



České  
vysoké  
učení technické  
v Praze

Fakulta elektrotechnická  
Katedra radioelektroniky

Bakalářská práce

# Algoritmy určování polohy pomocí komunikace ve viditelném světle

Šimon Ondřej

Vedoucí práce:

doc. Ing. Stanislav Vítek, Ph.D.

Praha, Květen 2023



## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Ondřej** Jméno: **Šimon** Osobní číslo: **492394**  
Fakulta/ústav: **Fakulta elektrotechnická**  
Zadávající katedra/ústav: **Katedra radioelektroniky**  
Studijní program: **Elektronika a komunikace**

## II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

**Algoritmy určování polohy pomocí komunikace ve viditelném světle**

Název bakalářské práce anglicky:

**Positioning Algorithms Using Visible Light Communication**

Pokyny pro vypracování:

- 1) Prostudujte metody určování polohy uvnitř budov pomocí komunikace ve viditelném světle
- 2) Na základě studie proveďte implementaci nejméně dvou metod určení polohy pomocí komunikace ve viditelném světle s využitím infrastruktury dostupné na katedře radioelektroniky
- 3) Vyhodnoťte přesnost algoritmů experimentálními měřeními
- 4) Diskutujte dosažené výsledky

Seznam doporučené literatury:

- [1] BARTOŠ, Filip. Experimentální ověření komunikace ve viditelném světle k lokalizaci osob. Master's Thesis. České vysoké učení technické v Praze. 2015
- [2] BASTIAENS, Sander, et al. Experimental Assessment of the Accuracy of Modulated and Unmodulated Visible Light Positioning. In: 2020 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB). IEEE, 2020. p. 1-5.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

**doc. Ing. Stanislav Vítek, Ph.D. katedra radioelektroniky FEL**

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **15.02.2023**

Termín odevzdání bakalářské práce: **26.05.2023**

Platnost zadání bakalářské práce: **22.09.2024**

\_\_\_\_\_  
doc. Ing. Stanislav Vítek, Ph.D.  
podpis vedoucí(ho) práce

\_\_\_\_\_  
doc. Ing. Stanislav Vítek, Ph.D.  
podpis vedoucí(ho) ústavu/katedry

\_\_\_\_\_  
prof. Mgr. Petr Páta, Ph.D.  
podpis děkana(ky)

## III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

\_\_\_\_\_  
Datum převzetí zadání

\_\_\_\_\_  
Podpis studenta



## Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze, 14. května

.....

Šimon Ondřej

## Poděkování

Velmi děkuji doc. Ing. Stanislavu Vítkovi, Ph.D. za vstřícnost, ochotu a cenné rady při vypracování mé bakalářské práce. Dále děkuji mým blízkým za podporu během celého studia.

## Abstrakt

Cílem této práce je návrh metod pro určování polohy pomocí komunikace ve viditelném světle. Důraz je kladen zejména na jejich využití ve vnitřních prostorech. Nejprve je vysvětleno z jakého důvodu je důležité se určováním polohy v budovách zabývat a jaké výhody technologie viditelného světla nabízí. Následně je představen systém skládající se z vysílačů, přijímače a přesně definovaného měřicího prostředí, v němž je systém testován. Na základě tohoto systému jsou navrženy a implementovány dvě metody určování polohy pomocí komunikace ve viditelném světle. Obě z představených metod určují polohu v jedné konkrétní výšce nad zemí, tedy pouze v 2D prostoru. První z nich využívá strojové učení, konkrétně algoritmus K-Nearest Neighbors. Tato metoda určuje polohu v diskrétních bodech a dosahuje přesnosti 25 cm s více než 99% úspěšností. Druhá z metod je založená na triangulaci a k určení polohy volí analytický přístup. Poloha je určována spojitě a je dosaženo průměrné chyby 6.27 cm. Na závěr jsou shrnuty slabiny implementovaných metod, kterými by se mohly zabývat navazující práce.

