

Kierunek: **Informatyka algorytmiczna (INA)**

**PRACA DYPLOMOWA**  
**MAGISTERSKA**

**Mechanizm multilateracji w  
rozproszonej sieci sensorów audio**

**Multilateration mechanism in  
distributed net of audio sensors**

Gabriel Budziński

Opiekun pracy  
**dr inż. Przemysław Błaśkiewicz**

Słowa kluczowe: multilateracja, sensory audio, synchronizacja czasu

---

## Streszczenie

Problem pozycjonowania w przestrzeni na podstawie emitowanego dźwięku obiektu pozycjonowanego wiąże się z wykorzystaniem możliwie zsynchronizowanych w czasie węzłów (mikrofonów) i pomiarze różnic czasu odbioru dźwięku przez czujniki. W pracy zostanie zbudowana sieć (co najmniej 4 sztuki) sensorów audio połączonych bezprzewodowo między sobą i ze stacją główną. Zadaniem sieci będzie wskazanie lokalizacji w przestrzeni punkowego przedmiotu emitującego dźwięk. Oprócz wyboru i implementacji algorytmu multilateracji zaproponowane zostanie rozwiązanie problemu synchronizacji czasu między sensorami, minimalizacji opóźnień w komunikacji oraz kalibracji systemu.

**Słowa kluczowe:** multilateracja, sensory audio, synchronizacja czasu

## Abstract

The problem of positioning in space based on the emitted sound of the positioned object involves the use of as closely synchronized nodes (microphones) as possible in time and measuring the differences in the time of sound reception by sensors. In the work, a network (of at least 4 units) of audio sensors connected wirelessly to each other and to the main station will be built. The network's task will be to indicate the location in space of a point-like object emitting sound. In addition to selecting and implementing the multilateration algorithm, a solution to the problem of time synchronization between sensors, minimizing communication delay, and system calibration will be proposed.

**Keywords:** multilateration, WASN, clock synchronization

# Spis treści

<b>Wstęp</b>	<b>4</b>
<b>1. Sformułowanie problemu</b>	<b>5</b>
1.1. Aktualny stan wiedzy	5
1.2. Obrane rozwiązanie	6
1.3. Przykład problemu	7
<b>2. Sprzęt systemowy</b>	<b>9</b>
2.1. Broker MQTT	9
2.2. Serwer obliczeniowy	9
2.3. Węzły	9
<b>3. Eksperyment zerowy</b>	<b>11</b>
3.1. Opis działania	11
3.2. Ewaluacja działania systemu	13
3.3. Interpretacja wyników i wnioski	15
<b>4. Synchronizacja czasu</b>	<b>16</b>
4.1. Synchronizacja programowa	16
4.2. Synchronizacja sprzętowa	19
<b>5. Multilateriacja</b>	<b>22</b>
5.1. Przygotowanie danych wejściowych	22
5.2. Ocena działania systemu	25
5.3. Interpretacja i wnioski	32
<b>Podsumowanie</b>	<b>34</b>
<b>Literatura</b>	<b>35</b>

# Wstęp

Multilateracja jest metodą lokalizacji opartą na odległościach od punktów o znanych współrzędnych do szukanego punktu. Zazwyczaj odległości te uzyskuje się na podstawie czasu, w którym sygnał dotarł do odbiornika (*ang. time of arrival, TOA*). Problem wyliczania pozycji punktu w przestrzeni jest kluczowy w wielu współczesnych serwisach lokalizacyjnych, takich jak GPS (*ang. Global Positioning System*). Był on badany przynajmniej od lat 60., ze znacznymi postępami poczynionymi w następnej dekadzie podczas opracowywania systemu GPS, a w latach 90. w związku z błyskawicznym wzrostem liczby telefonów komórkowych i potrzebą lokalizacji osób komunikujących się przy ich użyciu z służbami ratunkowymi [10]. W ostatnich latach coraz większym zainteresowaniem cieszy się zastosowanie metod multilateracji w sieciach urządzeń IoT [6] w celu autonomicznego uzyskiwania informacji lokalizacji przestrzennej bez potrzeby użycia dedykowanej do tego celu aparatury.

Na przestrzeni lat rozwijano wiele wariantów rozwiązań problemu lokalizacji na podstawie czasu otrzymania sygnału. Pierwszym były metody iteracyjne, dzięki którym w prosty sposób otrzymywano rozwiązanie równań nieliniowych, ale nie gwarantowały ani zbiegania do poprawnego rozwiązania, ani krótkiego czasu jego otrzymania. W latach 80. wprowadzono metody manipulacji układami równań tak, by otrzymać równanie kwadratowe jednej niewiadomej dające dwa rozwiązania, z których jedno było poprawne. Wreszcie w połowie lat 90. zaczęto używać układów równań liniowych, których rozwiązania otrzymywano dzięki programowaniu liniowemu. Zaletą tej metody jest jej wszechstronność, ponieważ daje możliwość łatwego rozszerzenia problemu o więcej pomiarów. Te dodatkowe informacje mogą być dodatkowymi pomiarami TOA lub każdymi innymi związanymi z problemem przy pomocy równań liniowych. Dzięki temu mogły rozwijać się algorytmy lokalizacji wykorzystujące kąt przybycia (*ang. angle of arrival, AOA*), różnicę częstotliwości (*ang. frequency difference of arrival, FDOA*), moc odebranego sygnału (*ang. received signal strength, RSS*) oraz kierunek przybycia<sup>1</sup> (*ang. direction of arrival, DOA*). W poniższej pracy skupimy się na podstawowym wariancie multilateracji, czyli TOA.

Założeniem pracy jest zaprojektowanie i zaimplementowanie systemu multilateracyjnego działającego w domenie fal dźwiękowych, opartego o szeroko dostępne mikrokontrolery z rodziny **Arduino**. System będzie składał się z zsynchronizowanych ze sobą węzłów odbiorczych (mikrofonów) oraz węzła nadawczego (brzęczyka). W skład urządzeń systemowych będzie wchodził również centralny serwer obliczeniowy, który będzie wykonywał obliczenia multilateracyjne na podstawie odległości estymowanych przy pomocy różnicy czasu nadania i odebrania sygnału dźwiękowego przez każdy z węzłów. Zanim system będzie zdolny zwracać choćby w przybliżeniu poprawne lokalizacje, potrzebne będzie rozwiązanie problemów wynikających z nieadekwatnej synchronizacji zegarów urządzeń systemowych i nieoczekiwanych wpływów wzmacniacza operacyjnego, wbudowanego w zastosowane mikrofony. Każdy z tych podproblemów, jak również ostateczna multilateracja, opatrzone będą odpowiednimi eksperymentami, których wyniki przedstawione będą na wykresach.

---

<sup>1</sup>Różnicą między DOA a AOA jest odległość, z której odbierany jest sygnał. DOA zakłada, że źródło sygnału jest na tyle daleko, że AOA jest wspólny dla wszystkich odbiorników. W AOA różnice pomiarów są na tyle duże, aby móc użyć ich do triangulacji.

# Rozdział 1

## Sformułowanie problemu

Weźmy punkt  $(x, y, z) \in \mathbb{R}^3$  i oznaczmy go  $N$ , ponadto weźmy zbiór  $n$  punktów  $\{(x_i, y_i, z_i) : i \in [n]\} \subset \mathbb{R}^3$ , który oznaczmy  $O$ , a punkty należące do zbioru odpowiednio  $O_i$ . Niech  $d_i$  będzie odległością  $d(N, O_i)$ . Znając współrzędne punktów  $O_i$  oraz odległości  $d_i$  chcemy znaleźć  $(x, y, z)$ . Można to osiągnąć rozwiązując układ równań

$$\begin{cases} (x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 = d_1^2 \\ (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 = d_2^2 \\ \vdots \\ (x - x_n)^2 + (y - y_n)^2 + (z - z_n)^2 = d_n^2. \end{cases} \quad (1.1)$$

### 1.1. Aktualny stan wiedzy

W dostępnej literaturze obejmującej temat multilateracji i systemów multilateracyjnych opisane są trzy główne podejścia:

- iteracyjne,
- algebraiczne,
- liniowe.

Metoda iteracyjna dąży do minimalizacji błędu przybliżonego rozwiązania układu równań 1.1. Typowymi narzędziami w niej wykorzystywanymi są: metoda najszybszego spadku [7], metoda Newtona oraz szereg Taylora [5]. Praca [5] zasuguje na szczególną uwagę ze względu na skupienie się na zastosowaniu metody do celów lokalizacji, w przeciwieństwie do ogólnego podejścia do problemu optymalizacyjnego. Największymi problemami metod iteracyjnych są niepewność zbiegania do rozwiązania optymalnego, a także złożoność obliczeniowa, która wpływa negatywnie na czas działania.

Metoda algebraiczna zaproponowana w pracy [1], w której pochylono się nad algebraicznym rozwiązaniem układów równań w systemie GPS, ma znaczącą przewagę nad metodami iteracyjnymi otrzymywania dwóch możliwych rozwiązań natychmiastowo, z których można następnie wybrać bardziej odpowiadające. Dodatkowo, ta metoda daje możliwość operowania na pomiarach z arbitralnej liczby punktów, w przeciwieństwie do metody zaproponowanej w pracy [2], w której skupiono się na implementacji modelu VHDL pozycjonującego na podstawie odczytów z czterech satelitów.

Wyżej wymienione metody, choć są wciąż używane, nie pojawiają się w publikacjach z ostatnich lat. Większość rozwiązań prezentowanych w ostatnich dekadach opiera się na układzie równań liniowych z dodatkową zmienną ze względu na nieliniowość problemu. Pierwsze tego typu rozwiązanie zaprezentowano w pracy [3]. Zmienna nieliniowa była tam niwelowana przez odejmowanie od siebie kolejnych równań. Inny wariant przedstawiono w czasopiśmie [4], który zredagowano w czytelniejszej formie w pracy [9]. Właśnie ta publikacja była inspiracją do niniejszej pracy. Niedawno opublikowany przegląd tematu [11] zawiera szeroki opis innych metod wykorzystywanych do rozwiązania omawianego tutaj zagadnienia.

## 1.2. Obrane rozwiązanie

Wytypowano rozwiązanie przedstawione w publikacji [9]. Przekształćmy wyjściowy układ równań 1.1 do postaci

$$\begin{cases} (x^2 + y^2 + z^2) - 2x_1x - 2y_1y - 2z_1z = d_1^2 - x_1^2 - y_1^2 - z_1^2 \\ (x^2 + y^2 + z^2) - 2x_2x - 2y_2y - 2z_2z = d_2^2 - x_2^2 - y_2^2 - z_2^2 \\ \vdots \\ (x^2 + y^2 + z^2) - 2x_nx - 2y_ny - 2z_nz = d_n^2 - x_n^2 - y_n^2 - z_n^2, \end{cases} \quad (1.2)$$

a następnie do reprezentacji macierzowej

$$\begin{bmatrix} 1 & -2x_1 & -2y_1 & -2z_1 \\ 1 & -2x_2 & -2y_2 & -2z_2 \\ & & \vdots & \\ 1 & -2x_n & -2y_n & -2z_n \end{bmatrix} \begin{bmatrix} x^2 + y^2 + z^2 \\ x \\ y \\ z \end{bmatrix} = \begin{bmatrix} d_1^2 - x_1^2 - y_1^2 - z_1^2 \\ d_2^2 - x_2^2 - y_2^2 - z_2^2 \\ \vdots \\ d_n^2 - x_n^2 - y_n^2 - z_n^2 \end{bmatrix}, \quad (1.3)$$

którą można przedstawić jako

$$\mathbf{A} \cdot \mathbf{x} = \mathbf{b}, \quad (1.4)$$

gdzie

$$\mathbf{A} = \begin{bmatrix} 1 & -2x_1 & -2y_1 & -2z_1 \\ 1 & -2x_2 & -2y_2 & -2z_2 \\ & & \vdots & \\ 1 & -2x_n & -2y_n & -2z_n \end{bmatrix}, \quad (1.5)$$

$$\mathbf{x} = \begin{bmatrix} x^2 + y^2 + z^2 \\ x \\ y \\ z \end{bmatrix}, \quad (1.6)$$

$$\mathbf{b} = \begin{bmatrix} d_1^2 - x_1^2 - y_1^2 - z_1^2 \\ d_2^2 - x_2^2 - y_2^2 - z_2^2 \\ \vdots \\ d_n^2 - x_n^2 - y_n^2 - z_n^2 \end{bmatrix}, \quad (1.7)$$

przy założeniu, że  $\mathbf{x} \in E$ , gdzie

$$E = \left\{ (x_0, x_1, x_2, x_3)^T \in \mathbb{R}^4 \mid x_0 = x_1^2 + x_2^2 + x_3^2 \right\}. \quad (1.8)$$

### Liniowa metoda najmniejszych kwadratów

Aby rozwiązać powyższe równanie liniowe obrano liniową metodę najmniejszych kwadratów. Z definicji rozwiązaniem równania

$$\mathbf{A} \cdot \mathbf{x} = \mathbf{b}, \quad (1.9)$$

gdzie  $\mathbf{A} \in \mathbb{R}^{n \times m}$ ,  $\mathbf{b} \in \mathbb{R}^n$  są znane, będzie taki  $\hat{\mathbf{x}} \in \mathbb{R}^m$ , który minimalizuje

$$\sum_{i=1}^n \left| \sum_{j=1}^m A_{ij} x_j - b_i \right|^2 = \|\mathbf{A}\hat{\mathbf{x}} - \mathbf{b}\|^2. \quad (1.10)$$

Taki  $\hat{\mathbf{x}}$  obliczamy przy użyciu powszechnie znanego w kontekście najmniejszych kwadratów *równania normalnego* postaci

$$\mathbf{A}^T \mathbf{A} \hat{\mathbf{x}} = \mathbf{A}^T \mathbf{b} \quad (1.11)$$

lub po przekształceniu

$$\hat{\mathbf{x}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}. \quad (1.12)$$

Jeśli żadna trójka odbiorników nie jest współliniowa ( $\text{rank } \mathbf{A} = m$ ) oraz pomiarów jest więcej niż wymiarów ( $n > m$ ) możemy zagwarantować odwracalność  $\mathbf{A}$ , a co za tym idzie istnienie  $\hat{\mathbf{x}}$ .

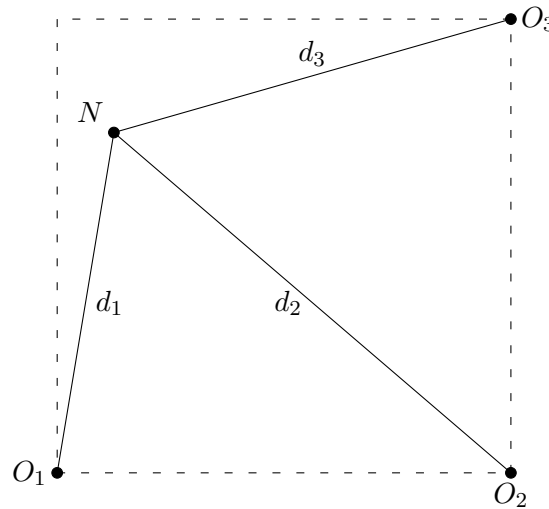
### Forma ogólna

Zaletą reprezentacji problemu 1.4 zawartą w obranym rozwiązaniu jest łatwość rozszerzenia obliczeń na dowolną  $m$ -wymiarową przestrzeń oraz  $n$ -odbiorników:

$$\begin{bmatrix} 1 & -2\mathbf{x}^{(1)} \\ 1 & -2\mathbf{x}^{(2)} \\ \vdots & \vdots \\ 1 & -2\mathbf{x}^{(n)} \end{bmatrix} \begin{bmatrix} \sum_{i=1}^m x_i^2 \\ x_1 \\ \vdots \\ x_m \end{bmatrix} = \begin{bmatrix} d_1^2 - \sum_{i=1}^m x_i^{(1)2} \\ d_2^2 - \sum_{i=1}^m x_i^{(2)2} \\ \vdots \\ d_n^2 - \sum_{i=1}^m x_i^{(n)2} \end{bmatrix} \quad (1.13)$$

z analogicznym do 1.8 ograniczeniem  $\mathbf{x} \in E$ .

