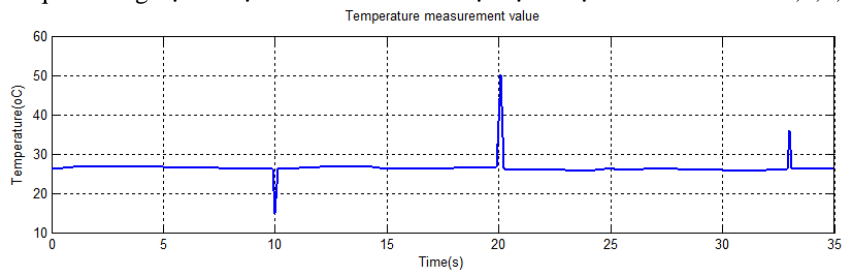
Tập dữ liệu đo 2 thông số nhiệt độ và độ ẩm là kết quả chạy trên hệ thống IoT Gateway như trên hình 2. Khi chưa áp dụng thuật toán lọc, công việc xử lý để loại bỏ các giá trị đo bất thường, không hợp lý được áp dụng bằng phương pháp quan sát bảng dữ liệu và xóa thủ công. Công việc này mất rất nhiều thời gian và không hiệu quả khi làm việc với nhiều dữ liệu. Cá biệt trong một số ứng dụng yêu cầu phải xử lý online thì giải pháp xử lý trên là không khả thi.

Gói dữ liệu lưu trong cơ sở dữ liệu Excel dùng để test thử nghiệm thuật toán mô tả như bảng 1

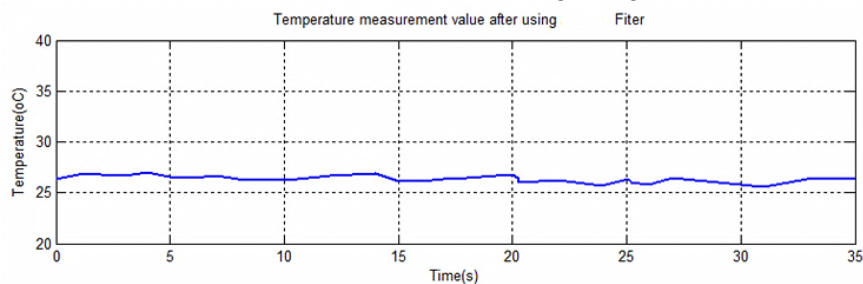
Bảng 1. Dữ liệu đo xa 2 thông số nhiệt độ và độ ẩm trên excel

id	sensor	nhiệt_đo	đo_ẩm	time_act
2574	"node2"	"26.8"	"81"	"2020-12-14 15:08:14"
2575	"node2"	"26.8"	"81"	"2020-12-14 15:08:16"
2576	"node2"	"26.7"	"81"	"2020-12-14 15:08:17"
2577	"node2"	"26.7"	"82"	"2020-12-14 15:08:19"
2578	"node2"	"26.7"	"82"	"2020-12-14 15:08:20"
2579	"node2"	"26.7"	"82"	"2020-12-14 15:08:21"
2580	"node2"	"26.7"	"82"	"2020-12-14 15:08:23"
2581	"node2"	"15.7"	"96"	"2020-12-14 15:08:24"
2582	"node2"	"26.7"	"82"	"2020-12-14 15:08:25"
2583	"node2"	"26.7"	"81"	"2020-12-14 15:08:27"
2584	"node2"	"26.7"	"80"	"2020-12-14 15:08:28"
2585	"node2"	"26.7"	"80"	"2020-12-14 15:08:30"
2586	"node2"	"26.6"	"80"	"2020-12-14 15:08:31"
2587	"node2"	"26.7"	"79"	"2020-12-14 15:08:34"
2588	"node2"	"26.7"	"80"	"2020-12-14 15:08:35"

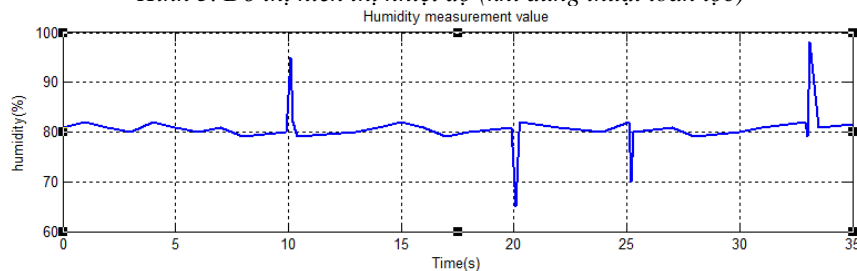
Kết quả thử nghiệm thuật toán cho 2 biến nhiệt độ và độ ẩm trên các hình 4,5,6,7.



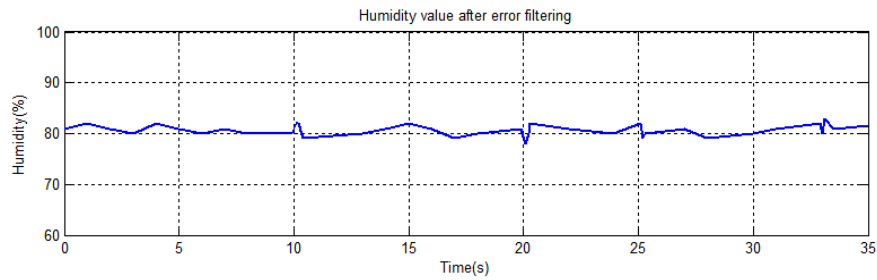
Hình 4. Đồ thị hiển thị nhiệt độ (giá trị gốc)



Hình 5. Đồ thị hiển thị nhiệt độ (khi dùng thuật toán lọc)



Hình 6. Đồ thị hiển thị độ ẩm (giá trị gốc)



Hình 7. Đồ thị hiển thị độ ẩm (khi dùng thuật toán lọc)

Trên hình 4 và hình 6 là giá trị dữ liệu bản gốc xuất hiện các xung nhiễu với các giá trị bất thường không phù hợp so với giá trị trung bình. Hình 5 và hình 7 là kết quả quá trình chạy chương trình lọc sai số thô.