**Klíčová slova:** komunikace ve viditelném světle, určování polohy uvnitř budov, lokalizace, K-Nearest Neighbors, triangulace, transimpedanční zesilovač

## Abstract

The aim of this thesis is to propose positioning methods which use visible light communication. The emphasis is mainly on their use in indoor environments. First, it is explained why it is important to deal with positioning in buildings and what advantages does visible light positioning offer. Then, a system consisting of transmitters, a receiver and a well-defined measurement environment in which the system is tested is presented. Based on this system, two visible light positioning methods are proposed and implemented. Both of them determine the position at one specific height above the ground, which means only in 2D space. The first one uses machine learning, specifically the K-Nearest Neighbors algorithm. It determines the receiver's location at discrete points and achieves an accuracy of 25 cm with more than 99% success rate. The second method is based on triangulation and chooses an analytical approach to estimate the position. The position is determined continuously and an average error of 6.27 cm is achieved. Finally, the problems of the implemented methods, which could be addressed in follow-up work, are summarized.

**Keywords:** visible light communication (VLC), indoor positioning, localization, K-Nearest Neighbors, triangulation, transimpedance amplifier

# Obsah

<b>1</b>	<b>Úvod</b>	<b>1</b>
<b>2</b>	<b>Určování polohy v budovách</b>	<b>3</b>
2.1	Využití lokalizačních systémů v budovách . . . . .	3
2.2	Technologie pro určování polohy v budovách . . . . .	3
2.3	Výhody a nevýhody VLP . . . . .	5
2.4	Metody pro určení polohy . . . . .	5
2.4.1	Triangulace . . . . .	5
2.4.2	Fingerprinting . . . . .	8
2.4.3	Proximity . . . . .	8
2.4.4	Vision analysis . . . . .	9
2.4.5	Hybrid algorithms . . . . .	10
<b>3</b>	<b>Systém pro testování metod VLP</b>	<b>11</b>
3.1	Vysílače . . . . .	11
3.2	Robot s přijímačem . . . . .	12
3.2.1	Zpracování světelného signálu . . . . .	12
3.2.2	Software mikrokontroléru . . . . .	14
3.3	Aplikace vyhodnocující polohu . . . . .	15
3.4	Testovací prostředí . . . . .	17
<b>4</b>	<b>Lokalizace algoritmem KNN</b>	<b>19</b>
4.1	Strojové učení . . . . .	19
4.2	Algoritmus KNN . . . . .	20
4.3	Implementace a testování . . . . .	21
4.3.1	Implementace . . . . .	21
4.3.2	Dosažené výsledky . . . . .	22
<b>5</b>	<b>Lokalizace metodou DFT triangulace</b>	<b>23</b>
5.1	Princip metody . . . . .	23
5.2	Implementace metody . . . . .	24
5.3	Dosažené výsledky . . . . .	26
<b>6</b>	<b>Závěr</b>	<b>27</b>
	<b>Použitá literatura</b>	<b>29</b>





## Seznam obrázků

2.1	Koncept VLP systému z [10]. . . . .	4
2.2	Obvyklá HW struktura VLP. Převezato z práce [10]. . . . .	4
2.3	Metody TOA a RTT . . . . .	6
2.4	TDOA. Upravený obrázek z [15]. . . . .	7
2.5	Základní koncept AoA. Převezato z práce [16]. . . . .	8
2.6	Dírkový model kamery. . . . .	9
3.1	Světla využitá k lokalizaci. . . . .	11
3.2	TIA. . . . .	12
3.3	Transimpedanční zesilovač připevněný na podvozku robota. . . . .	13
3.4	Software běžící na ESP32. . . . .	15
3.5	Hlavní okno aplikace vyhodnocující polohu. . . . .	16
3.6	Zavěšení světla. . . . .	17
3.7	Uspořádání světla. . . . .	17
4.1	Myšlenka KNN. Převezato z [36]. . . . .	20
4.2	Body v nichž byl měřen signál pro trénování metody. Zároveň se jedná o okno z aplikace zobrazující polohu. . . . .	21
5.1	Funkce $f(x, y)$ a $g(x, y)$ . . . . .	24
5.2	Závislost amplitudy DFT vzdalujícího se světla na vzdálenosti. . . . .	25
5.3	Odhady poloh. . . . .	26