### 1.3. Przykład problemu



Rys. 1.1: Egzemplarz problemu multilateracji

Staramy się ustalić pozycję punktu  $N$  na podstawie współrzędnych punktów  $O_1, O_2, O_3$  oraz odległości  $d_1, d_2, d_3$ . Wykorzystując równanie 1.12, podstawiamy znane wartości, otrzymując  $\mathbf{x}$  zawierający szukane współrzędne.

Jak łatwo zauważyć, pierwsza część lewej strony równości zależy jedynie od macierzy  $\mathbf{A}$ , która nie zawiera zmiennych związanych z węzłem źródłowym, co pozwala, przy założeniu niezmienności pozycji odbiorników, na wcześniejsze i jednorazowe przeprowadzenie kosztownych przekształceń tej macierzy. W tej sytuacji jednorazowe obliczenie rozwiązania wymaga jedynie obliczenia wektora  $\mathbf{b}$  oraz iloczynu skalarnego.

Przyjmijmy wartości dla odbiorników  $O_1(0; 0), O_2(1; 0), O_3(1; 1)$  oraz założmy, że punkt  $N$  znajduje się na pozycji  $(0, 125; 0, 75)$ . Odległości  $d_1, d_2, d_3$  wynoszą odpowiednio  $\sqrt{0, 578125}, \sqrt{1, 328125}$  i  $\sqrt{0, 828125}$ . Wprowadźmy teraz błąd pomiaru poprzez zaokrąglenie długości do 0, 05. Nasze odległości  $d'_1, d'_2, d'_3$  wynoszą 0, 75; 1, 15 oraz 0, 9. Dla tych danych macierz  $\mathbf{A}$  ma postać

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & -2 & 0 \\ 1 & -2 & -2 \end{bmatrix},$$

a wektor  $\mathbf{b}$

$$\mathbf{b} = \begin{bmatrix} 0, 5625 \\ 0, 3225 \\ -1, 19 \end{bmatrix}.$$

Idąc dalej prawa część równania 1.12 przyjmie postać

$$(\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} = \left( \begin{bmatrix} 1 & 1 & 1 \\ 0 & -2 & -2 \\ 0 & 0 & -2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0,5 & -0,5 & 0 \\ 0 & 0,5 & -0,5 \end{bmatrix} \right)^{-1} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 0 & -2 & -2 \\ 0 & 0 & -2 \end{bmatrix} \cdot \begin{bmatrix} 0,5625 \\ 0,3225 \\ -1,19 \end{bmatrix},$$

co daje nam rozwiązanie aproksymacyjne

$$\hat{\mathbf{x}} = \begin{bmatrix} 0,5625 \\ 0,12 \\ 0,75625 \end{bmatrix}.$$

Z niego odczytujemy współrzędne odbiornika  $(0,12; 0,75625)$ , które są bliskie rzeczywistej pozycji  $N$ .

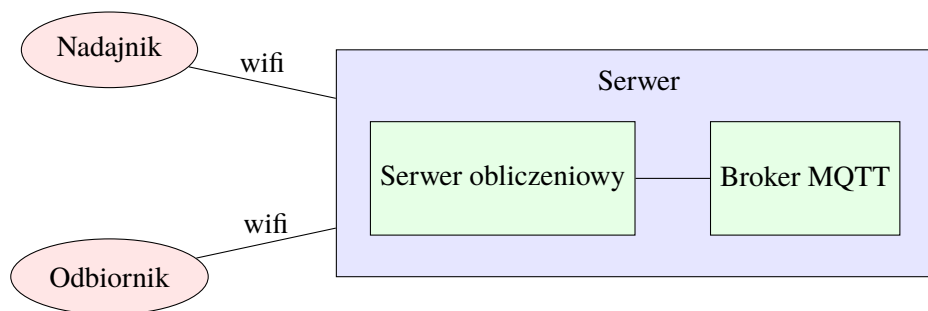


## Rozdział 2

# Sprzęt systemowy

Niemal wszystkie prace odnoszące się do tematu multilateracji opierają się na systemach urządzeń działających w zakresie fal elektromagnetycznych. W tej pracy poświęcimy uwagę systemowi działającemu w domenie fal dźwiękowych i sprawdzimy, jak ten aspekt wpływa na skuteczność i dokładność rozwiązania problemu multilateracji.

Na poniższym rysunku przedstawiono schematycznie elementy systemu testowego. Składa się on z kilku głównych elementów, które pozwolą wygenerować dane testowe i sprawdzić skuteczność metod przedstawionych w poprzednim rozdziale.



Rys. 2.1: Topologia systemu

### 2.1. Broker MQTT

Urządzenia systemowe porozumiewają się przy użyciu protokołu MQTT. Każdy z węzłów oraz serwer łączą się z centralnym brokerem, który przekierowuje wiadomości opatrzone tematem do klientów subskrybujących dany temat. Broker jest oparty o otwarty projekt *Mosquitto*.

### 2.2. Serwer obliczeniowy

Centralnym urządzeniem systemu jest serwer obliczeniowy kumulujący dane otrzymane z sensorów do rozwiązania problemu multilateracji. Do implementacji serwera zdecydowano się użyć języka *python* ze względu na dostępność bibliotek oferujących narzędzia potrzebne do działania serwera, takie jak pakiet *paho* oferujący klienta MQTT czy pakiet matematyczny *numpy*. Ponadto prostota składni tego języka pozwoliła na szybkie wdrażanie modyfikacji działania kolejnych aspektów serwera. Serwer hostowany jest na tej samej maszynie co broker MQTT.

### 2.3. Węzły

Każdy z węzłów oparty jest o mikrokontroler ESP8266-01s zaprogramowany przy użyciu Arduino IDE w języku C++. W systemie występują dwa rodzaje węzłów:

- nadajnik,

- odbiornik.

Nadajnik wyposażony jest w przełącznik cewkowy sterowany przez mikrokontroler, który służy do kontrolowania brzęczyka zasilanego napięciem  $12V$ . Wybrano brzęczyk o głośności  $90dB$  w celu zmaksymalizowania zasięgu działania systemu. Odbiornik jest natomiast wyposażony w mikrofon elektretowy, którego sygnał wzmacniany jest przez wzmacniacz operacyjny. Sygnał analogowy jest ostatecznie zmieniany na sygnał binarny na podstawie odniesienia do określonego poziomu napięcia. Czułość mikrofonu dostrajana jest ręcznie poprzez potencjometr.

W systemie wykorzystano jeden nadajnik oraz zmienną liczbę odbiorników. Podstawą przeprowadzonych eksperymentów była próba określenia pozycji nadajnika względem odbiorników wyłącznie przy wykorzystaniu sygnałów dźwiękowych (impulsów) nadajnika.

## Rozdział 3

# Eksperyment zerowy

Po przygotowaniu komponentów systemu, wstępnie zaimplementowano program rozwiązujący zagadnienie multilateracji, aby zbadać czy problem nie jest zbyt trywialny, aby opisać go w pracy, lub przeciwnym razie, na podstawie wyników eksperymentu, zastanowić się jakie przeszkody stoją na drodze do rozwiązania o zadowalającej precyzji.

### 3.1. Opis działania

Programy dla urządzeń w systemie zaimplementowano na podstawie rozwiązania aproksymacyjnego 1.12 postaci

$$\hat{\mathbf{x}} = \left(\mathbf{A}^T \mathbf{A}\right)^{-1} \mathbf{A}^T \mathbf{b} \quad (3.1)$$

Poniżej przedstawiono je w formie pseudokodów. Współrzędne odbiorników, a zatem macierz  $\mathbf{A}$  są znane i stałe dla każdej konfiguracji ustawienia odbiorników.

#### Programy węzłów

---

##### Program 1 Program nadajnika

---

```
1: buzz ← False
2: buzzTime ← 0
3: lastBuzzTime ← 0
4: syncTime ← 0
5: function ONMESSAGE(message, topic)
6:   if topic is TOPIC then
7:     buzz ← message
8:   end if
9:   if topic is TIME_TOPIC then
10:    syncTime ← micros()
11:   end if
12: end function
13: loop
14:   if buzz and micros() − lastBuzzTime > BUZZ_INTERVAL then
15:     buzzTime ← micros() − syncTime
16:     publish(buzzTime)
17:     lastBuzzTime ← buzzTime
18:     buzzer()
19:   end if
20: end loop
```

---

**Program 2** Program odbiornika

---

```

1: micState  $\leftarrow$  LOW
2: micTime  $\leftarrow$  False
3: lastMicTime  $\leftarrow$  0
4: syncTime  $\leftarrow$  0
5: function ONMESSAGE(topic)
6:   if topic is TIME_TOPIC then
7:     syncTime  $\leftarrow$  micros()
8:   end if
9: end function
10: function MICINTERRUPT
11:   micState  $\leftarrow$  HIGH
12: end function
13: loop
14:   if micState is HIGH and micros() – lastMicTime > MIC_INTERVAL then
15:     micTime  $\leftarrow$  micros() – syncTime
16:     publish(micTime)
17:     lastMicTime  $\leftarrow$  micTime
18:     micState  $\leftarrow$  LOW
19:   end if
20: end loop

```

---

- *micros*() - funkcja zwracająca liczbę mikrosekund od uruchomienia urządzenia,
- *buzzer*() - funkcja nadająca sygnał dźwiękowy przy użyciu brzęczyka,
- *TOPIC* - nazwa kanału MQTT korespondującego do danego węzła,
- *TIME\_TOPIC* - nazwa ogólnego kanału MQTT służącego do przesyłania wiadomości z poleceniem synchronizacji czasu,
- *BUZZ\_INTERVAL* - interwał czasowy pomiędzy sygnałami brzęczyka,
- *MIC\_INTERVAL* - interwał czasowy pomiędzy dwoma oddzielnymi rejestracjami wysokiego poziomu sygnału na mikrofonie,
- ONMESSAGE - funkcja wywołania zwrotnego wywoływana przy otrzymaniu wiadomości na dowolnym z subskrybowanych tematów,
- *publish*() - funkcja publikująca wiadomość na zadanym kanale MQTT.

W celu zminimalizowania opóźnień funkcja MICINTERRUPT została zaimplementowana przy użyciu przerwania sprzętowego. Jest wywoływana przy zmianie stanu pinu (do którego podłączony jest mikrofon) z niskiego na wysoki.

**Program serwera****Program 3** Program serwera

---

```

1:  $M \leftarrow (A^T A)^{-1} A^T$ 
2: function calc_position
3:   for node in NODES do
4:      $d \leftarrow (SS/10^6) \cdot time_{node} - time_{source}$ 
5:      $b_{node} \leftarrow d - \sum_{x \in coords_{node}} x^2$ 
6:   end for
7:   return  $M \cdot b$ 
8: end function
9: loop main
10:   sleep( $t_1$ )
11:   calc_position()
12: end loop

```

---

```

13: loop sync_clock
14:   publish(TIME_TOPIC)
15:   sleep( $t_2$ )
16: end loop

```

- $A$  - macierz zawierająca współrzędne obrotów z równania 1.4
- $SS$  - prędkość dźwięku  $[\frac{m}{s}]$ ,
- $time$  - czas dostarczony w ostatniej wiadomości od węzła,
- $coords$  - współrzędne węzłów nadawczych,
- $t_1$  - czas pomiędzy obliczeniami pozycji,
- $t_2$  - czas pomiędzy wiadomościami synchronizującymi zegary.

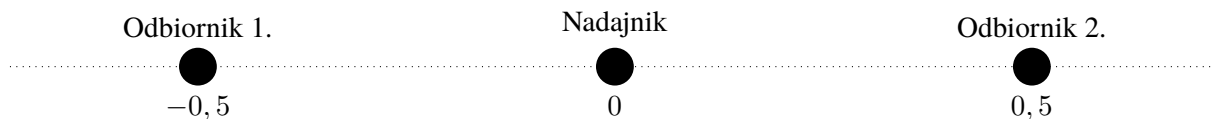
### Opis algorytmu

W programie serwera zaimplementowano funkcję rozwiązującą równanie 1.12 przy pomocy funkcji bibliotecznych dostępnych w pakiecie *numpy*. Wywołania następują co interwał  $t_1$ . Co interwał  $t_2$  serwer wysyła wiadomość synchronizującą z założeniem, że węzły odbierają ją równocześnie i w ten sposób synchronizują zegary.

## 3.2. Ewaluacja działania systemu

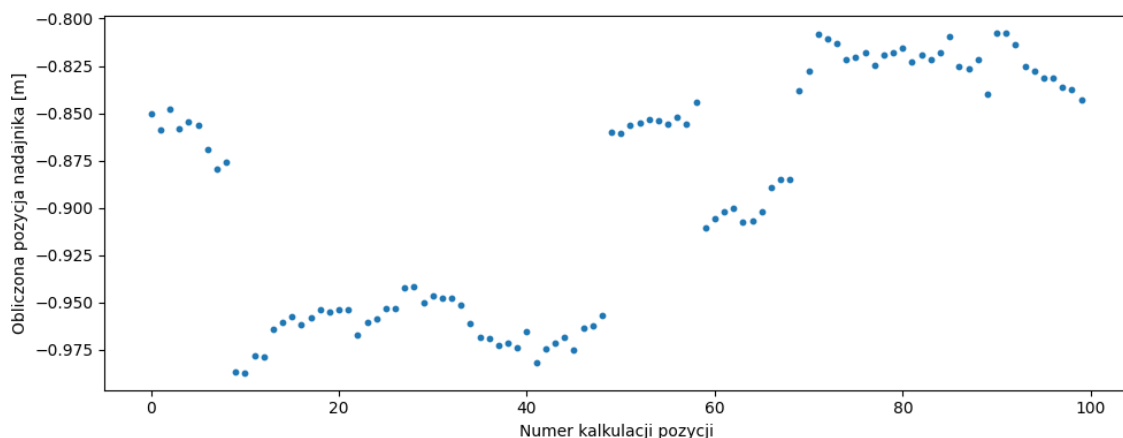
Pierwsze testy przeprowadzone zostały na systemie operującym w jednym wymiarze (to jest wszystkie węzły są współliniowe) w celu jak największego uproszczenia, pozwalającego na szybsze wykrywanie i rektyfikację błędów. Wszystkie odległości będą podawane w metrach.