Chương trình được thử nghiệm với hệ số lọc $q = \text{const}$ do biên độ của các xung nhiễu không có sự dao động ngẫu nhiên lớn cho nên các kết quả trên thỏa mãn được mục tiêu và yêu cầu đặt ra. Tùy thuộc vào môi trường làm việc của hệ thống, thuật toán có thể thay đổi hệ số q này sao cho phù hợp để nâng cao hiệu quả lọc sai số.

Thực tế xử lý dữ liệu với nhiều nguồn gây nhiễu khác nhau, tín hiệu đa sóng hài có thể trộn lẫn vào tín hiệu đo. Khi đó đề xuất giải pháp là có thể ghép tầng các bộ lọc trên với nhau để tăng hiệu quả của thuật toán.

5. Kết luận

Bài báo cung cấp một thuật toán lọc bỏ sai số thô để xử lý dữ liệu đo lường. Nhóm nghiên cứu đã ứng dụng thành công vào xử lý dữ liệu đo nhiệt độ và độ ẩm được thực hiện trên mô hình đo xa ứng dụng công nghệ IoT gateway.

Thuật toán được đánh giá là dễ dàng áp dụng và thực hiện, nhất là việc lập trình tính toán không phức tạp, khả năng tính toán nhanh. Với kết quả nghiên cứu bước đầu này cho phép thay thế các phép kiểm tra thống kê thủ công trước đây để phát hiện sai số.

Kết quả nghiên cứu được các tác giả thực hiện trên 1 gói dữ liệu với số lượng phép đo không nhiều cũng như biên độ các xung còn chưa đa dạng. Tuy nhiên thuật toán cũng đã đáp ứng được mục tiêu đặt ra.

Trong thực tế hoạt động của các hệ thống đo lường xa, nguồn nhiễu sẽ xuất hiện ngẫu nhiên với với nhiều thành phần sóng hài được trộn lẫn vào tín hiệu đo. Đề xuất tiếp tục nghiên cứu cải tiến ghép tầng thuật toán và thay đổi hệ số lọc ở các tầng khác nhau để có thể nâng cao chất lượng của bộ lọc và hiệu quả làm việc của thuật toán.

Đóng góp các tác giả :

1. Tác giả 1: Đề xuất ý tưởng, nghiên cứu cơ sở lý thuyết, xây dựng thuật toán, sơ đồ nguyên lý của hệ thống, tổng hợp và viết bài.
2. Tác giả 2: Lập trình viết code, phát triển mềm nhúng cho thuật toán và chạy thực nghiệm

Tài liệu tham khảo

Bài báo đăng trên tạp chí

Trần Ngọc Đông, Nguyễn Hà, (2014). Phương pháp kiểm tra sai số vượt giới hạn trong các trị đo quan trắc lún công. Tạp chí KHCN Xây dựng, số 1, 45-52.

Trần Đình Trọng, Đào Duy Toàn, Vũ Đình Chiêu, (2014). Phương pháp Dikin phát hiện trị đo chứa sai số thô. Tạp chí Khoa học và Công nghệ 52 (4B), 519-526.

Y. Yang, (1999). Robust estimation of geodetic datum transformation, Journal of geodesy 73, 268–274.

Khodabandeh A. Amiri-Simkooei A. R, (2011). Recursive algorithm for L1 norm estimation in linear models, Journal of Surveying Engineering (ISSN 0733-9453) 137 (1) 1-8.

Koch KR, Yang Y, (1998). Robust Kalman filter for rank deficient observation model. Journal of Geodesy 72: 436±441

Sách tham khảo

Đặng Văn Chí, (2017). Kỹ thuật đo lường điện và các thuật toán đo phi điện, NXB Xây dựng, 298

trang.

Đặng Văn Chí & nnk (2021). Đo lường điều khiển bằng máy tính - ứng dụng công nghệ IoT-Lora Gateway để giám sát và điều khiển xa trên web server, NXB Khoa học – kỹ thuật, 285 trang.

ABSTRACT

Algorithm to filter raw error in measurement system using IoT gateway technology

Đặng Văn Chí*, Đào Hiếu, Nguyễn Thế Lực
Hanoi University of Mining and Geology

The article introduces error filtering algorithm to automatically filter inappropriate measured values, it is an abnormal measured value that is scattered due to many causes. Besides, the algorithm also determines the average measurement result, assuming that the error of the measured values is a random error. In which the measured values are collected through a telemetry system applying Gateway technology. In which the measured values are collected through a telemetry system applying Gateway technology, the measurement data is received from the measurement sensors and monitored on the Web Server. With the research results and running on the actual model, it will allow the application of this algorithm to telemetry systems applying IoT technology to improve the accuracy of measurements.

Keywords: error filtering; random error; intelligent filtering algorithm; IoT gateway