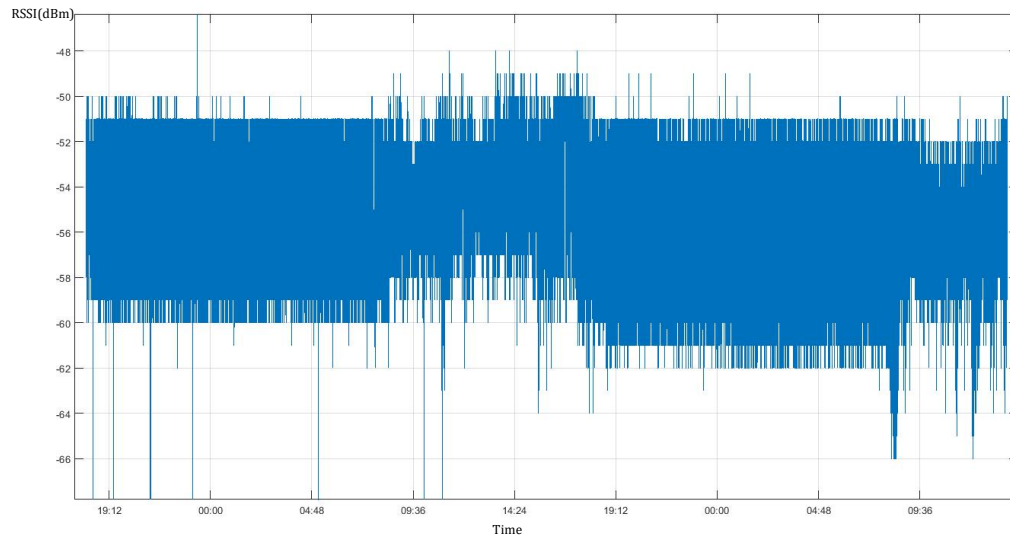


Chương 7. Kết quả thực nghiệm

7.1 Lựa chọn thông số Kalman

Theo như đã đề cập tại phần 4.2, ta cần phải biết trước các thông số $= \pm\Delta RSSI$, và $P_{k-1} = Q$ bằng phương pháp thực nghiệm.

Tiến hành lắp đặt một Beacon và một ESP32 để thu thập dữ liệu RSSI thô, ta có được hình sau:



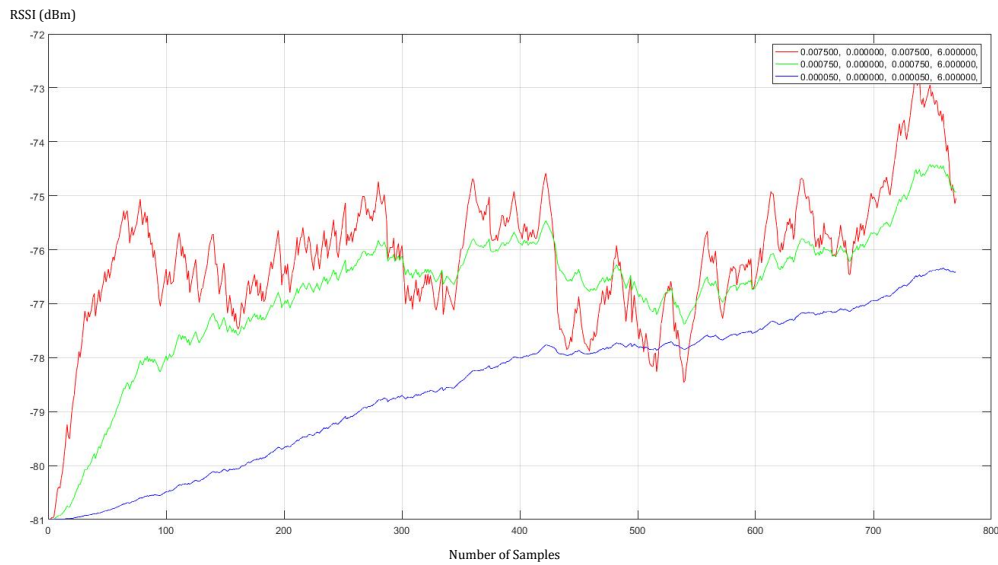
Hình 7.1: Dữ liệu RSSI thô

Qua hình 7.1, ta có thể thấy RSSI dao động từ -62 đến -50 dBm, vậy lúc này $R = \pm\Delta RSSI = \pm 6 \text{ dBm}$. Chọn $R = 6$.