### Struktura systemu testowego

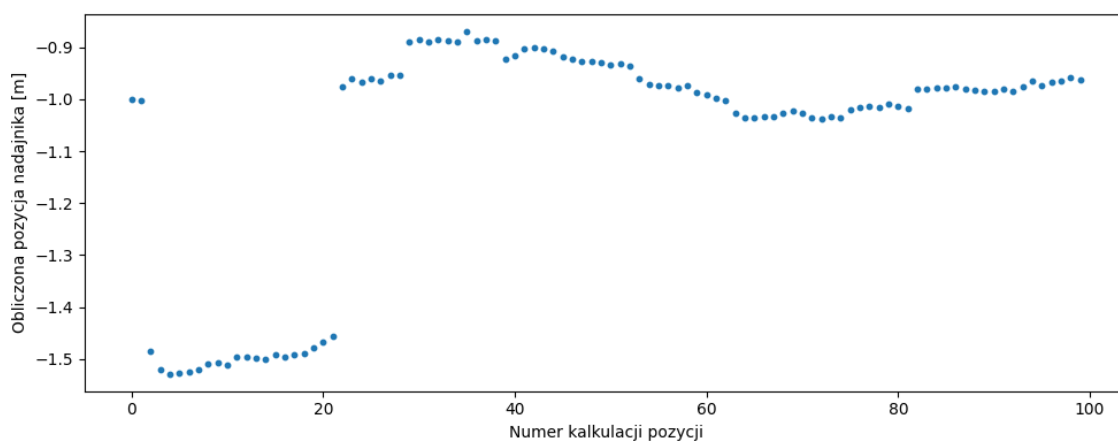


Rys. 3.1: Układ systemu testowego

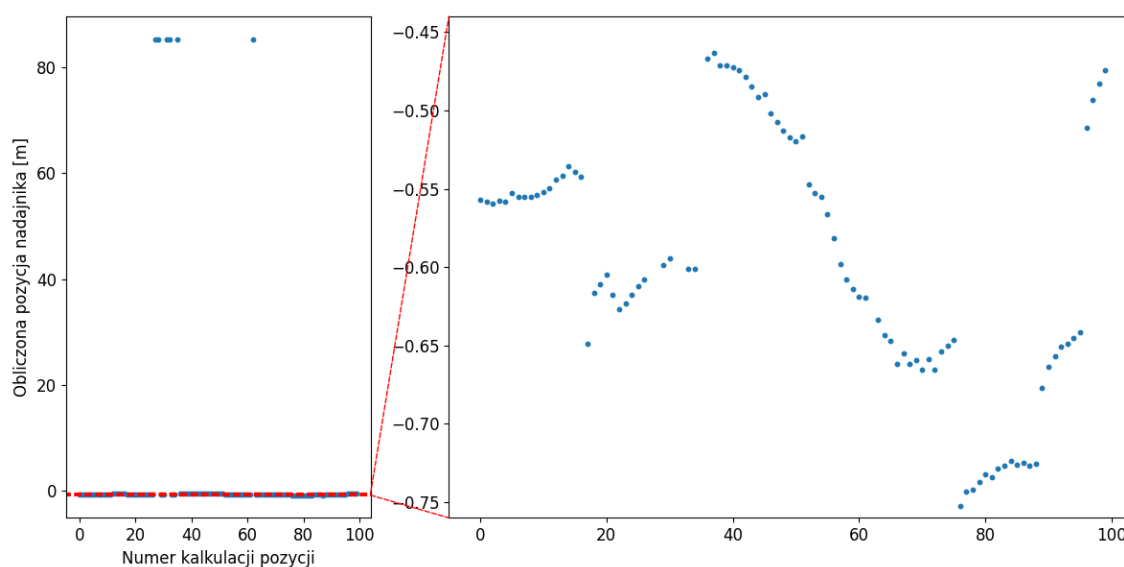
Nadajnik umiejscowiono w punkcie (0), natomiast dwa odbiorniki w punktach odpowiednio  $(-0,5)$  i  $(0,5)$ , wszystkie węzły były stacjonarne. Nadajnik co  $0,5$  s nadawał sygnał o długości  $10$  ms, a serwer co  $0,5$  s zwracał wynik zagadnienia multilateracji na podstawie ostatnio otrzymanych danych. Wykonano 5 następujących bezpośrednio po sobie eksperymentów. Na poniższych wykresach (3.2) przedstawiono ich wyniki.



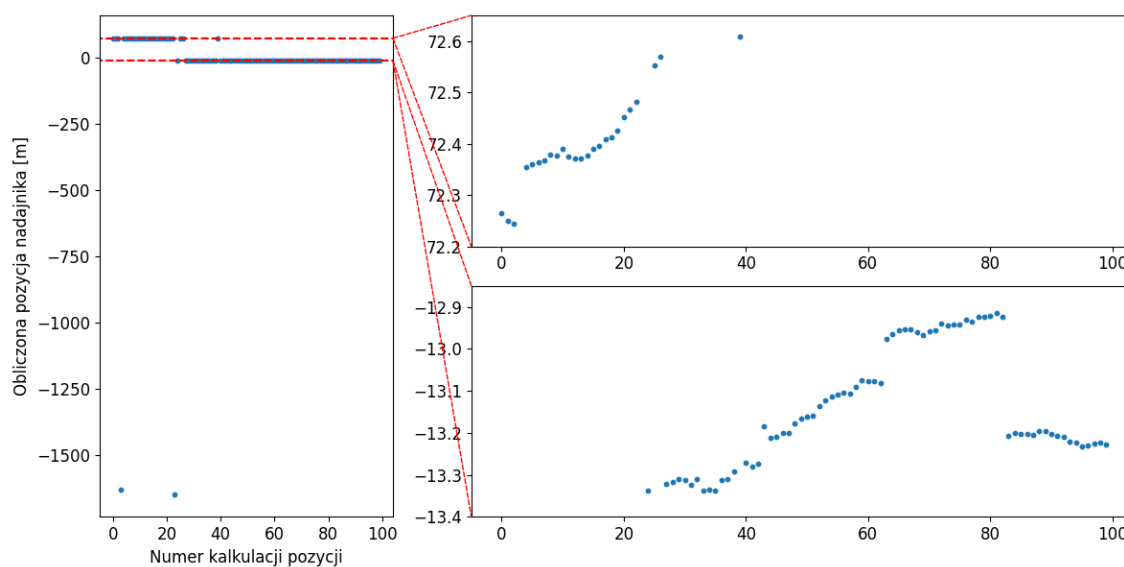
(a) Test 1.



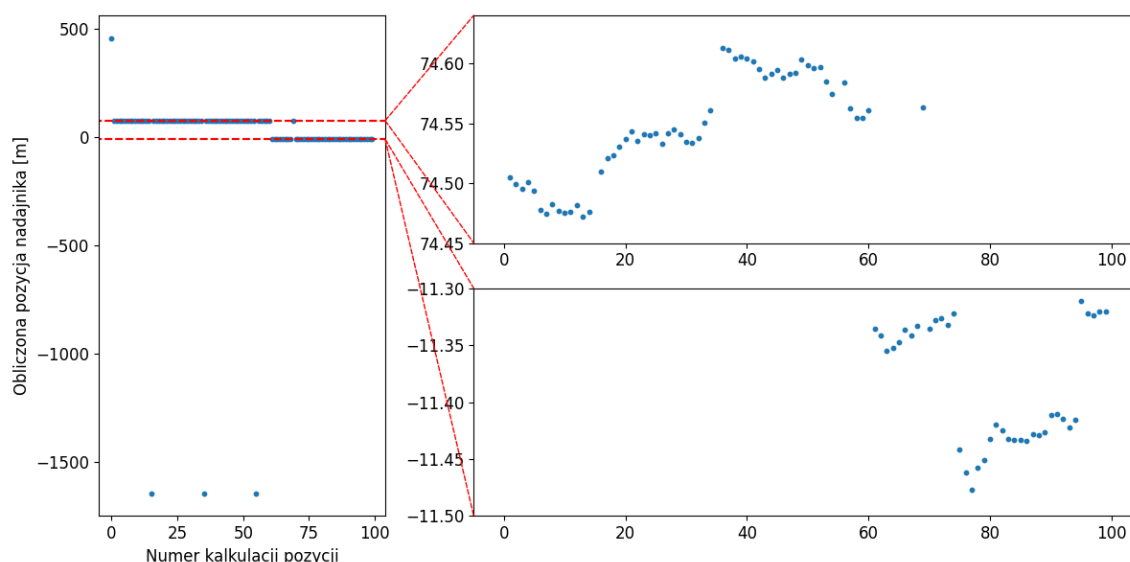
(b) Test 2.



(c) Test 3. Diagram po prawej stanowi powiększenie zaznaczonego obszaru diagramu po lewej.



(d) Test 4. Diagramy po prawej stanowią powiększenie odpowiednich obszarów diagramu po lewej.



(e) Test 5. Diagramy po prawej stanowią powiększenie odpowiednich obszarów diagramu po lewej.

Rys. 3.2: Wykres obliczonej pozycji odbiornika w zależności od czasu

### 3.3. Interpretacja wyników i wnioski

Już na pierwszym z wykresów widać, że wartości wyjściowe algorytmu są niestabilne, współrzędna nadajnika waha się w przedziale  $(-0,98; -0,8)$ , a także co jakiś czas następują nagłe przeskoki między następującymi po sobie wartościami. Coraz większa rozbieżność następuje wraz z upływem czasu. Już w drugim teście różnica ekstremalnych odchyłeń jest większa niż odległość między nadajnikami, a dokładność jest absolutnie niezadowalająca. Kolejne testy jedynie utwierdzają te obserwacje.

Na niską dokładność i stabilność wyników może mieć wpływ wiele czynników takich jak:

- niepoprawna synchronizacja zegarów węzłów,
- niepoprawna kalibracja czułości mikrofonów,
- fałszywe odczyty wynikające z odbić fali dźwiękowej,
- niska jakość rozwiązania aproksymacyjnego.

Obserwując skoki pomiędzy sąsiadującymi obserwacjami położenia, rozdzielającymi okresy względnej stabilności i mając na uwadze statyczność otoczenia systemu, możemy śmiało wysnuć wniosek, że tak nagła zmiana średniej obliczanej wartości może wynikać najprawdopodobniej z błędnej synchronizacji czasu. Wygląda na to, że sygnał wysyłany przez serwer obliczeniowy nie jest odbierany we wszystkich węzłach równocześnie.

Następny rozdział będzie poświęcony badaniu i tworzeniu algorytmu synchronizacji zegarów w węzłach.

## Rozdział 4

# Synchronizacja czasu

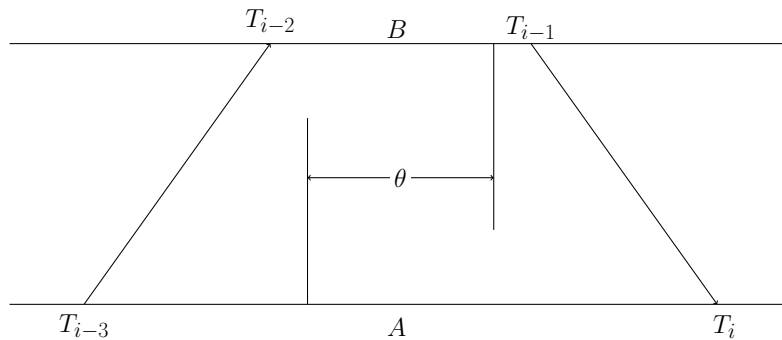
Synchronizacja czasu jest jednym z podstawowych problemów systemów opartych o wiele urządzeń posiadających własne zegary. Wiedza o dokładnym czasie zachodzących w sieci zdarzeń jest niezbędna do szybkiego przesyłu danych, koordynacji procesów czy aktualizacji systemu plików. W przypadku niniejszej pracy dokładne określenie czasu zachodzących zdarzeń jest kluczowe do wiarygodnego określenia odległości pomiędzy węzłami na podstawie interwału czasowego pomiędzy nadaniem a odbiorem sygnału dźwiękowego.

### 4.1. Synchronizacja programowa

Pierwszym podejściem do rozwiązania problemu synchronizacji zegarów było zastosowanie synchronizacji programowej [12], w której urządzenie-host utrzymuje wysokiej rozdzielczości licznik czasowy i wysyła sygnały o jego wartości pozostałym węzłom w sieci. Na podstawie tych informacji każdy z węzłów oblicza różnicę w zegarach i wprowadza odpowiednie przesunięcie własnego licznika.

#### Algorytm synchronizacji NTP

Jednym z najbardziej rozpowszechnionych protokołów synchronizacji programowej jest *Network Time Protocol* opisany w pracy [8]. Zachowując oznaczenia w niej zastosowane opiszmy w skrócie zasadę działania tego protokołu.



Rys. 4.1: Pomiar opóźnienia transmisji i przesunięcia zegara

Na rysunku 4.1 przedstawiono schemat działania protokołu NTP. Urządzenia A i B wymieniają wiadomości zawierające sygnatury czasowe. Niech  $T_i$ ,  $T_{i-1}$ ,  $T_{i-2}$ ,  $T_{i-3}$  będą czterema ostatnimi wiadomościami oraz niech  $a = T_{i-2} - T_{i-3}$  oraz  $b = T_{i-1} - T_i$ . Wtedy całkowity czas transmisji  $\delta_i$  i przesunięcie zegara  $\theta_i$  urządzenia B względem urządzenia A to

$$\delta_i = a - b \quad \text{oraz} \quad \theta_i = \frac{a + b}{2}.$$

Dodatkowym wnioskiem przedstawionym w powyższej pracy jest własność prawdziwego przesunięcia względem aktualnie obliczonego:

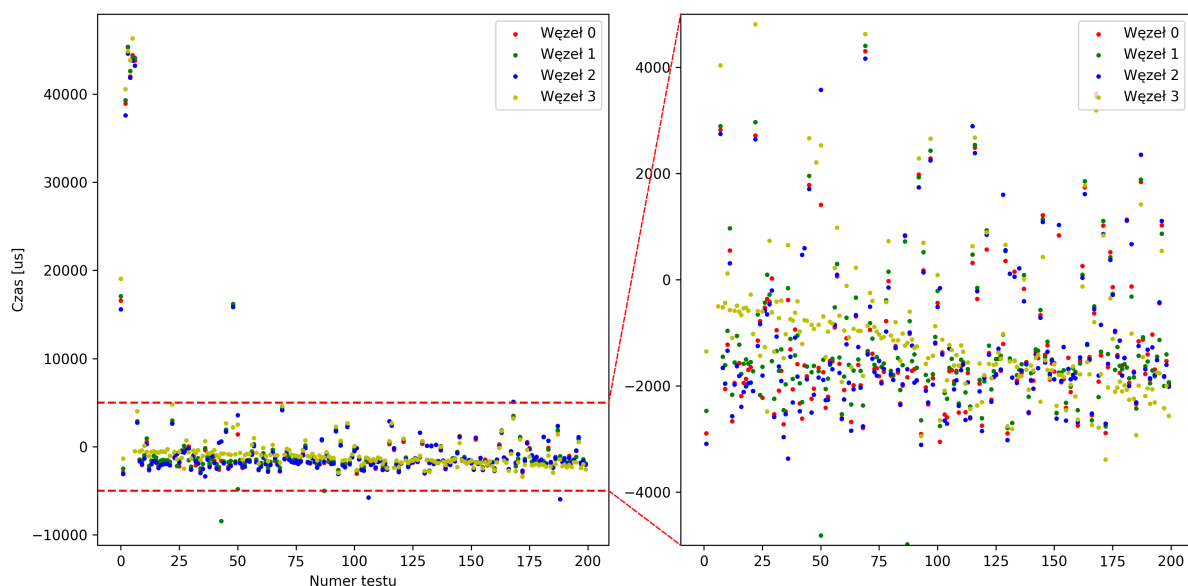


$$\theta_i - \frac{\delta_i}{2} \leq \theta \leq \theta_i + \frac{\delta_i}{2}.$$

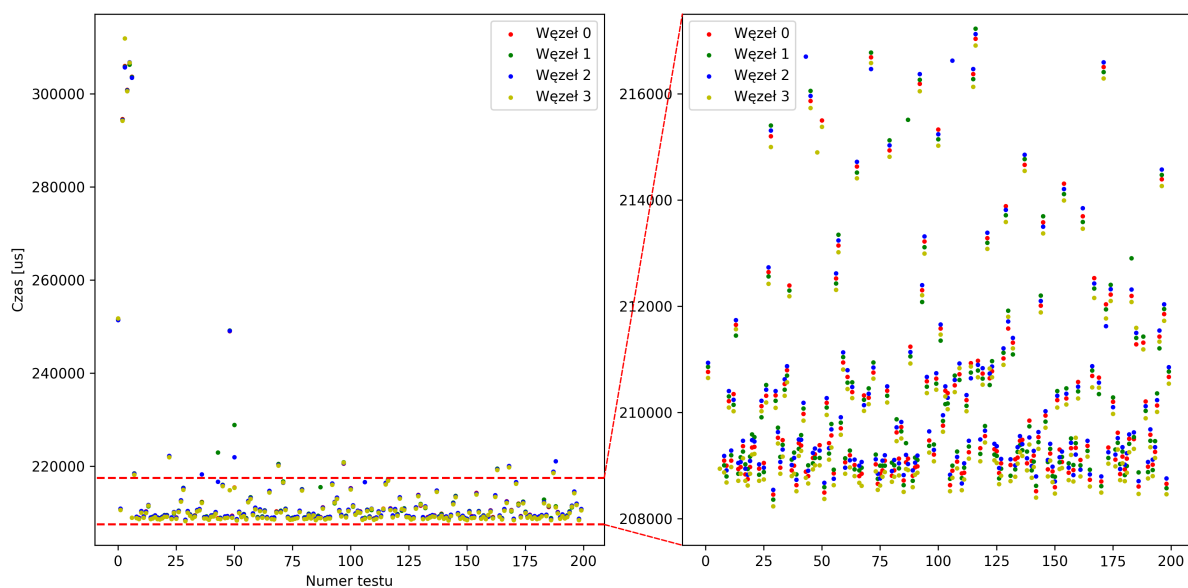
Jak łatwo zauważyć, im krótszy jest czas propagacji, tym lepsze przybliżenie dostajemy.