# 1 Úvod

Určování polohy ve vnějších prostorech bylo dlouhodobě velmi důležité pro mnoho různých oblastí lidské činnosti. Velký pokrok nastal se vznikem družicových systémů, jakými jsou například GPS nebo GLONASS. Na základě toho vzniká myšlenka, že by podobný systém mohl nalézt uplatnění i ve vnitřních prostorech o velké rozloze. Družicové systémy však v budovách neposkytují dostatečnou přesnost, primárně kvůli velkému útlumu signálu a jeho vícecestnému šíření. Z toho důvodu je nutné přijít s alternativní technologií, která družicové systémy v těchto místech nahradí. Tímto problémem se v posledních dvaceti letech zabývalo velké množství prací, ale i přesto nevznikla shoda na tom, která technologie je nejlepší. Jednou ze slibných technologií je komunikace ve viditelném světle. Jejich výhod je mnoho, ale asi nejvýznamnější je vysoká přesnost. Velmi často je dosaženo jednotek centimetrů, v některých případech dokonce i milimetrů. Většina jiných technologií je z hlediska přesnosti až o několik řádů horší. Komunikace ve viditelném světle má i své nevýhody. Tou je například, že se signál šíří téměř výhradně na přímou viditelnost. Je tedy velmi jednoduché nechtěně přerušit přenos signálu mezi vysílačem a přijímačem.

Cílem práce je implementovat dvě metody, které budou na komunikaci ve viditelném světle založené a dokážou ve stanoveném testovacím prostředí polohu jednoznačně určovat. Tyto metody ke svému fungování potřebují určitou hardwarovou a softwarovou infrastrukturu. K tomu jsou využita tři světla, která umožňují vysílat světelné signály různých průběhů. Také je vytvořen obvod s fotodiodou pro příjem těchto signálů a software pro jejich zpracování. První z navržených metod stojí na algoritmu K-Nearest Neighbors, který spadá pod strojové učení. Druhá je zcela odlišná a určuje polohu pomocí triangulace. Obě z nich jsou implementovány jako součást vytvořené infrastruktury a nakonec je provedeno jejich otestování a zhodnocení.



## 2 Určování polohy v budovách

Určování polohy slouží k lokalizaci osob nebo objektů. Ve vnějších prostorech jsou k tomuto úkolu využívány převážně družicové polohové systémy (GNSS). Ty však nefungují dobře uvnitř budov, kvůli čemuž je třeba pro vnitřní prostory vymyslet alternativní systém. Tato kapitola se zabývá otázkou proč je určování polohy ve vnitřních prostorech užitečné, jaké technologie by mohly být využity a nakonec popisuje existující algoritmy, které polohu vyhodnocují.

### 2.1 Využití lokalizačních systémů v budovách

V první řadě lze určování polohy použít k nalezení osob a předmětů. V případě předmětů lze jejich sledováním například zabránit krádeži nebo umožnit rychlé nalezení, pokud se předmět ztratí. U určitých povolání by zase znalost polohy osob mohla pomoci ke zvýšení bezpečnosti na pracovišti.

Dále je třeba popsat asi nejvýznamnější aplikaci, kterou je využití v navigačním systému. Navigace je proces nalezení ideální cesty mezi počátečním a koncovým bodem a následné monitorování polohy, s jejíž pomocí se upravuje cesta k cíli. Kvůli velkému úspěchu navigačních systémů ve vnějších prostorech se v posledních letech stala právě navigace ve vnitřních prostorech velmi významnou oblastí výzkumu. Způsoby využití jsou často ve velkých budovách jako jsou vlaková nádraží, nemocnice, areály vysokých škol, letiště a mnoho dalších [1]. V těchto místech bývá obtížné se zorientovat a rychle se dostat na potřebné místo, což by mohla vnitřní navigace značně usnadnit. Obzvláště prospěšná by mohla být pro osoby s poruchou zraku, čímž se zabývá například [2]. Dále lze navigační systém využít k dohledání předmětů a přístrojů, a tím usnadnit práci zaměstnancům velkých skladišť nebo rozšířit využití robotů. Práce [3] se pak zabývá vnitřní navigací pro využití u hasičů a to i s ohledem na možnost nasazení systému při zásahu aniž by byl předem v budově instalován.