Tiếp theo, ta tiếp tục sử dụng mô hình trên để thu thập dữ liệu RSSI thô, tuy nhiên dữ liệu thô này sẽ được đưa vào Kalman Filter với $R = 6$ trong một vài trường hợp có $P = Q$ khác nhau.

Xét 3 bộ thông số Kalman $[P, K, Q, R]$ như sau:

$$\begin{cases} P = 0.007500, K = 0, Q = 0.007500, R = 6 & (1) \\ P = 0.000750, K = 0, Q = 0.000750, R = 6 & (2) \\ P = 0.000050, K = 0, Q = 0.000050, R = 6 & (3) \end{cases}$$



Hình 7.2: Lựa chọn thông số Kalman

Hình 7.2 thể hiện RSSI đã qua Kalman Filter với các bộ thông số khác nhau. Ta có thấy nhận thấy, bộ thông số thứ (3) (màu xanh dương) thể hiện dữ liệu sau khi lọc khá mượt và trơn, tuy nhiên nếu so với hai bộ dữ liệu còn lại thì ta thấy bộ dữ liệu này mượt quá mức, làm mất hết các đặc tính ảnh hưởng của môi trường xung quanh, ngoài ra còn làm chậm quá trình xác lập, cho nên bộ dữ liệu này không tốt.

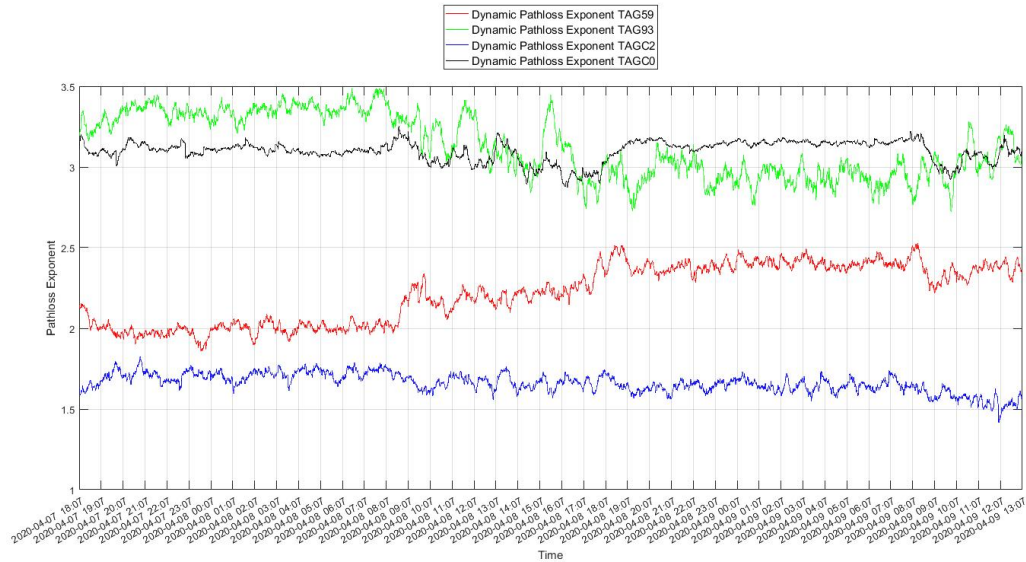
Xét bộ dữ liệu thứ (2) (màu xanh lá cây), ta thấy tuy độ mượt và trơn không bằng bộ dữ liệu thứ (3) nhưng bộ dữ liệu này vẫn giữ được các đặc tính ảnh hưởng của môi trường, nhưng độ sai lệch không quá lớn như bộ dữ liệu thứ (1).

Vậy ta sẽ sử dụng bộ thông số Kalman thứ (2) cho các bài thí nghiệm sau này.