Na podstawie tych informacji podjęto próbę synchronizacji programowej zegarów węzłów w systemie multilateracyjnym. Przy użyciu czterech węzłów przeprowadzono eksperymenty mające na celu próbę ustalenia przesunięć zegarów każdego z urządzeń względem centralnego serwera. Opisany wyżej schemat wymiany czterech wiadomości powtarzano  $n$  razy, wspólnie dla wszystkich węzłów, zapisując obliczone przesunięcia oraz czasy propagacji.

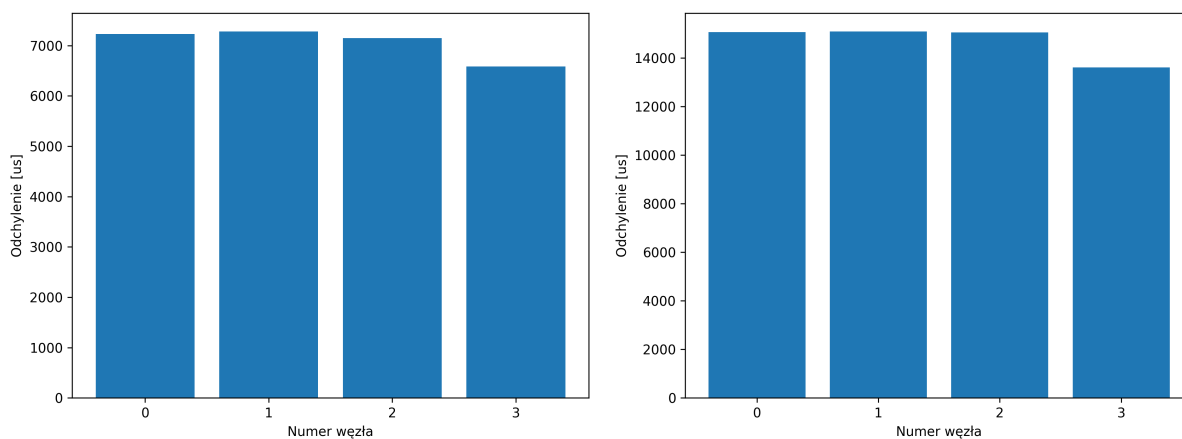
W celu poprawy czytelności, wyniki przesunięć zostały znormalizowane poprzez przesunięcie o wartość średnią dla każdego z węzłów.



Rys. 4.2: Wyniki pomiarów przesunięć zegarów. Diagram po prawej stanowi powiększenie zaznaczonego obszaru diagramu po lewej.



Rys. 4.3: Wyniki pomiarów czasu propagacji. Diagram po prawej stanowi powiększenie zaznaczonego obszaru diagramu po lewej.



(a) Odchylenia standardowe przesunięć zegarów

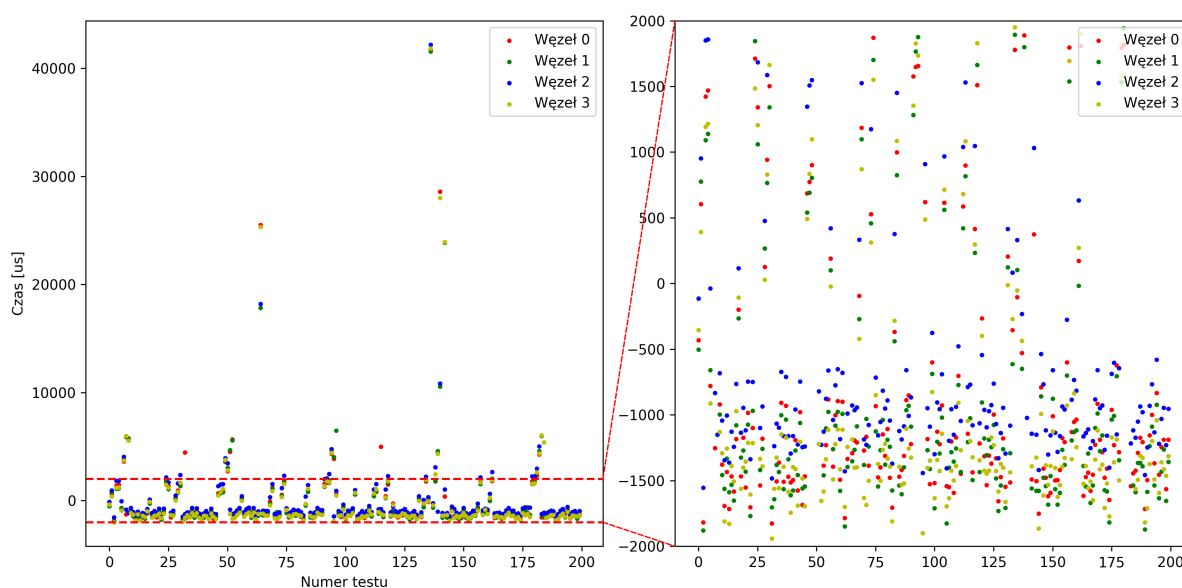
(b) Odchylenia standardowe czasu propagacji

Rys. 4.4: Odchylenia standardowe pomiarów

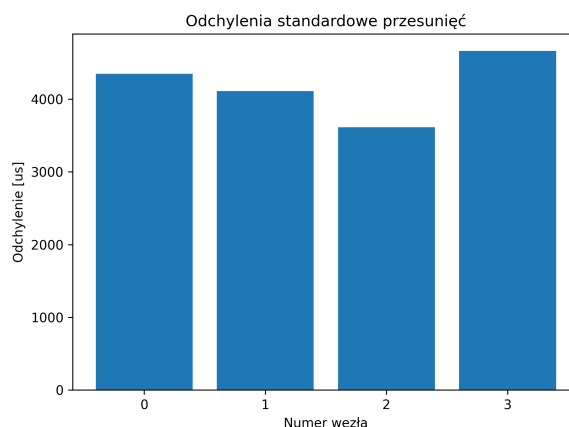
Na podstawie otrzymanych wykresów łatwo zauważyć, że czas propagacji wiadomości w systemie jest nieprzewidywalny i zmienia się znacząco z pomiaru na pomiar. Interesującą nas statystyką są jednak zaobserwowane przesunięcia zegarów, bez których nie jesteśmy w stanie poprawnie ocenić odległości od źródła dźwięku. Tutaj wartości wyglądają na bardziej skoncentrowane, jednak nie widać wyraźnych tendencji koncentracji wokół wartości średnich dla żadnego z węzłów. Ponadto zaobserwowane odchylenia standardowe  $\sigma_i$ ,  $i \in \{0, 1, 2, 3\}$  wielkości  $\approx 7000 \mu s$  są nieakceptowalne, ponieważ w takim czasie dźwięk w powietrzu pokonuje  $\frac{7000}{1000000} s \cdot 343 \frac{m}{s} = 2,401 m$ . Możliwe jest, że wielokrotne powtarzanie pomiarów da zadowalającą wartość średnią, pozwalającą na centymetrową precyzję obliczanych odległości. W porównaniu z pozostałymi sposobami, brane będą pod uwagę wartości uśrednione.

### Bezpośredni pomiar przesunięć zegarów

Biorąc pod uwagę zauważoną nieprzewidywalność i rozrzut czasów propagacji (spowodowanych najprawdopodobniej użyciem protokołu MQTT do przesyłu wiadomości pomiędzy urządzeniami) następnym pomysłem schematu synchronizacji jest bezpośrednie badanie względnego przesunięcia zegarów. Węzeł wysyła  $n$  wiadomości zawierających aktualną wartość zegara, która po odebraniu przez serwer jest porównywana z zegarem w nim dostępnym.



Rys. 4.5: Wyniki pomiarów przesunięć zegarów. Diagram po prawej stanowi powiększenie zaznaczonego obszaru diagramu po lewej.



Rys. 4.6: Odchylenia standardowe pomiarów

Wykresy przesunięć wygenerowane na podstawie tych testów na pierwszy rzut oka są skoncentrowane podobnie jak poprzednie, jednakże odchylenie standardowe są prawie dwukrotnie mniejsze niż uprzednio, co daje nadzieję na bardziej wiarygodne wyniki. W porównaniu z pozostałymi sposobami, brane będą pod uwagę wartości uśrednione.

## 4.2. Synchronizacja sprzętowa

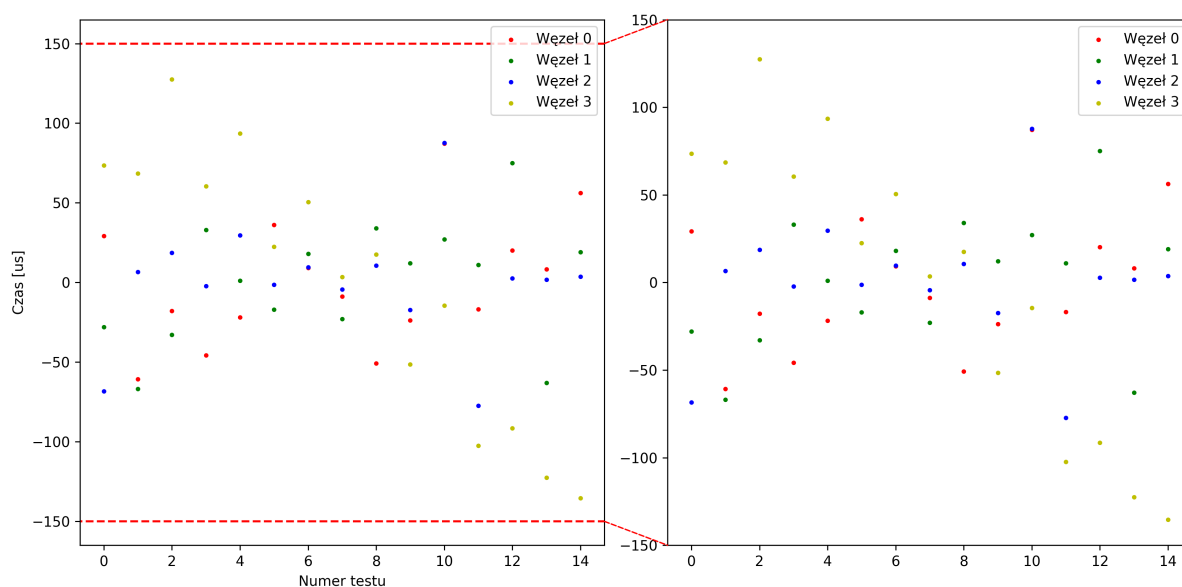
Alternatywną metodą do synchronizacji programowej jest synchronizacja sprzętowa. Ze względu na konieczność wyposażenia węzłów w odpowiednie sensory potrzebne do pełnienia tego zadania, opcja programowa jest często bardziej atrakcyjnym wyborem, jednakże w przypadku systemu proponowanego w tej pracy mikrofony, w które są wyposażone węzły odbiorcze (wraz z brzęczykiem węzła nadawczego) mogą pełnić tę rolę.

### Synchronizacja z użyciem mikrofonów

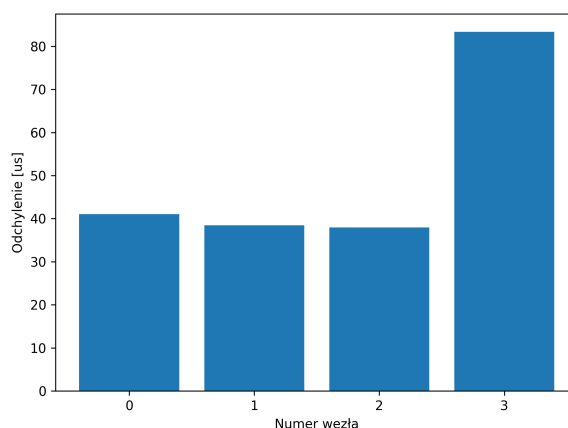
Jedyną potrzebną do obliczeń odległości informacją jest przesunięcie naszego zegara względem zegara w węźle nadawczym, dlatego wystarczającym będzie porównanie czasu nadania i odebrania sygnału dźwiękowego. Ponadto ten rodzaj synchronizacji, w przeciwieństwie do synchronizacji programowej, która uzgadniała ze sobą jedynie zegary na podstawie wymienianych wiadomości, wlicza w czas transmisji wszelkie nie wzięte wcześniej pod uwagę opóźnienia, takie jak:

- czas pomiędzy wysłaniem wiadomości o nadaniu sygnału a zamknięciem kontaktora i poruszeniem membraną brzęczyka,
- czas pomiędzy odebraniem sygnału przez mikrofon a zmianą stanu zmiennej na to wskazującej.

Przeprowadzono testy tego typu synchronizacji, których wyniki przedstawiono na wykresach 4.7. W celu zwiększenia czytelności odrzucono pierwszy z pomiarów oraz przesunięto wyniki o wartość średnią.



Rys. 4.7: Wyniki pomiarów przesunięć zegarów. Diagram po prawej stanowi powiększenie zaznaczonego obszaru diagramu po lewej.