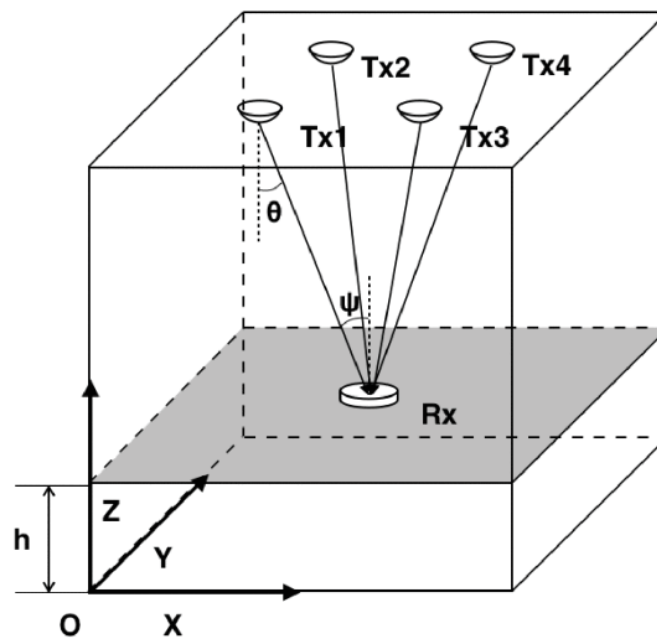
### 2.2 Technologie pro určování polohy v budovách

Jak již bylo zmíněno, ve vnějších prostorech slouží k určení polohy systémy GNSS. Nejznámějším a nejrozšířenějším z nich je Global Positioning System (GPS) [4]. Využití těchto systémů v budovách by mohlo být výhodné, protože infrastruktura již existuje. Bez modifikace však GPS v budovách použít nelze kvůli problémům jako jsou slabá síla signálu, interference vln nebo jevu s názvem multipath. Multipath nastává v situaci, kdy se signál od vysílače k přijímači dostane z více směrů kvůli odrazům, čímž se sníží přesnost určení polohy [5]. Dále je také ve vnitřních prostorech kladen větší důraz na přesnost systémů, protože prostory nejsou tak rozlehlé a nepřesné určení polohy i o pár metrů může způsobit, že informace o poloze je nepoužitelná. I přes zmíněné problémy existují návrhy, například [6], jak zajistit použitelnost GPS v budovách.

Alternativou k GPS by mohlo být mnoho různých technologií. Nejčastější jsou návrhy využívající přenosu informací v rádiovém spektru, jako například Bluetooth, Wi-Fi, UWB nebo RFID. Například [7] kombinuje technologie Wi-Fi a Bluetooth Low Energy s dosaženou přesností lehce pod jeden metr. Práce [8] zase využívá UWB a do-

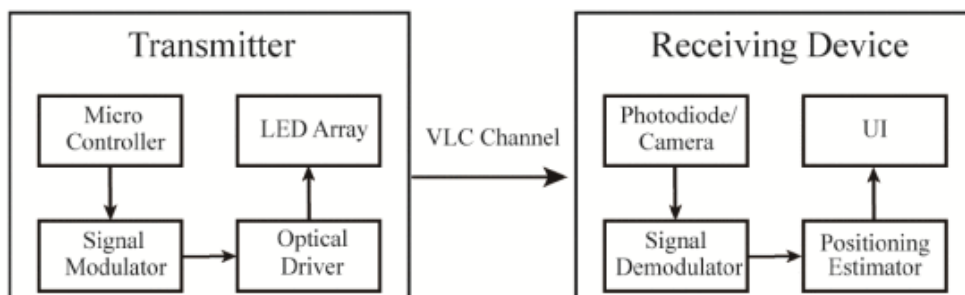
sahuje přesnosti 1-2 decimetry. Existují také technologie využívající magnetické pole Země [9], komunikaci ve viditelném světle (VLC), infračervené záření, ultrazvuk a další.