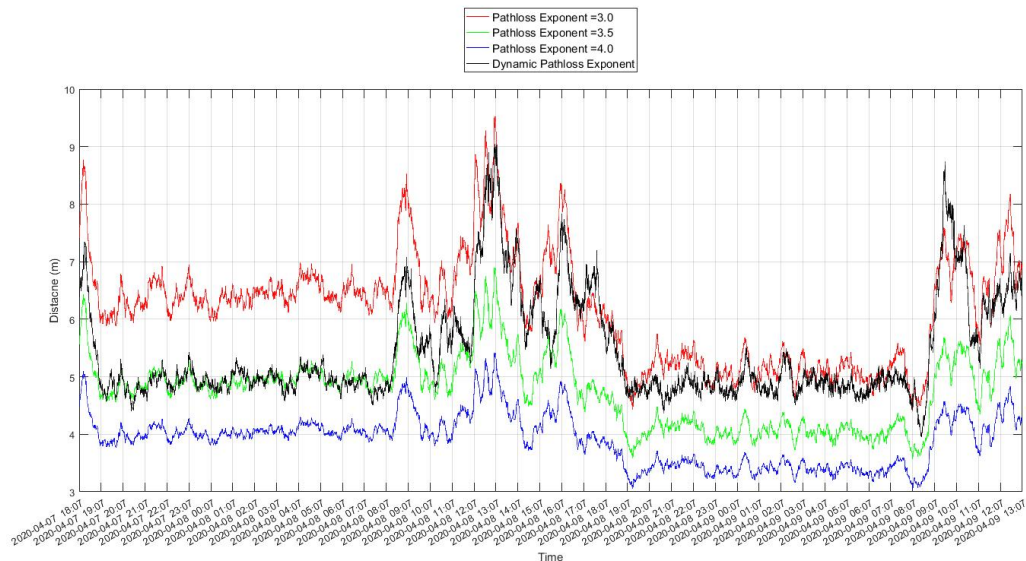
$$P = 0.000750, K = 0, Q = 0.000750, R = 6 \quad (7.1)$$

7.2 Đánh giá Pathloss Exponent Improvement

Trong phần này chúng ta sẽ đánh giá độ chính xác của giải thuật Pathloss Exponent Improvement. Tuy nhiên trong phần này ta sẽ không áp dụng thuật toán tương quan RSSI với góc để có cái nhìn đúng nhất về giải thuật này, nhưng để hạn chế sai số, ta sẽ đặt Pathloss và Tag không lệch góc nhau quá lớn ($<20^\circ$).



Hình 7.3: Tham số Pathloss Exponent thay đổi theo thời gian



Hình 7.4: Sự khác biệt giữa Static và Dynamic Pathloss Exponent

Trong hình 7.3, ta xem xét nhiều tham số Pathloss cùng một lúc để có đánh giá chung về tham số này. TAGXX có nghĩa là thiết bị Tag với tín hiệu RSSI của Beacon có 2 ký tự cuối trong Mac Address là XX. Bốn thiết bị này được đặt xa nhau, đặt trong bốn góc của căn phòng. Từ đây ta thấy, tuy là chung một căn phòng nhưng tham số Pathloss có thể khác nhau, dẫn đến các kết quả khác nhau.

Ta chọn một thiết bị trong bốn thiết bị trên để tính khoảng cách dự vào tham số Pathloss tìm được ở trên. Ta gọi tham số Pathloss được cài đặt cứng là **Static Pathloss Exponent**, tham số Pathloss tìm được theo thời gian là **Dynamic Pathloss Exponent**.

Trong hình 7.4, ta chọn thiết bị TAG93, lúc này khoảng cách từ Tag và Pathloss lần lượt là 5.95m và 7.04m. Lúc này sai số khi sử dụng Dynamic Pathloss Exponent lặp lại sau khoảng thời gian biến động mạnh về môi trường (ở giữa biểu đồ). Chứng tỏ Dynamic Pathloss Exponent hoạt động khá tốt.

Nếu ta sử dụng Static Pathloss Exponent, ta có thể thấy sau khoảng thời gian biến động mạnh về môi trường, khoảng cách tính được của thiết bị khác với khoảng cách tính được trước lúc biến động, lúc này ta không thể cố định được sai số tính toán, gây ảnh hưởng đến kết quả cuối cùng của chúng ta.

Vậy, giải thuật Dynamic Pathloss Exponent tỏ ra khá hiệu quả.