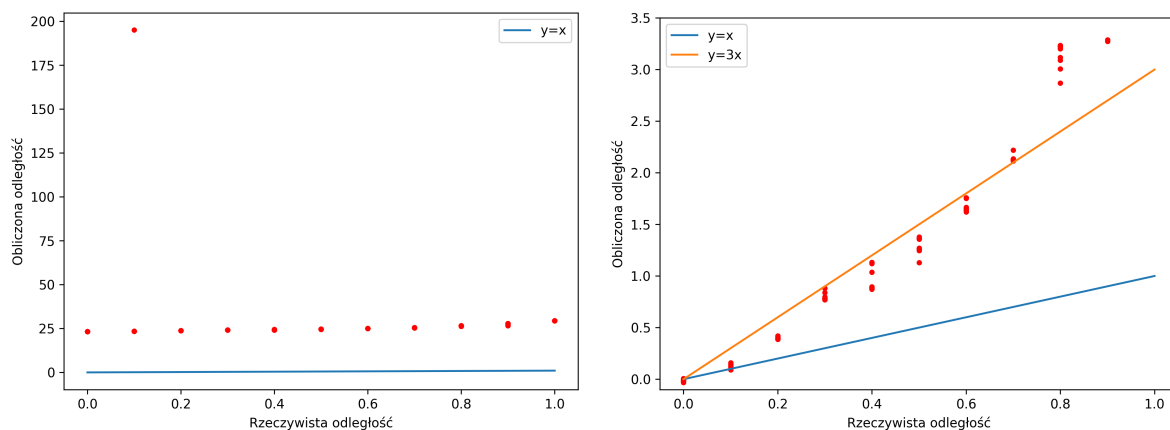


Rys. 4.8: Odchylenia standardowe pomiarów

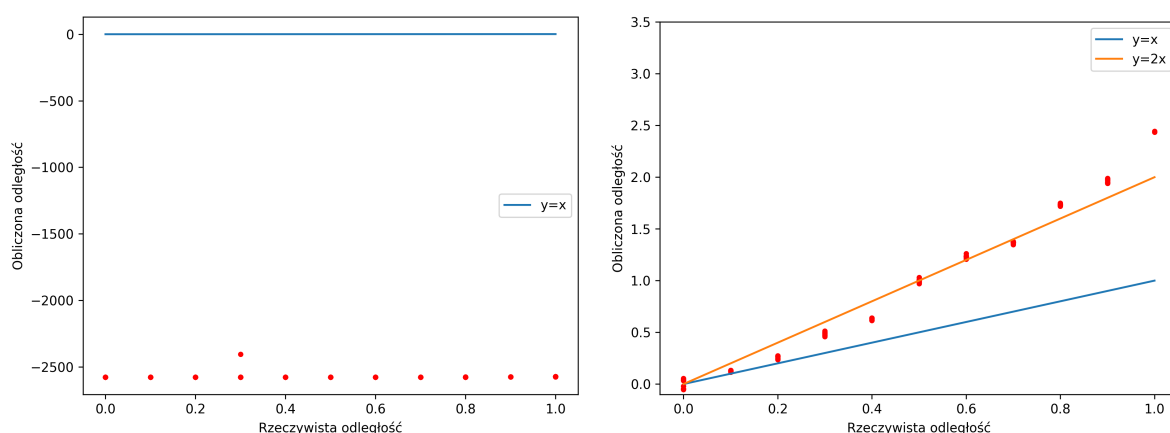
Łatwo zauważyć, że udało się zmniejszyć odchylenia obliczonych przesunięć zegara aż o dwa rzędy wielkości. Taka dokładność daje znacznie lepsze przybliżenie rzeczywistej odległości między węzłami, ponieważ w czasie  $40\mu s$  dźwięk pokona jedynie  $\frac{40}{10000000} s \cdot 343 \frac{m}{s} \approx 0,014m$ . Mając tak dokładne odległości, będziemy mogli wprowadzić je do modelu multilateracyjnego.

### Porównanie metod

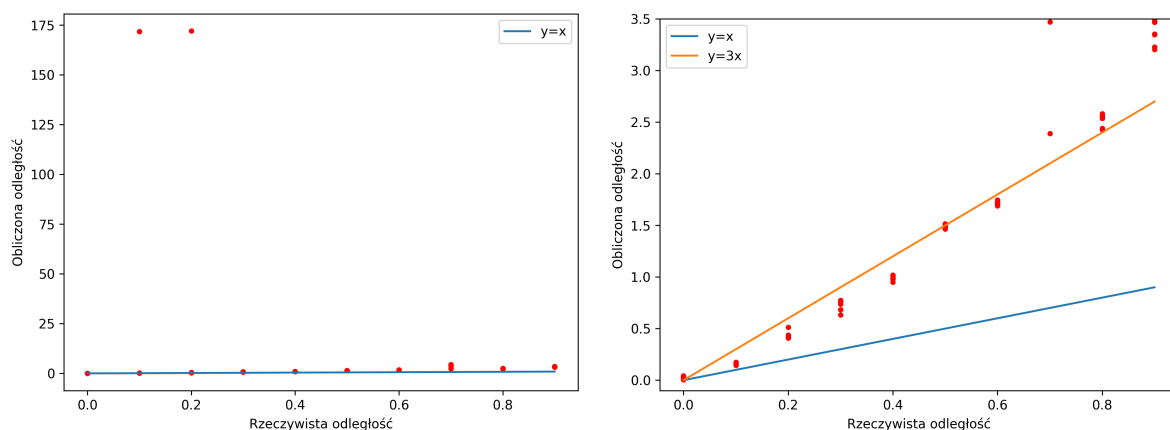
Porównajmy teraz dokładność obliczanych odległości na podstawie interwału czasowego pomiędzy nadaniem dźwięku a jego odbiorem w różnych wariantach synchronizacji zegarów. Wykresy uzyskane przy zastosowaniu synchronizacji czasowej zostały przedstawione przed i po znormalizowaniu poprzez przesunięcie tak, by średnia pomiarów rozpoczynała się od 0. Pomiary oparte o synchronizację z użyciem mikrofonów nie wymagały tego dodatkowego kroku.



(a) synchronizacja NTP 4.1



(b) synchronizacja pomiaru przesunięć 4.1



(c) synchronizacja mikrofonowa 4.2

Rys. 4.9: Wyniki pomiarów odległości z użyciem rozpatrywanych synchronizacji. Na prawych diagramach wrysowano proste, które w przybliżeniu odpowiadają współczynnikowi skalowania obliczonych odległości.

Na wykresach widać, że wszystkie trzy algorytmy synchronizacji po normalizacji dają wyniki o podobnej dokładności, adekwatnej do użycia w danych wejściowych multilateracji. Ciekawą jest natomiast obserwacja skalowania obliczonych odległości. Widzimy, że w przypadku 4.9a oraz 4.9c otrzymane wartości leżą blisko trzykrotności rzeczywistej badanej odległości, natomiast w 4.9b blisko dwukrotności. W kolejnym rozdziale pochylimy się nad możliwymi przyczynami i rozwiązaniem tego zachowania.

## Rozdział 5

# Multilateriacja

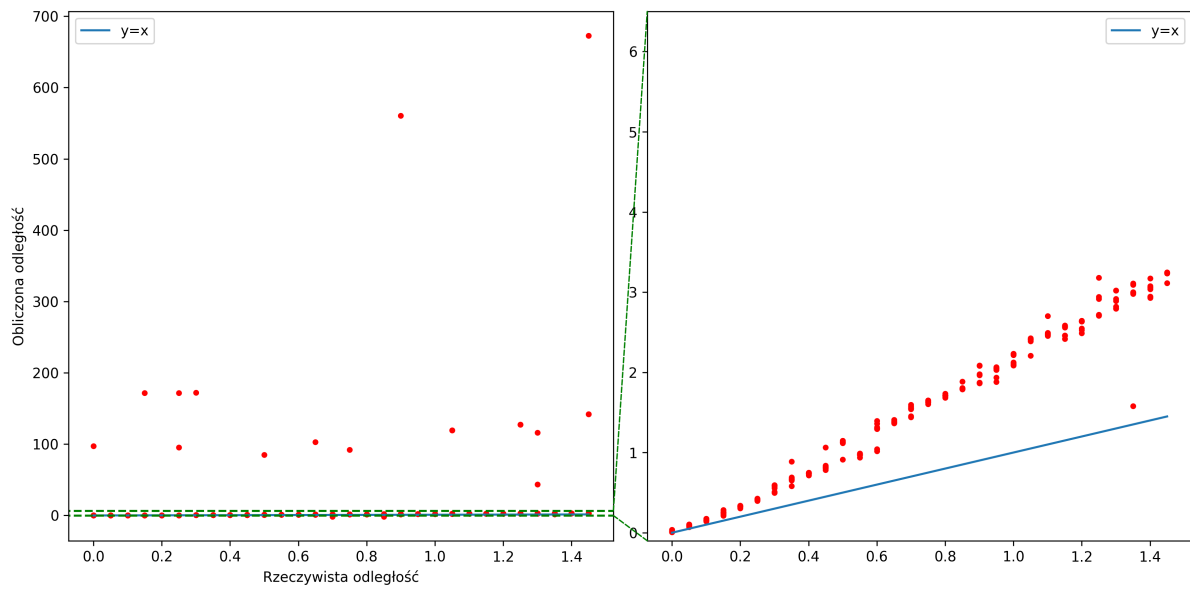
### 5.1. Przygotowanie danych wejściowych

Aby otrzymać poprawne wyniki algorytmu multilateracji, dane wejściowe, w naszym przypadku odległości między węzłami odbiorczymi a nadajnikiem, powinny być jak najbliższe rzeczywistym odległościom z możliwie małymi odchyleniami. Będziemy kontynuować usprawnianie metod uzyskiwania poprawnych wyników zapoczątkowane w rozdziale poprzednim.

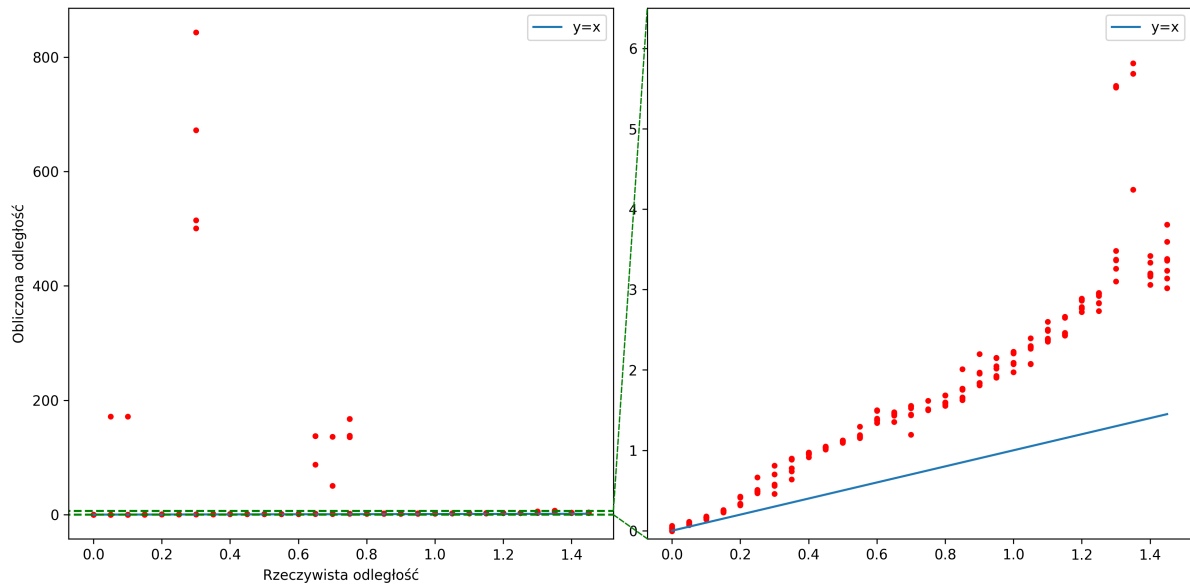
#### Korekcja odległości

W wynikach eksperymentów porównawczych metod synchronizacji czasu węzłów przeprowadzonych w rozdziale 4. zaobserwowaliśmy skalowanie wyników na pierwszy rzut oka zachowujące się liniowo. Przyjrzyjmy się teraz dokładnie temu zjawisku. W tym i kolejnych przypadkach będziemy używać już jedynie synchronizacji sprzętowej z użyciem mikrofonów, ze względu na brak konieczności dodatkowej kalibracji przesunięcia punktu 0.

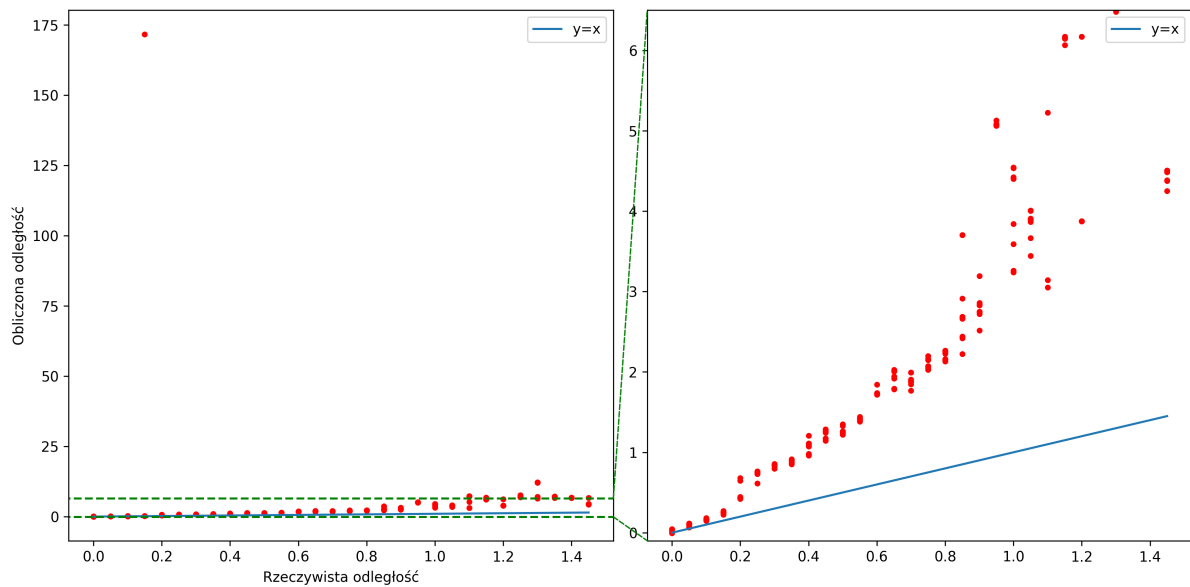
Prawdopodobnym powodem skalowania obliczanych odległości mogą być różnice w rejestracji sygnału mikrofonowego odczytywanego przez mikrokontroler. Poniższe wykresy są wynikiem czterech kolejnych eksperymentów, w których jedyną zmienną była czułość zintegrowanego wzmacniacza mikrofonu. Wzmacniacz ten nie pozwala na precyzyjną regulację, a jedynie na zmianę rezystancji wbudowanego potencjometru. Ponieważ przedział czułości odpowiadający wykrywaniu sygnału węzła nadającego przy jednoczesnym zminimalizowaniu fałszywych aktywacji jest niewielki (około  $\frac{1}{8}$  obrotu potencjometru), cztery zbadane przypadki nie dzielą równo badanego zakresu. Rozpoczynając od największej możliwej czułości, przy każdym kolejnym eksperymencie zmniejszano ją, póki pozwalała wciąż na wykrywanie sygnału z badanych odległości.



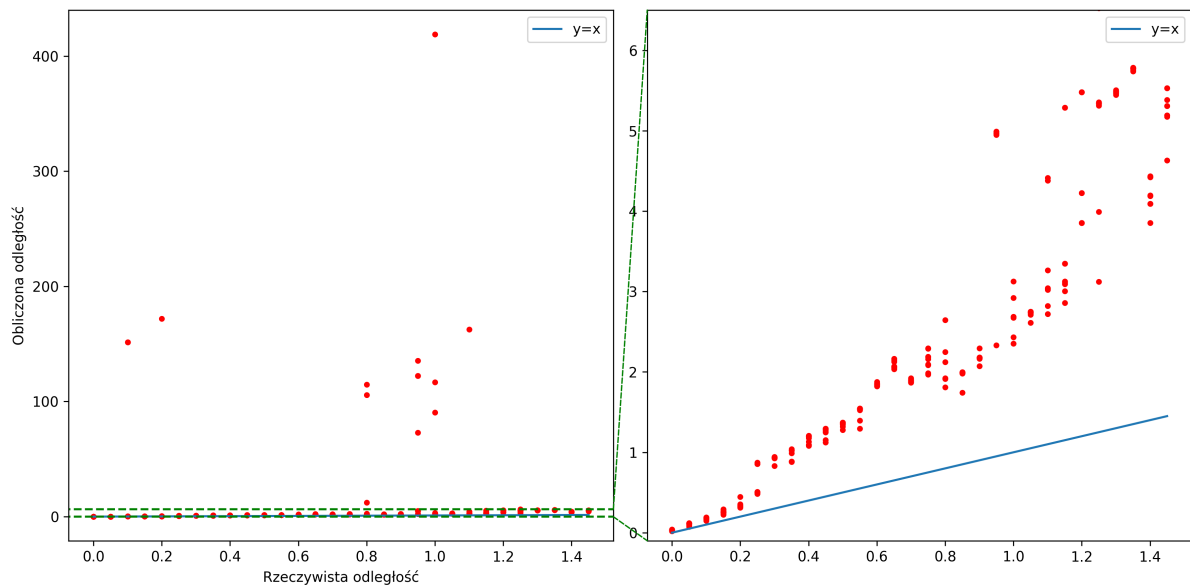
(a) pomiar 1.