Systémy využívající VLC pro určování polohy se obecně nazývají Visible Light Positioning (VLP). Tyto systémy se v poslední době staly významným kandidátem díky mnoha výhodám popsaným v následující podkapitole. Architektura VLP je podobná jako u systémů využívajících Wi-Fi nebo Bluetooth. Zjednodušený koncept VLP systému zobrazuje obrázek č. 2.1. Vysílače bývají tvořené LED světly a jsou umístěné na stropu. Pohyblivý přijímač je tvořen optickým senzorem a nachází se mezi zemí a světly. Vysílače posílají data k přijímači a ten podle nich určuje svojí polohu v prostoru [10].



Obrázek 2.1: Koncept VLP systému z [10].

Druhý obrázek zobrazuje strukturu VLP systému z hlediska hardwaru. Alternativou k poli LED uvedenému na obrázku bývá často pouze jedna samostatná LED u každého z vysílačů.



Obrázek 2.2: Obvyklá HW struktura VLP. Převzato z práce [10].

## 2.3 Výhody a nevýhody VLP

Určování polohy uvnitř budov pomocí VLP má mnoho výhod. První z nich je možnost využít již existující infrastrukturu osvětlení. Díky existující infrastruktuře se významně sníží náklady na implementaci těchto systémů. Další výhody jsou dané tím, že hlavním zdrojem osvětlení bývají LED. Ty mají životnost dosahující 25–50 tisíc hodin, měrný světelný výkon přesahující 100 lm/W, ale přesto velmi nízkou cenu [11]. Světlo má narozdíl od vln na rádiových frekvencích velmi krátkou vlnovou délku. Kvůli tomu neproniká skrz zdi, čímž je zajištěna vysoká bezpečnost. Dále je nutné zmínit, že tyto systémy dosahují velmi vysoké přesnosti. Často je dosaženo přesnosti řádově centimetrů. V práci [12] se podařilo navrhnout a počítačově simulovat systém s chybou polohy menší než 5 milimetrů. Světelné záření je také imunní vůči interferencím s rádiovým spektrem, což může být důležité v nemocnicích nebo u aplikací v oblasti letectví [11].

Samozřejmě existují i nevýhody VLP systémů. Hlavní z nich je, že překážky zablokuje signál, který se nedokáže dobře šířit okolo ani skrz. Velká hustota a vhodné rozmístění přijímačů může v neměnném prostředí tento problém vyřešit. Nicméně i při dobrém rozmístění vysílačů se nelze vyhnout efektu, který je v anglické literatuře nazýván „shadowing“. Jedná se o situaci, kdy síla přijatého signálu na daném místě kolísá vlivem měnících se překážek, kterými mohou být například lidé [13]. Dalším problémem mohou být odrazy světla od blízkých předmětů.

Nakonec je nutné zmínit, že při návrhu systémů VLP a nebo obecněji VLC je nutné při modulaci signálu dát pozor, aby nevznikalo blikání viditelné lidským okem [11]. Nejedná se přímo o nevýhodu, ale spíše důležitý požadavek, který je třeba brát v potaz.

## 2.4 Metody pro určení polohy

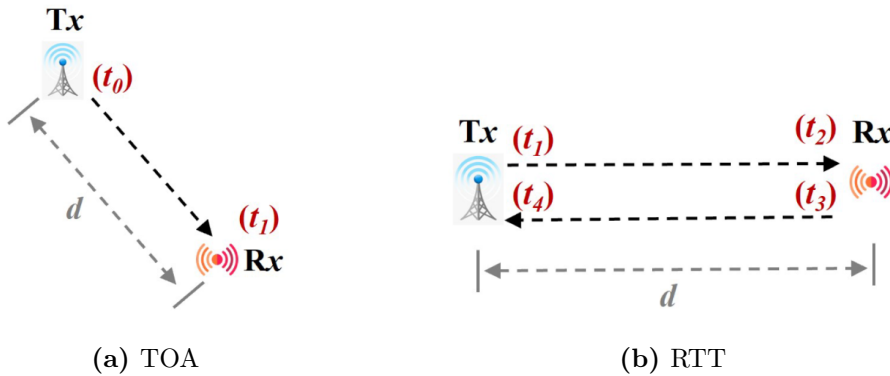
Rozdělení metod se často liší. Téměř vždy bývají uvedeny triangulace, fingerprinting, a proximity. V této práci je uvedené rozdělení vycházející z [14], která navíc uvádí metody vision analysis a hybrid algorithms.