7.3 Particle Swarm Optimization

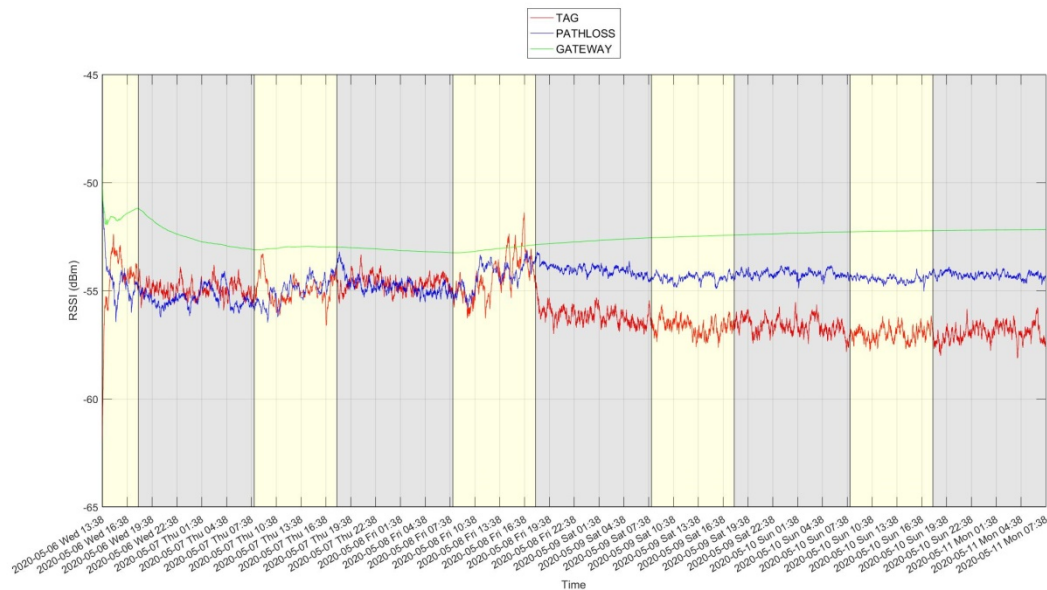
Sau khi xác định được bộ lọc Kalman và giải thuật Pathloss Exponent Improvement hoạt động khá tốt, ta sẽ xem xét đến mức độ hiệu quả khi áp dụng giải thuật Particle Swarm Optimization. Trong phần này, ta vẫn chưa dùng đến phương pháp chỉnh sửa RSSI để có cái nhìn tổng quan về giải thuật này.

Tiến hành lắp đặt thiết bị trong căn phòng, chọn một vị trí làm mốc (0,0), đo đạc vị trí của các thiết bị ta có bảng sau:

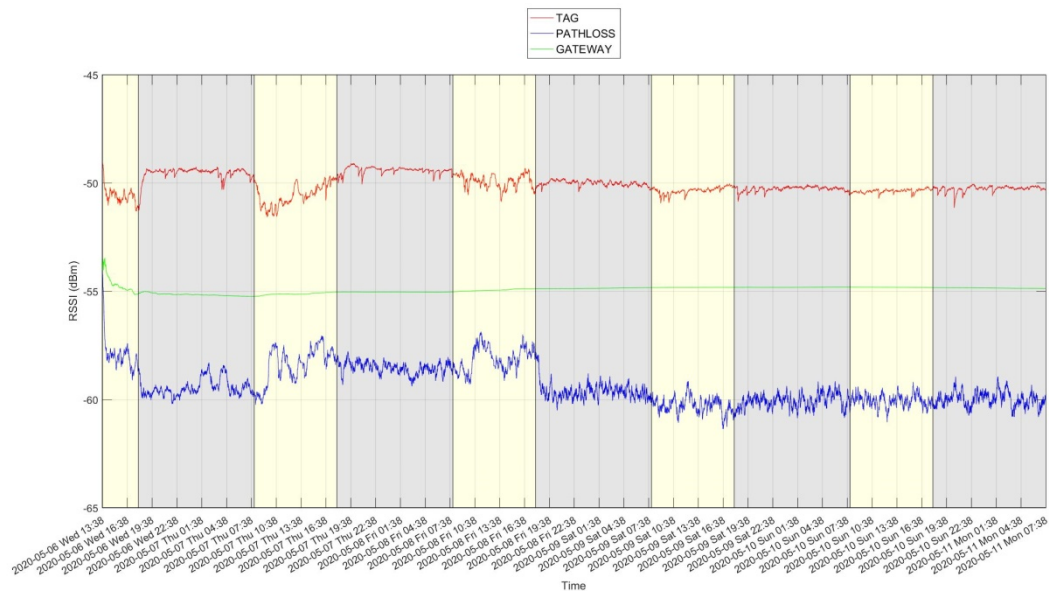
	B59	B93	BC2	BC0	Tag	Pathloss
x	1.34	7.55	7.55	1.50	2.43	2.33
y	0.20	1.50	11.66	11.66	6.33	4.53

Bảng 7.1: Vị trí các thiết bị được cài đặt.