(b) pomiar 2.



(c) pomiar 3.

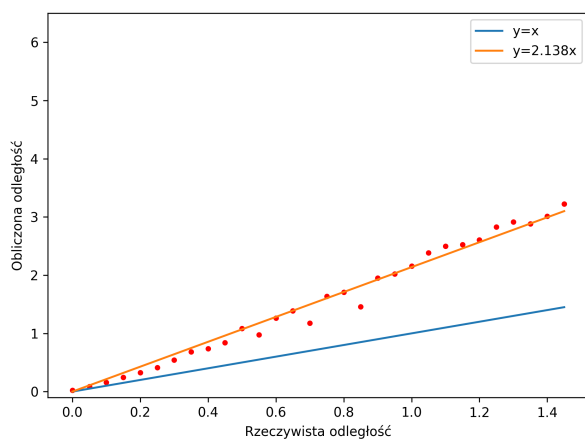


(d) pomiar 4.

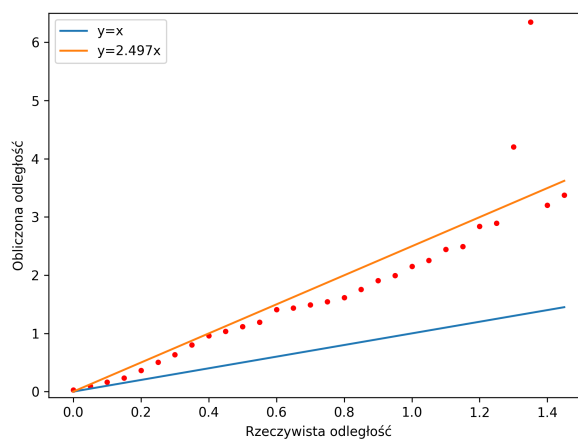
Rys. 5.1: Pomiar obliczanych odległości. Diagram po prawej stanowi powiększenie zaznaczonego obszaru diagramu po lewej.

Aby lepiej odczytać informacje z wykresu uśrednijmy pomiary dla każdej z badanych odległości i dodajmy do nich funkcje liniowe o współczynniku otrzymanym przy pomocy regresji liniowej z tychże uśrednionych punktów.

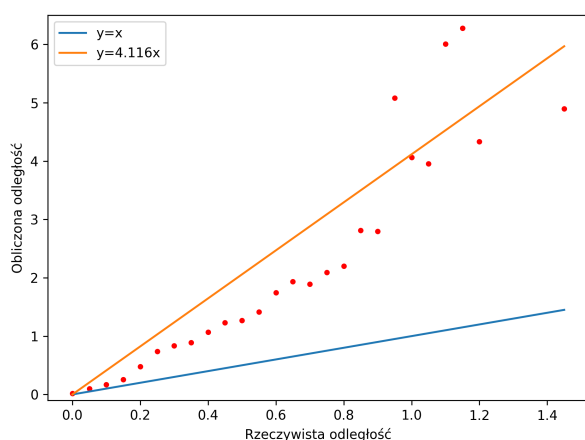




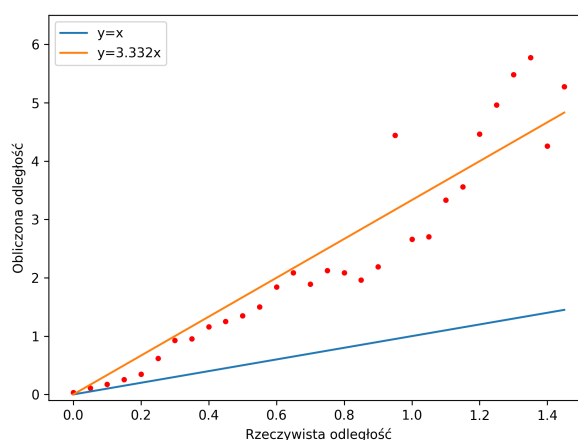
(a) pomiar 1.



(b) pomiar 2.



(c) pomiar 3.



(d) pomiar 4.

Rys. 5.2: Średnie obliczonych odległości

Na wykresach można zauważyć trendy związane ze zmniejszającą się czułością wzmacniacza mikrofonu:

- zmniejszanie się odległości punktu, w którym czułość jest zbyt mała by niezawodnie wyciągać nadawane sygnały,
- współczynnik prostej aproksymującej skalowane odległości rośnie.

Podobne efekty zaobserwowano, kiedy brzęczyk nie był kierowany bezpośrednio w kierunku odbiornika.

## 5.2. Ocena działania systemu

W celu oceny działania tak skonstruowanego systemu multilateracyjnego przeprowadzono szereg eksperymentów testujących jego działanie w zaproponowanych scenariuszach. Bazując na uprzednich obserwacjach, rozszerzono program serwera obliczeniowego tak, aby bezpośrednio przed multilateracją przeprowadzone były synchronizacja czasu oraz korekcja odległości. Zasada działania obu typów węzłów pozostała niezmieniona.

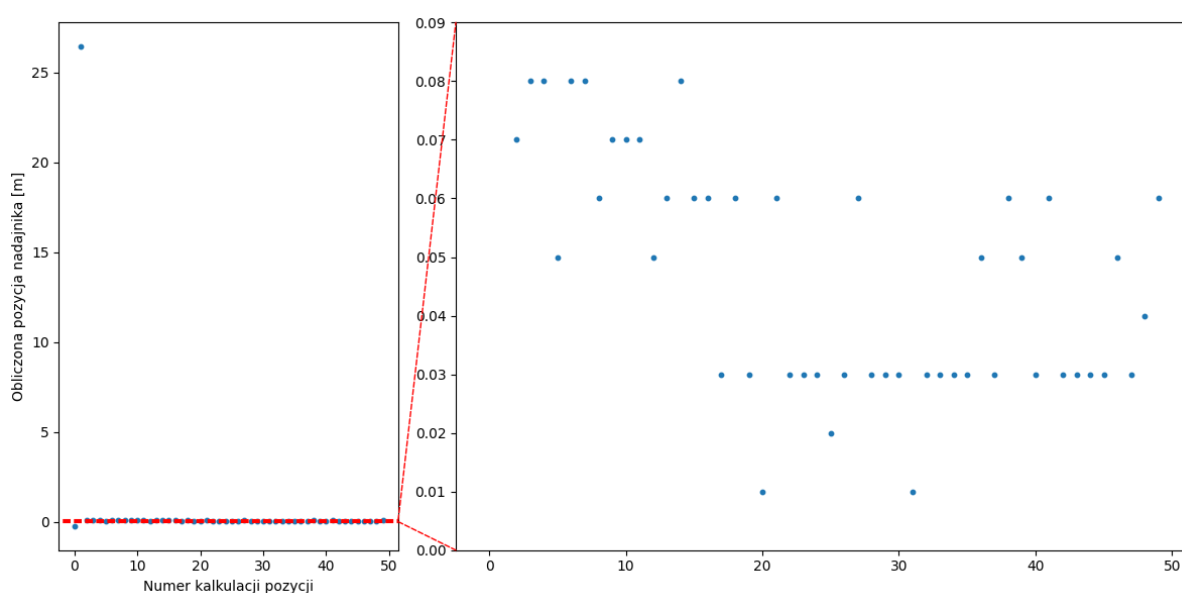
### Wyniki

Wyniki eksperymentów przedstawiono na poniższych wykresach. Mniejsze punkty o wyblakłym kolorze reprezentują badane punkty, w których umieszczony był węzeł nadawczy, natomiast większe, o nasyc-

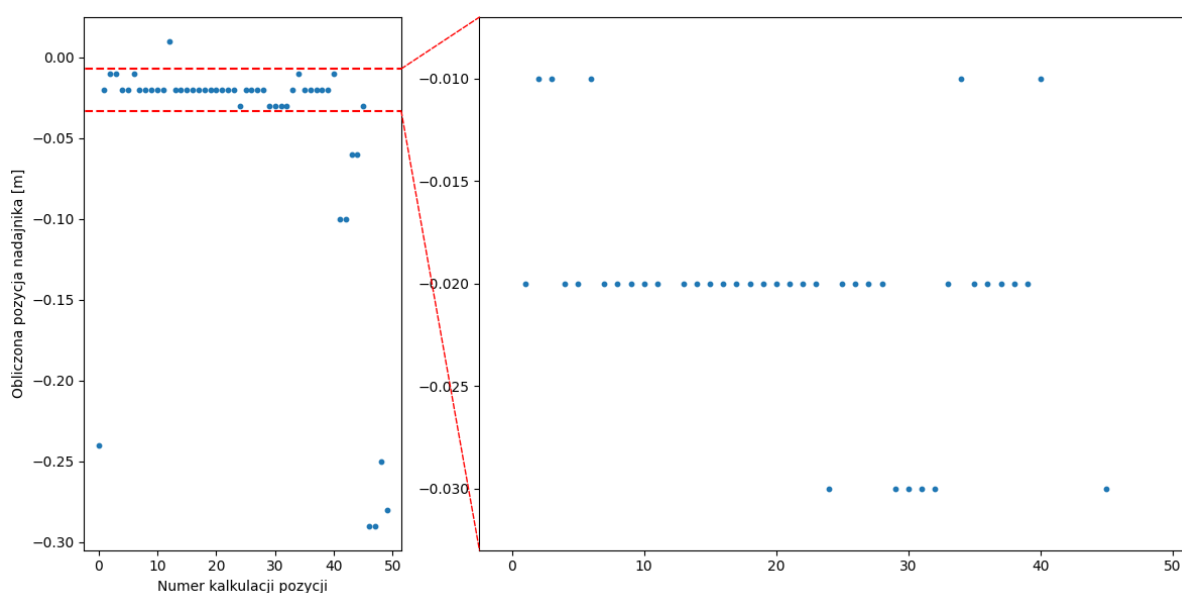
nym kolorze to odpowiadające im punkty będące uśrednionym wynikiem 25 powtórzeń polecenia lokalizacji w systemie. Wykresy należące do tej samej grupy zachowują skalę w celu łatwiejszego porównania. Dodatkowo, dla przypadku jednowymiarowego, dołączono wybrane wykresy zawierające osobne pomiary każdego z punktów, które składały się na wynik uśredniony.

### Eksperymenty jednowymiarowe

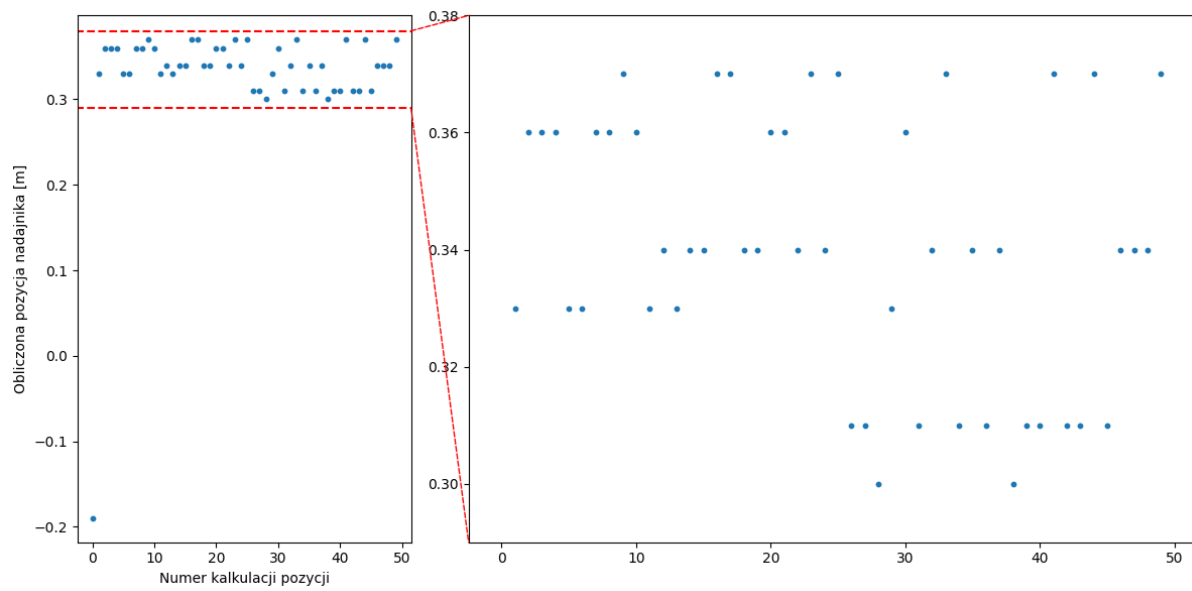
Ewaluację rozpoczęto, tak jak podczas eksperymentu zerowego, od przypadku jednowymiarowego, aby sprawdzić, czy dokładność i stabilność pomiarów w systemie została poprawiona. Tak samo jak pierwotnie nadajnik umiejscowiono w punkcie (0), natomiast dwa odbiorniki w punktach odpowiednio  $(-0, 5)$  i  $(0, 5)$ , wszystkie węzły były stacjonarne. Nadajnik co 0,5 s nadawał sygnał o długości 10 ms, a serwer co 0,5 s zwracał wynik zapytania multilateracji na podstawie ostatnio otrzymanych danych. Następnie zbadano dokładność aproksymacji dla punktów ze zbioru  $\{(-0, 5; -0, 25; 0, 25; 0, 5)\}$ , jak również powtórzono wszystkie pomiary z użyciem dwóch dodatkowych węzłów umiejscowionych w punktach  $(-0, 25)$  i  $(0, 25)$ . Wyniki przedstawiono na wykresach 5.3 oraz 5.4.