### 2.4.1 Triangulace

Triangulace využívá k určení polohy geometrii trojúhelníků. V anglické literatuře bývá rozdělena na angulaci a lateraci. Laterace určuje polohu na základě naměřených vzdáleností od referenčních bodů. Tyto vzdálenosti bývají určeny pomocí Received Signal Strength (RSS), Time of arrival (TOA) nebo Time difference of arrival (TDOA). Naпротив tomu angulace určuje polohu na základě úhlů. Technikou pro určení úhlů bývá nejčastěji Angle of Arrival (AOA) [14].

**RSS** udává výkon přijatý přijímačem. Tento parametr může být využit k odhadnutí vzdálenosti přijímače od vysílače. Myšlenka vychází z toho, že přijatý signál klesá s rostoucí vzdáleností od vysílače. Na základě znalosti vztahu mezi útlumem a vzdáleností je pak možné odhadnout, jak daleko od sebe přijímač s vysílačem jsou [15]. V praxi může u této metody vzniknout problém, pokud RSS závisí kromě vzdálenosti také na směru. U VLP systémů bývají důvodem této závislosti odrazy od blízkých předmětů nebo překážky blokující světelné záření [14].

**TOA** je založená na předpokladu, že vzdálenost vysílače a přijímače je přímo úměrná době, za kterou se vlna přenese od vysílače k přijímači. Pokud víme, kdy byl signál od vysílače vyslán a také kdy byl přijímačem přijat, lze ze znalosti rychlosti šíření elektromagnetické vlny ve vzduchu ( $c \approx 3 \cdot 10^8 \text{ ms}^{-1}$ ) vypočítat jejich vzdálenost ze vztahu  $d = c(t_1 - t_0)$  [m], kde  $t_1$  udává čas přijetí signálu a  $t_0$  čas jeho vyslání. Pro jednoznačné určení polohy v 2D prostoru je nutné mít k dispozici alespoň 3 vysílače [16]. Nevýhodou této metody je nutnost synchronizace vysílačů a přijímače a obvykle také požadavek na přenos informace o přesném čase vyslání daného signálu [15].



**Obrázek 2.3:** Metody TOA a RTT

Práce [17], z níž pochází obrázek 2.3 zmiňuje také metodu Round Trip Time (RTT). Ta je zobrazena na pravém z obrázků. RTT narozdíl od TOA vyžaduje obousměrnou komunikaci, ale jinak se jedná o prakticky stejnou metodou.

**TDOA** opět využívá dobu šíření signálu k určení polohy. Na rozdíl od TOA však není známá přesná doba, po kterou se signál přenášel. Místo toho se používá rozdíl dvou časů, v nichž signál dorazil. Systémy založené na TDOA obvykle používají jeden ze dvou způsobů pro určení tohoto rozdílu. První z nich pochází z [18] a využívá více stacionárních přijímacích stanic. Uvažujeme-li čas vyslání signálu  $t_e$ , rychlost šíření signálu  $c$ , a vzdálenosti přijímačů od vysílače  $r_1$  a  $r_2$ , pak lze časy  $t_1$  a  $t_2$ , kdy signál k přijímačům dorazil popsat rovnicemi