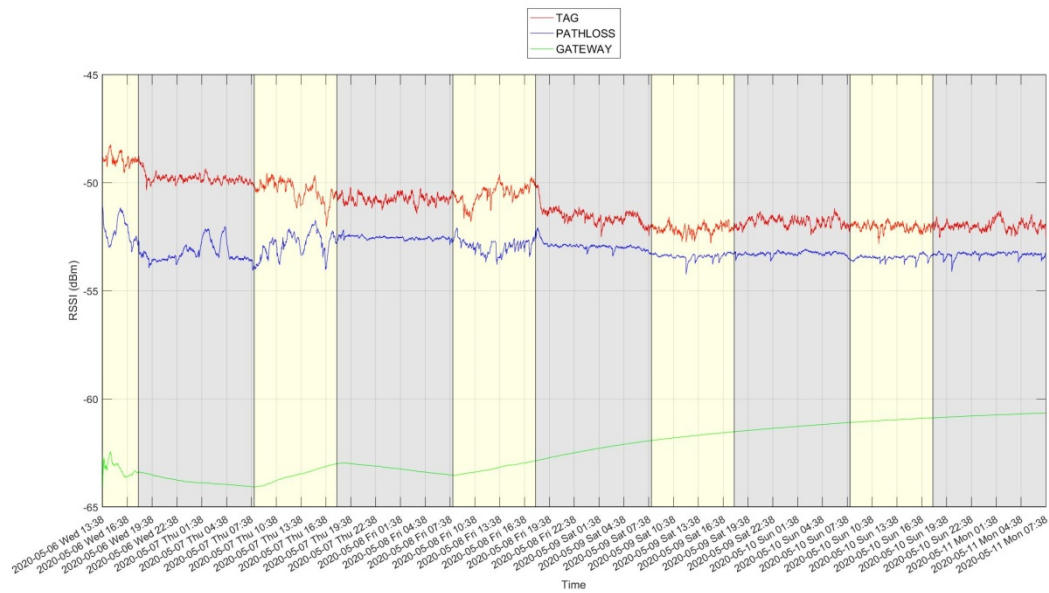
Trong đó BXX là Beacon có hai ký tự cuối trong Mac Address là XX. Lúc này ta có kết quả như sau:



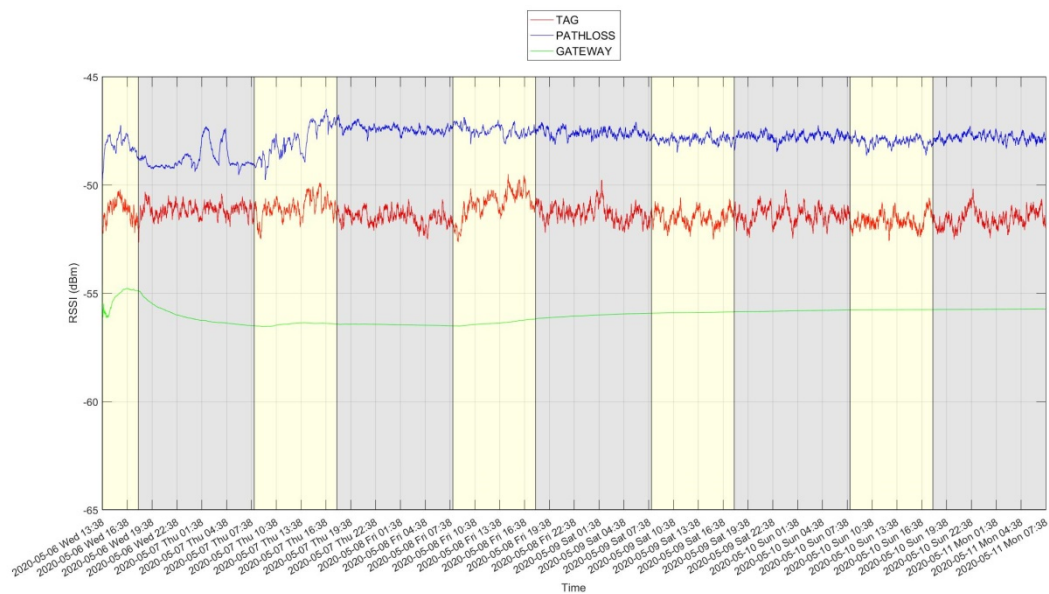
Hình 7.5: Kết quả RSSI B59 sau Kalman



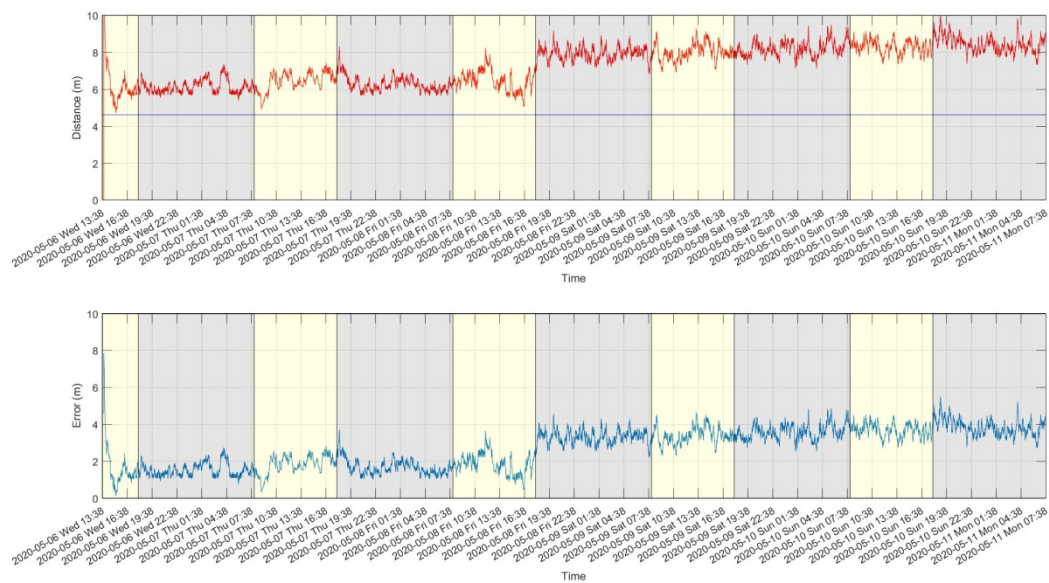
Hình 7.6: Kết quả RSSI B93 sau Kalman



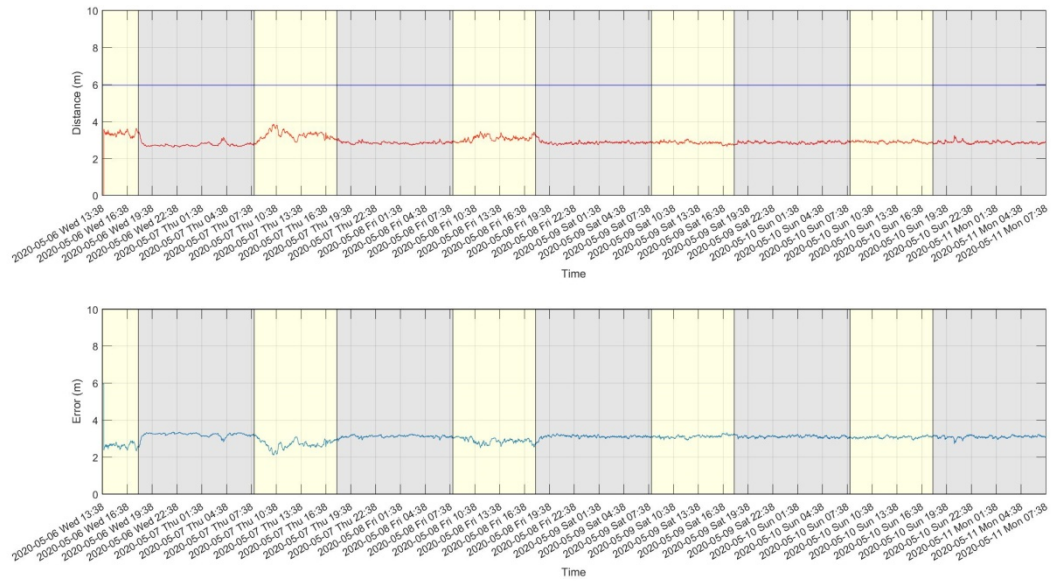
Hình 7.7: Kết quả RSSI BC2 sau Kalman