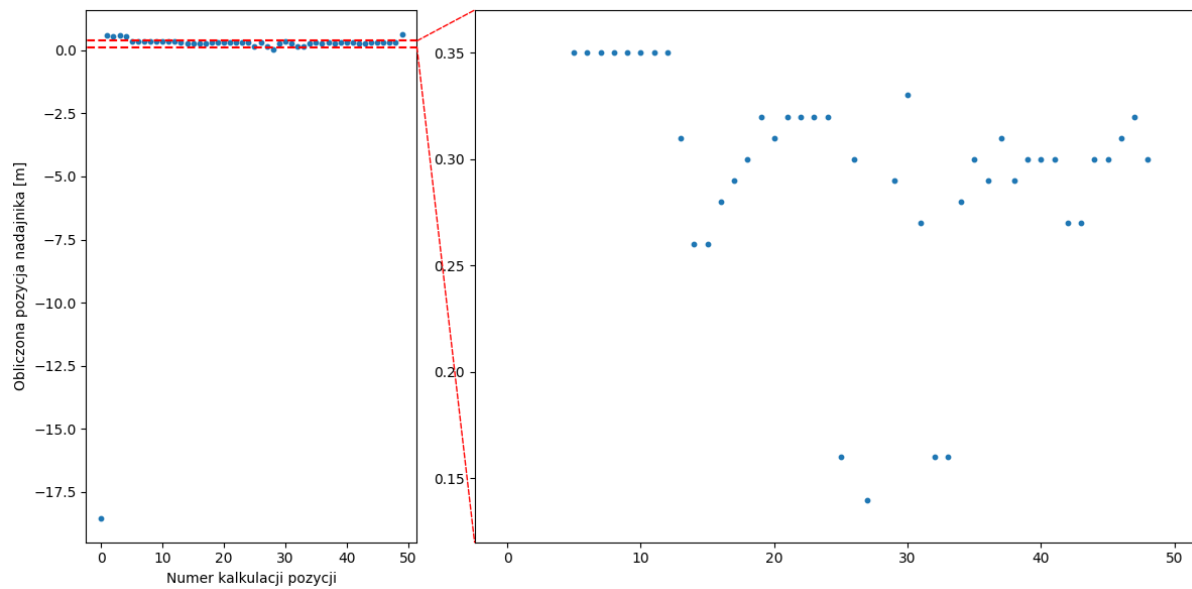
(a) Punkt w pozycji (0), 2 mikrofony



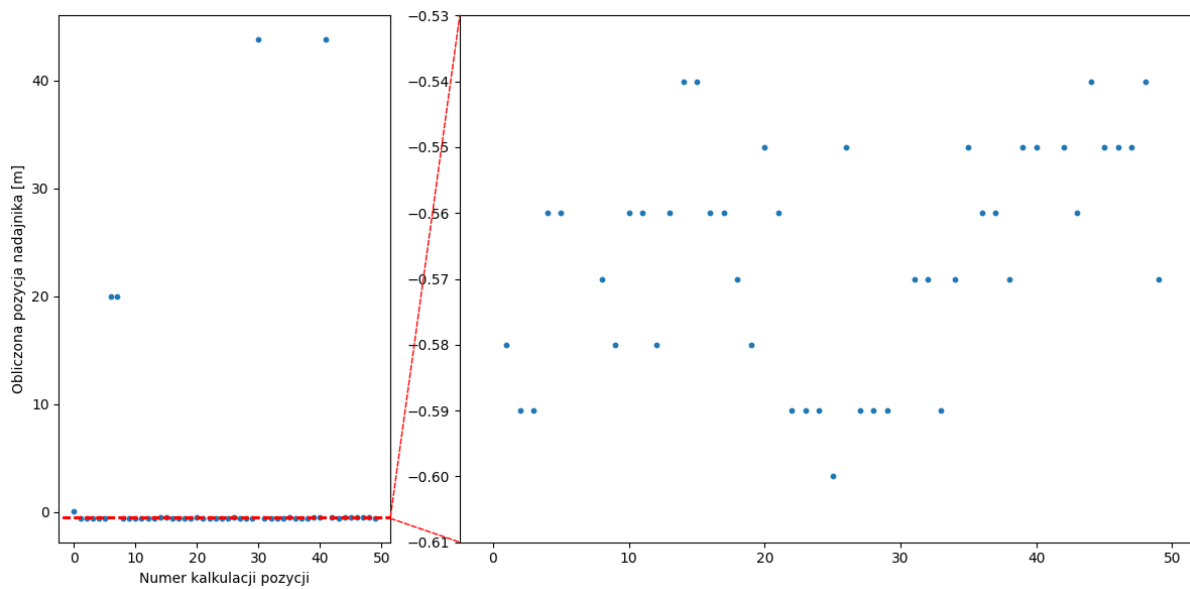
(b) Punkt w pozycji (0), 4 mikrofony



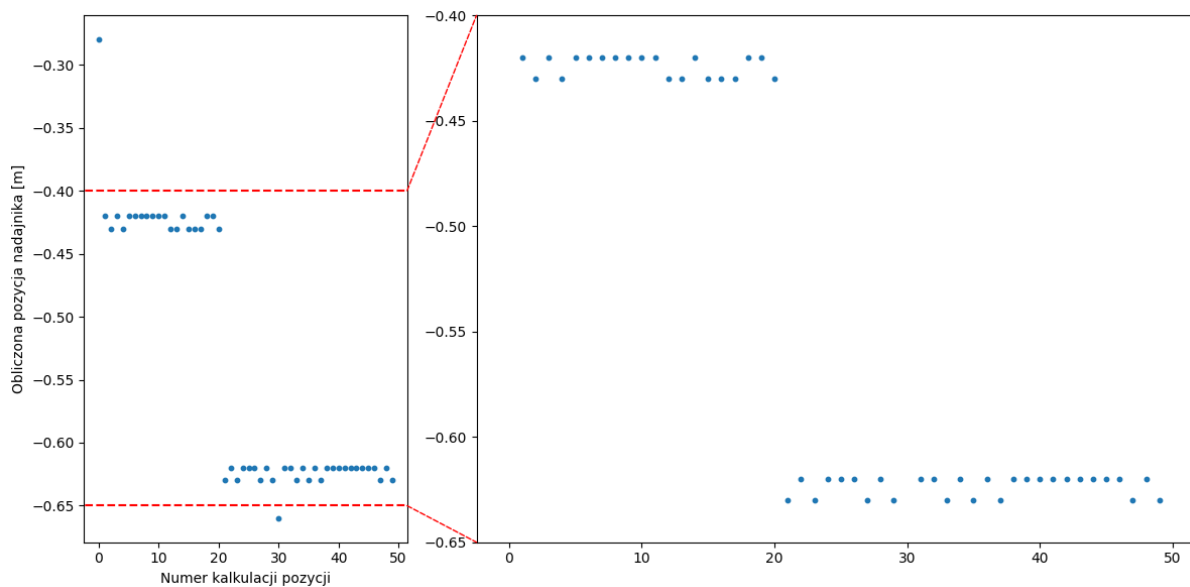
(c) Punkt w pozycji (0,25), 2 mikrofony



(d) Punkt w pozycji (0,25), 4 mikrofony

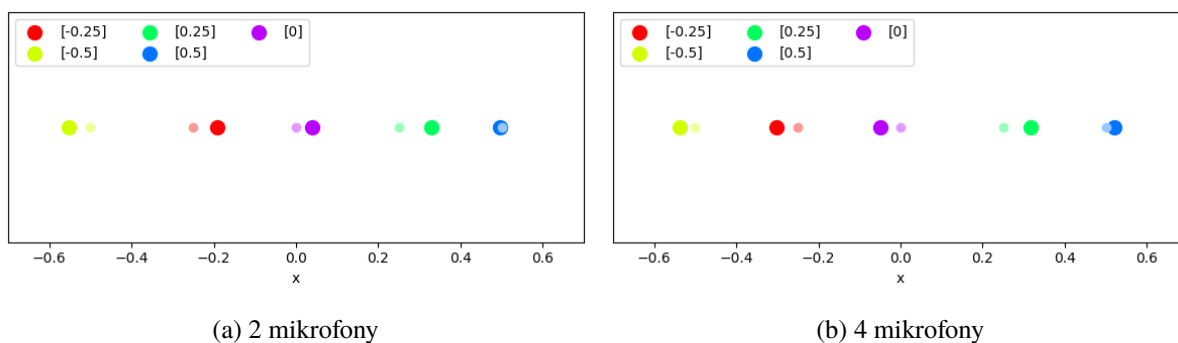


(e) Punkt w pozycji (-0,5), 2 mikrofony



(f) Punkt w pozycji (-0,5), 4 mikrofony

Rys. 5.3: Wykres obliczonej pozycji odbiornika w zależności od czasu



(a) 2 mikrofony

(b) 4 mikrofony

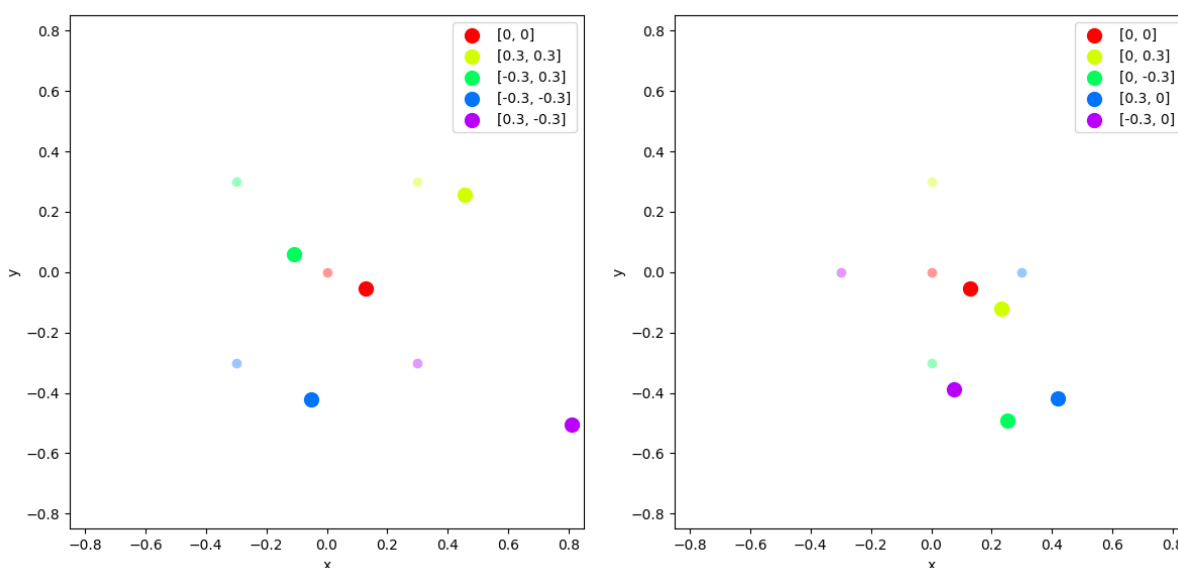
Rys. 5.4: Średnie obliczonych pozycji, wariant 1D. Małe, blade kropki to kolejne punkty testowe, w których umieszczano nadajnik, a duże kropki to średnie wyników zwracanych przez system multilateracyjny. Każdy kolor to inny punkt testowy o koordynatach podanych w legendzie.

### Eksperymenty dwuwymiarowe

Następnie sprawdzono system w wariancie dwuwymiarowym. Aby zachować podobną maksymalną odległość pomiędzy odbiornikami, zdecydowano się na umiejscowienie czterech węzłów na wierzchołkach kwadratu o boku 0,6 symetrycznie względem punktu  $(0; 0)$ , a więc na punktach o współrzędnych  $(\pm 0, 3; \pm 0, 3)$ . Zbadano dokładność lokalizacji w trzech zestawach punktów:

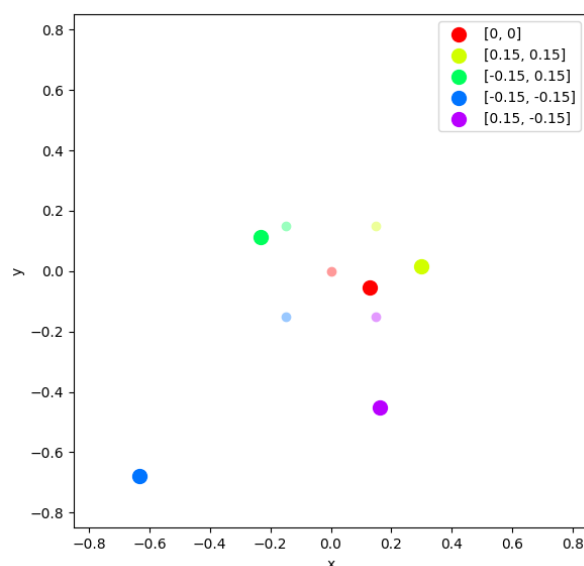
- $\{(0; 0), (0, 3; 0, 3), (-0, 3; 0, 3), (-0, 3; -0, 3), (0, 3; -0, 3)\}$
- $\{(0; 0), (0; 0, 3), (0; -0, 3), (-0, 3; 0), (0, 3; 0)\}$
- $\{(0; 0), (0, 15; 0, 15), (-0, 15; 0, 15), (-0, 15; -0, 15), (0, 15; -0, 15)\}$

Wyniki przedstawiono na wykresach 5.5.



(a) zestaw punktów 1.

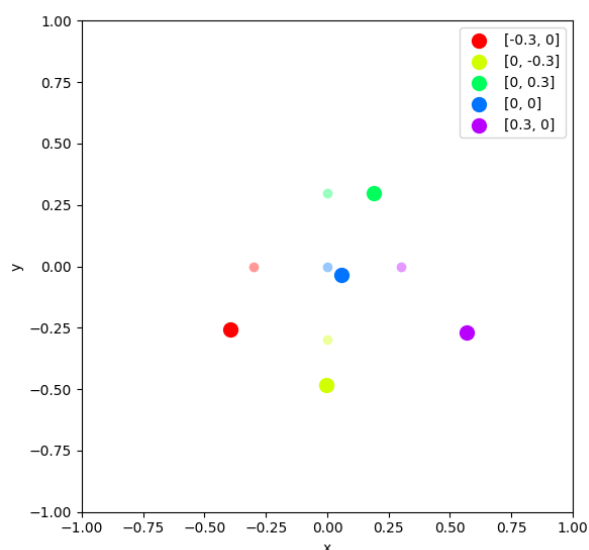
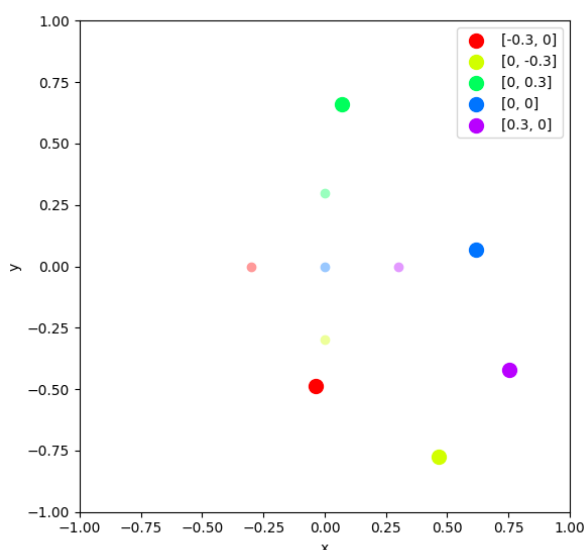
(b) zestaw punktów 2.

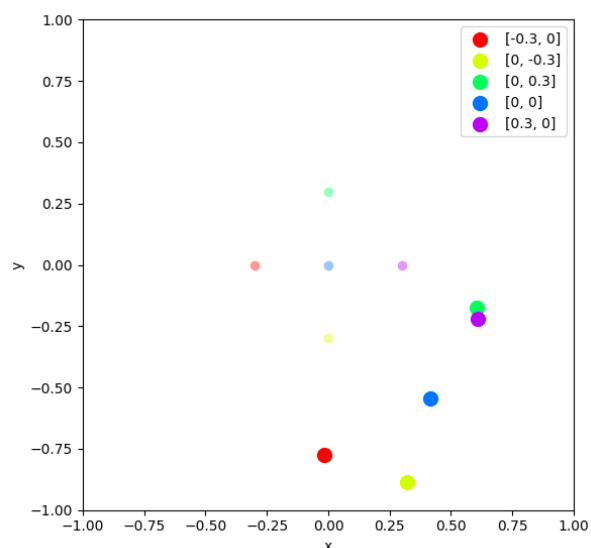


(c) zestaw punktów 3.

Rys. 5.5: Średnie obliczonych pozycji, wariant 2D. Małe, białe kropki to kolejne punkty testowe, w których umieszczano nadajnik, a duże kropki to średnie wyników zwracanych przez system multilateracyjny. Każdy kolor to inny punkt testowy o koordynatach podanych w legendzie.

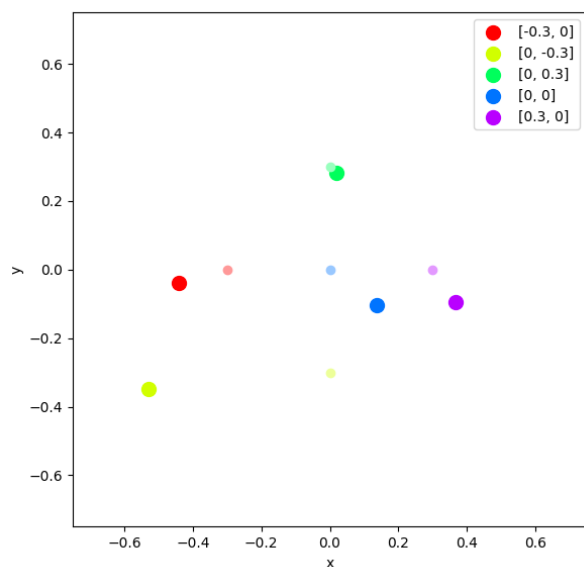
Następnie w celu ewaluacji wpływu charakterystyki otoczenia na działanie systemu przeprowadzono trzy następujące bezpośrednio po sobie eksperymenty zachowując ustaloną na początku synchronizację zegarów i stałe korekcji odległości. Sprawdzone dokładność estymacji tego samego zbioru punktów wyjściowo oraz po rotacji całego układu o  $45^\circ$  i  $90^\circ$  zgodnie z ruchem wskazówek zegara. Wyniki przedstawiono na wykresach 5.6.

(a) rotacja  $0^\circ$ (b) rotacja  $45^\circ$

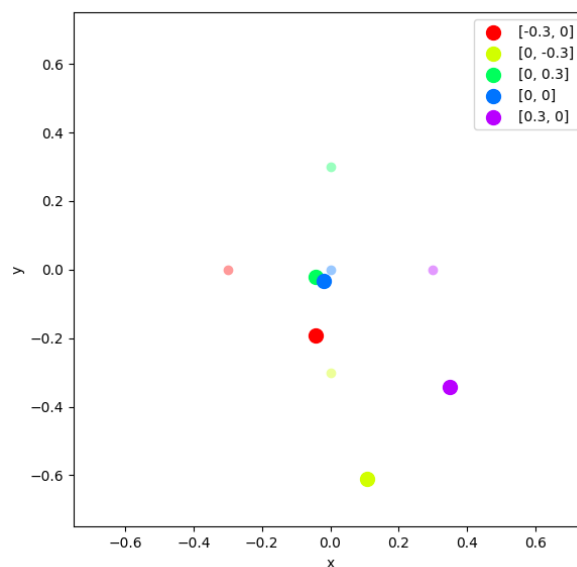
(c) rotacja  $90^\circ$ 

Rys. 5.6: Średnie obliczonych pozycji, wariant 2D z rotacją. Małe, blade kropki to kolejne punkty testowe, w których umieszczano nadajnik, a duże kropki to średnie wyników zwracanych przez system multilateracyjny. Każdy kolor to inny punkt testowy o koordynatach podanych w legendzie.

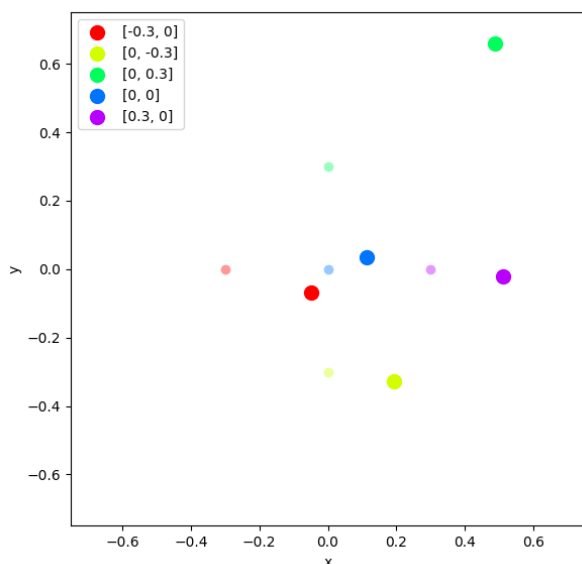
Ostatecznie sprawdzono zachowanie systemu wraz ze zmianą liczby węzłów odbiorczych od minimalnej liczby trzech aż do ośmiu. Węzły umiejscowiono na punktach ze zbioru  $\{(0, 3; 0, 3), (-0, 3; 0, 3), (-0, 3; -0, 3), (0, 3; -0, 3), (0; 0, 425), (-0, 425; 0), (0; -0, 425), (0, 425; 0)\}$  oraz dodawano je zgodnie z tą kolejnością. Wyniki przedstawiono na wykresach 5.7.



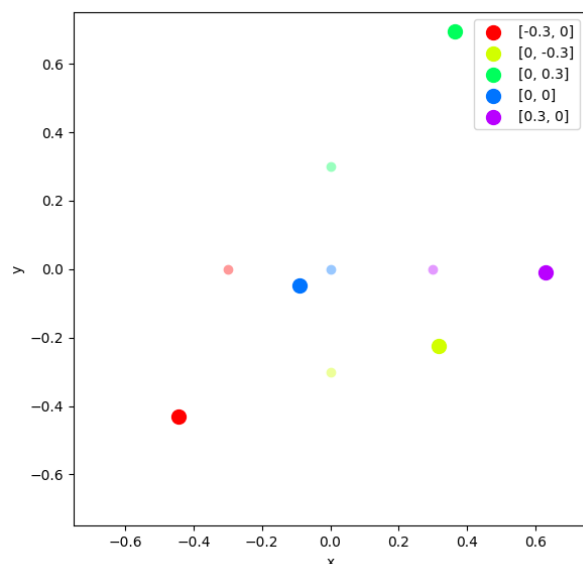
(a) 3 mikrofony



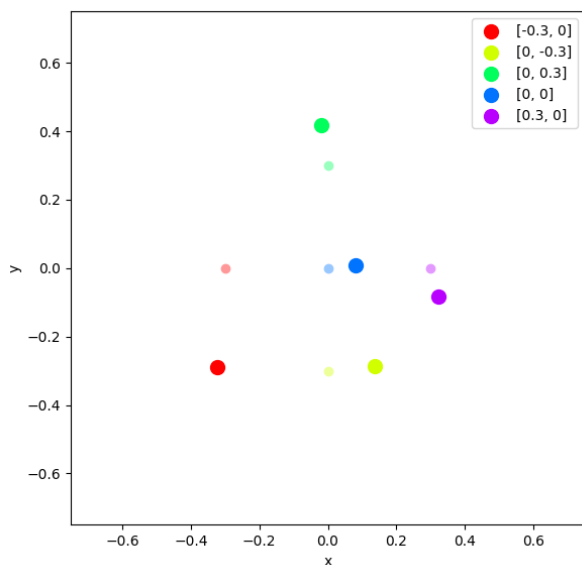
(b) 4 mikrofony



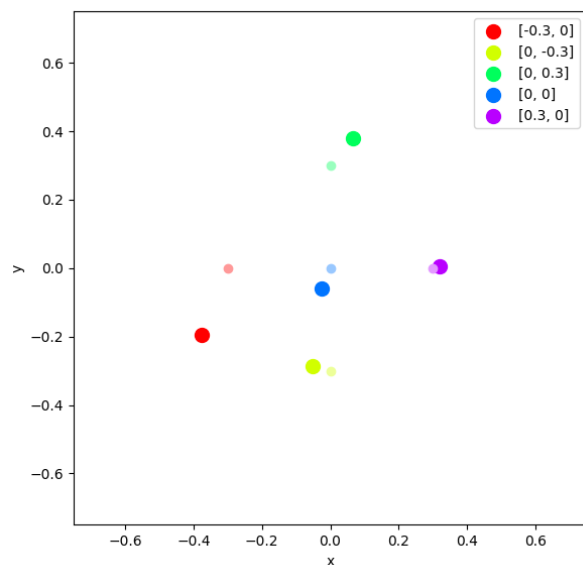
(c) 5 mikrofonów



(d) 6 mikrofonów



(e) 7 mikrofonów



(f) 8 mikrofonów

Rys. 5.7: Średnie obliczonych pozycji, wariant 2D ze zmienną liczbą mikrofonów. Małe, blade kropki to kolejne punkty testowe, w których umieszczano nadajnik, a duże kropki to średnie wyników zwracanych przez system multilateracyjny. Każdy kolor to inny punkt testowy o koordynatach podanych w legendzie.

### 5.3. Interpretacja i wnioski

Rozpoczynając od porównania wyników eksperymentu zerowego z rozdziału 3 z wynikami przypadku jednowymiarowego (sekcja 5.2), można od razu zauważyć poprawę zarówno dokładności jak i stabilności pomiarów. W pierwszej kolejności zwróćmy uwagę na znaczne zmniejszenie przesunięcia obliczanych lokalizacji nadajnika względem jego prawdziwego położenia, które wynika z zastosowania synchronizacji sprzętowej, opisaną w sekcji 4.2, biorącej pod uwagę wszelkie opóźnienia pomijane w przypadku synchronizacji sprzętowej (sekcja 4.1). Ponadto dokładność wyników multilateracji nie ulega gwałtownemu spadkowi wraz z upływem czasu (wyjątkiem w poprawionych eksperymentach jest przypadek przedstawiony na wykresie 5.3f, w którym zauważalna jest nagła zmiana średniej lokalizowanych punktów, czego przyczyną mogło być prawdopodobnie zmiana warunków otoczenia eksperymentu). Ostateczne wyniki



multilateracji jednowymiarowej przedstawione na rysunku 5.4 wskazują na zadowalającą (w zależności od zastosowań) dokładność pomiaru położenia nadajnika.

Przypadek jednowymiarowy jest jednak na tyle zdegenerowany, że do jego rozwiązania nie jest konieczne użycie metod multilateracyjnych, dlatego decydującym o użyteczności systemu było testowanie w wariancie dwuwymiarowym. Wyniki pierwszych testów (5.5) nie były obiecujące - układ punktów względem siebie był w przybliżeniu zachowany, jednak punkty zwracane przez system znajdują się czasami w znacznej odległości od ich rzeczywistych położenia. Kolejne dwie serie testów, przeprowadzone w celu sprawdzenia jakie aspekty wpływają pozytywnie, a jakie negatywnie na dokładność lokalizacji wykazały dwie zależności:

- podatność systemu na zmiany charakterystyki sonicznej otoczenia (5.6);
- wzrost dokładności lokalizacji wraz ze wzrostem liczby węzłów odbiorczych (5.7).

Pierwsza z zależności jest wyraźnie widoczna, ponieważ wraz z obrotem układu węzłów względem otoczenia błąd lokalizacji rośnie, jak również położenie punktów względem siebie ulega zmianie do tego stopnia, że trudno odgadnąć jaki kształt tworzyły w rzeczywistości. Druga zależność nie jest tak wydatna jak poprzednia, ponieważ błąd lokalizacji każdego z punktów nie jest ściśle malejący, czasem większa liczba węzłów nadawczych nie pomaga, a może nawet przeszkadzać (kolejny dodany węzeł mógł mieć niepoprawnie ustawioną czułość wzmacniacza, co znacznie wpłynęłoby na wyniki przy małej liczbie węzłów). Po zwiększeniu liczby do 7 i 8 widać zalety skali w takim systemie - nawet jeśli któraś z raportowanych odległości jest obciążona znaczącym błędem, rozwiązanie aproksymacyjne minimalizujące sumę kwadratów błędów będzie bliższe punktowi, na który wskazują odległości podane przez pozostałe węzły.

# Podsumowanie

Celem pracy było zaprojektowanie i zaimplementowanie systemu multilateracyjnego działającego w dziedzinie fal dźwiękowych. Dokonano wyboru metody multilateracji najlepiej nadającej się do tego zastosowania, jak również przeprowadzono szereg testów wykazujących nieprawidłowości działania, a następnie podjęto próby eliminacji tych błędów. Podczas eksperymentów ukazano znaczenie wpływu zastosowanej metody synchronizacji zegarów urządzeń w systemie, jak również interesujące skalowanie estymowanej odległości wynikającej z zastosowanych w mikrofonach wzmacniaczy operacyjnych.

Wyniki przeprowadzonych badań wskazują, że system multilateracyjny, oparty o szeroko dostępne moduły mikrofonowe, jest zdolny śledzić pozycję nadajnika sygnałów dźwiękowych w podzbiorze przestrzeni dwuwymiarowej, obejmowanym ograniczonym zasięgiem tychże mikrofonów w węzłach odbiorczych. Znaczącą wadą systemu jest duża podatność na warunki akustyczne otoczenia, których mała zmiana może znacząco negatywnie wpłynąć na skuteczność lokalizacji. Dodatkowo, zasięg mikrofonów stanowił ograniczenie, przez co obszar, w którym możliwa była lokalizacja miał mniej niż  $0,5m^2$ . Możliwość działania na większych odległościach między odbiornikami jest niepewna - zastosowanie lepszej jakości mikrofonów, o niższym poziomie szumu, mogłoby znacząco poprawić ich zasięg. Biorąc jednak pod uwagę fakt, że system nie jest kosztowny obliczeniowo, co pozwoliło na zastosowanie prostych, a co za tym idzie, tanich komponentów (koszt odbiornika wyniósł mniej niż 5\$, a serwer mógłby być hostowany na prostym komputerze SBC (*ang. single-board computer*) typu Raspberry Pi) zdolność do lokalizacji nadajnika przez zbudowany system powinna być zadowalająca.

Dalszymi krokami rozwoju systemu, których nie podjęto ze względu na ograniczony budżet czasowy mogłyby być:

- rozszerzenie do przestrzeni trójwymiarowej i ewaluacja działania w jej zakresie,
- bardziej wnikliwe testy i badania roli urządzeń przetwarzających sygnał dźwiękowy na cyfrowy; inwentyzacja wpływu typu rodzaju mikrofonu, wzmacniacza czy przetwornika ADC (*ang. analog to digital*) na estymację odległości,
- stworzenie lepszego algorytmu korekcji estymowanej odległości względem rzeczywistej,
- ewentualne dostosowywanie algorytmów synchronizacji zegarów do przyszłych zastosowań,
- próba minimalizacji wpływu charakterystyki dźwiękowej otoczenia na dokładność wyników.

Podsumowując, zaprojektowany i zaimplementowany system multilateracyjny, działający w dziedzinie fal dźwiękowych, udowodnił swoją skuteczność w śledzeniu pozycji nadajnika w dwuwymiarowej przestrzeni. Otworzył tym samym drogę do dalszych badań i udoskonaleń, które mogą znacząco poszerzyć jego zastosowanie w różnych dziedzinach.

# Literatura

- [1] S. Bancroft. An algebraic solution of the gps equations. *IEEE Transactions on Aerospace and Electronic Systems*, AES-21(1):56–59, 1985.
- [2] R. Bucher, D. Misra. A synthesizable vhdl model of the exact solution for three-dimensional hyperbolic positioning system. *VLSI Des.*, 15, 09 2002.
- [3] Y. Chan, K. Ho. A simple and efficient estimator for hyperbolic location. *IEEE Transactions on Signal Processing*, 42(8):1905–1915, 1994.
- [4] J. Chen, R. Hudson, K. Yao. Maximum-likelihood source localization and unknown sensor location estimation for wideband signals in the near-field. *IEEE Transactions on Signal Processing*, 50(8):1843–1854, 2002.
- [5] W. H. Foy. Position-location solutions by taylor-series estimation. *IEEE Transactions on Aerospace and Electronic Systems*, AES-12(2):187–194, 1976.
- [6] Y. Li, Y. Zhuang, X. Hu, Z. Gao, J. Hu, L. Chen, Z. He, L. Pei, K. Chen, M. Wang, X. Niu, R. Chen, J. Thompson, F. M. Ghannouchi, N. El-Sheimy. Toward location-enabled iot (le-iot): Iot positioning techniques, error sources, and error mitigation. *IEEE Internet of Things Journal*, 8(6):4035–4062, 2021.
- [7] D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.
- [8] D. Mills. Internet time synchronization: the network time protocol. *IEEE Transactions on Communications*, 39(10):1482–1493, 1991.
- [9] A. Norrdine. An algebraic solution to the multilateration problem. *Proceedings of the 15th international conference on indoor positioning and indoor navigation, Sydney, Australia*, wolumen 1315, 2012.
- [10] U. G. P. Office. Wireless communications and public safety act of 1999. <https://www.govinfo.gov/content/pkg/PLAW-106publ81/pdf/PLAW-106publ81.pdf>, 1999. [Dostęp 14-05-2024].
- [11] E. Widdison, D. G. Long. A review of linear multilateration techniques and applications. *IEEE Access*, 12:26251–26266, 2024.
- [12] Z. Yong-xing, Z. Chao, Y. Yu-lei. Research on computer time synchronization by software method. *2011 International Conference on Electronics, Communications and Control (ICECC)*, strony 998–1001, 2011.