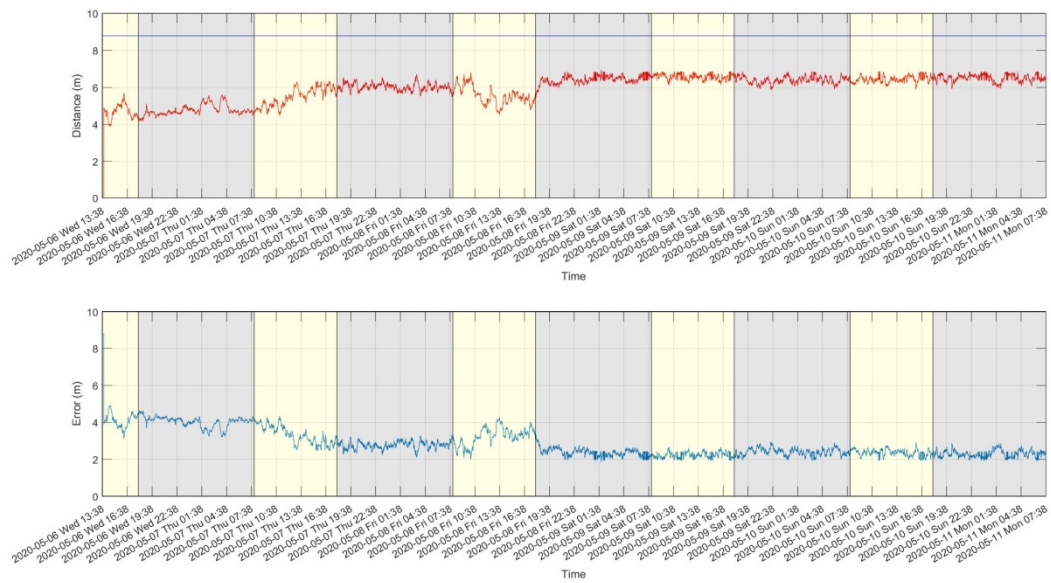
Hình 7.8: Kết quả RSSI BC0 sau Kalman



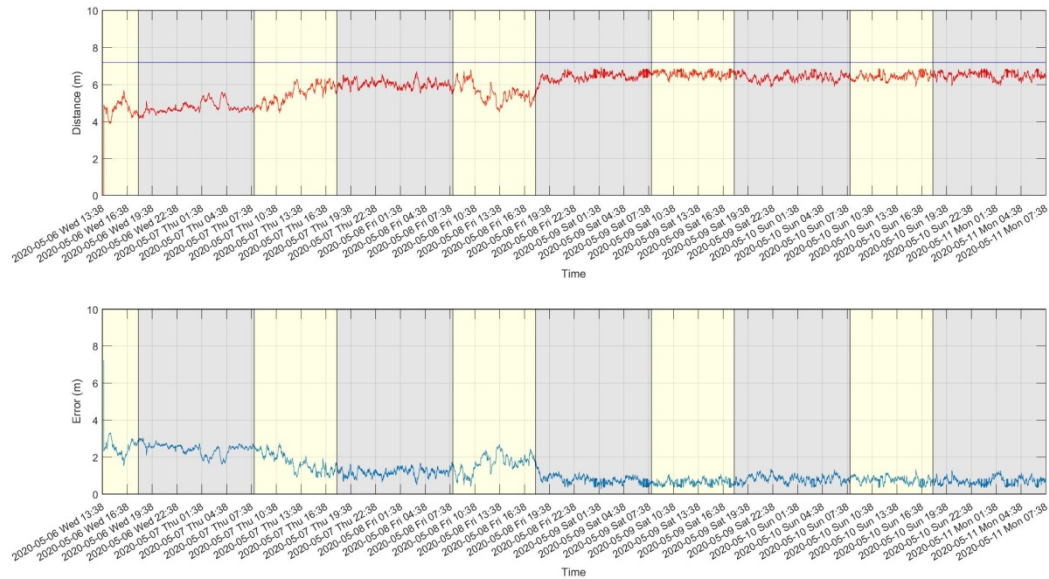
Hình 7.8: Khoảng cách và Sai số khoảng cách của B59



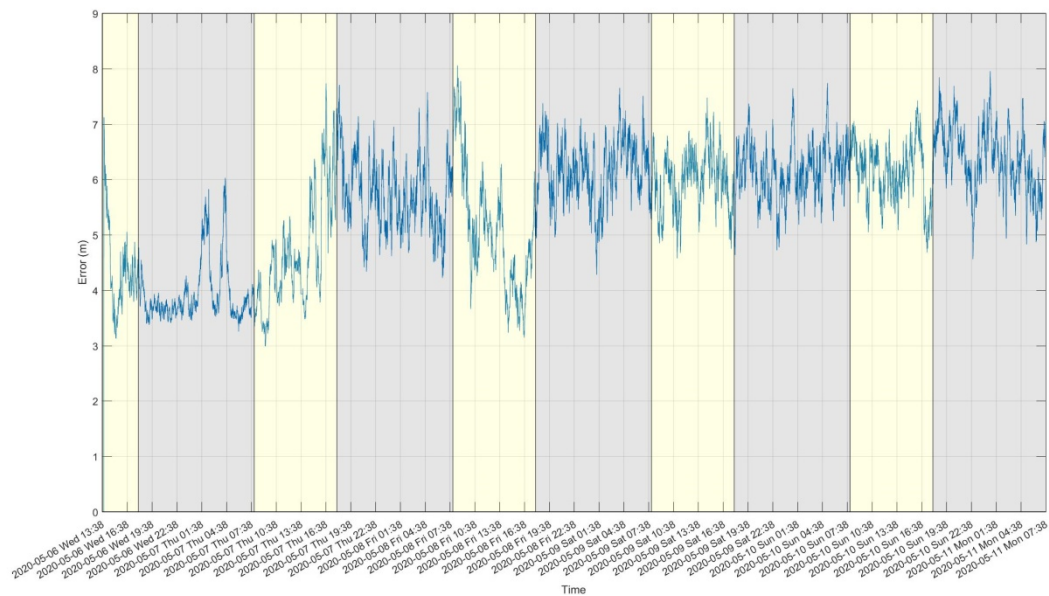
Hình 7.9: Khoảng cách và Sai số khoảng cách của B93



Hình 7.10: Khoảng cách và Sai số khoảng cách của BC2



Hình 7.10: Khoảng cách và Sai số khoảng cách của BC2



Hình 7.11: Sai số vị trí tìm được khi dùng Particle Swarm Optimization

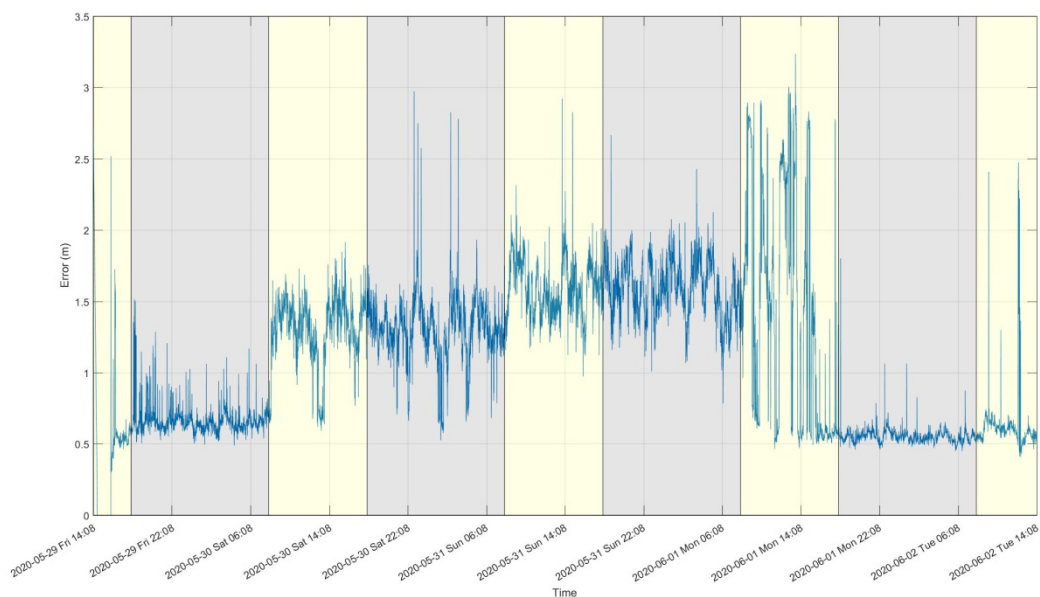
Ta thấy sai số vị trí đạt được tuy lớn nhưng có độ ổn định nhất định. Vậy nếu ta tối ưu được các tác động của môi trường thì sẽ đạt được sai số tối ưu ổn định.

7.4 Áp dụng giải thuật chỉnh sửa RSSI

Trong phần trước, khi áp dụng Particle Swarm Optimization ta đạt được sai số ổn định nhưng sai số còn lớn. Lúc này ta sẽ dùng giải thuật Chỉnh sửa RSSI để tối thiểu yếu tố ảnh hưởng từ Phân cực Antenna.

Tiếp tục sử dụng biểu đồ ở hình 5.10 để cải tiến cho Particle Swarm Optimization.

Khi đó ta sẽ có kết quả như sau:



Hình 7.12: Sai số đạt được khi áp dụng thêm Chỉnh sửa RSSI

Ta nhận thấy sau khi áp dụng giải thuật Chỉnh sửa RSSI, sai số cuối cùng giảm đi đáng kể, ở trường hợp bình thường sẽ đạt được sai số khoảng 0.5m, ở trường hợp xấu sẽ đạt sai số khoảng 2m. Điều này chứng tỏ thuật toán Chỉnh sửa RSSI hoạt động tốt, hiệu quả, giảm thiểu được yếu tố ảnh hưởng.

Trong chương kế tiếp, chúng ta sẽ tiến hành lắp đặt hệ thống trong không gian lớn hơn và thực hiện thí nghiệm di chuyển theo thời gian thực.