$$t_1 = t_e + \frac{r_1}{c} \text{ [s] ,} \quad (2.1)$$

$$t_2 = t_e + \frac{r_2}{c} \text{ [s] .} \quad (2.2)$$

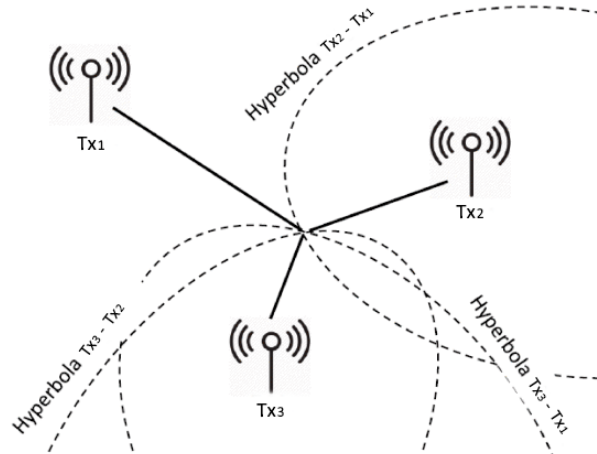
Tyto rovnice jsou pouze upravenou verzí výše uvedené rovnice pro TOA. Vzhledem k tomu, že přesný čas vyslání signálu neznáme, je třeba rovnice odečíst jednu od druhé a zbavit se proměnné  $t_e$ . Pokud zároveň přepíšeme vzdálenosti  $r_1$  a  $r_2$  a vynásobíme takto vzniklý vztah rychlostí světla  $c$ , pak dostáváme

$$\sqrt{(x_t - x_1)^2 + (y_t - y_1)^2 + (z_t - z_1)^2} - \sqrt{(x_t - x_2)^2 + (y_t - y_2)^2 + (z_t - z_2)^2} = c(t_1 - t_2) \text{ [m] ,} \quad (2.3)$$



kde souřadnice  $[x_t, y_t, z_t]$  popisují neznámou polohu vysílače,  $[x_1, y_1, z_1]$ ,  $[x_2, y_2, z_2]$  jsou souřadnice přijímačů a  $t_1 - t_2$  je naměřený rozdíl časů v nichž signál dorazil. Výsledný vztah reprezentuje hyperboloid. Pokud bychom pracovali se souřadnicovým systémem ve 2D, pak by výsledkem byla hyperbola. TDOA mezi dvěma přijímači tedy lokalizuje polohu vysílače na hyperbolu nebo hyperboloid. Pomocí pouhých 2 přijímačů není možné určit přesnou polohu vysílače v prostoru. K 2D lokalizaci jsou potřebné minimálně 3 přijímače a pro 3D minimálně 4, přičemž parametr TDOA se počítá vždy mezi dvěma z nich.

U druhého způsobu uvedeného v [15] je situace obrácená. Signál ze zde šíří z několika vysílačů k jednomu přijímači. Pomocí rozdílných časů příchozích signálů a rychlosti světla je jako u minulého způsobu napočítán dvojnásobek délky hlavní poloosy hyperboly. Nyní se však rozdíl časů v nichž signál dorazil, určuje na jediném přijímači místo více vysílačích. Namísto vysílače se tedy na hyperboloidu nachází přijímač a cílem je nalézt průsečík hyperboloidů, který přesně určí polohu přijímače.



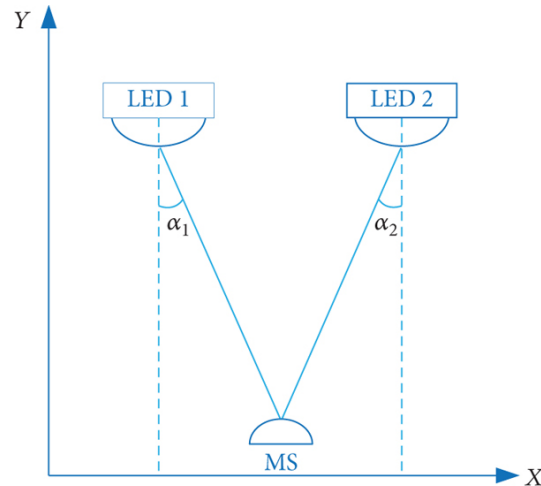
**Obrázek 2.4:** TDOA. Upravený obrázek z [15].

Výhodou oproti TOA je, že vysílací a přijímací stanice už nemusí být perfektně synchronizovány, požadavek na synchronizaci zůstává u prvního způsobu pouze u přijímačů a u druhého pouze u vysílačů.

**AOA** vyjadřuje úhel, pod kterým signál z vysílací stanice dorazil k přijímači. Konkrétně se jedná o úhel mezi normálou plochy na níž jsou umístěny vysílače a přímkou spojující vysílač a přijímač. Z pohledu geometrie lze ve 2D prostoru určit polohu přijímače pomocí průniku dvou přímek, které mají směr určený právě zmíněným úhlem. V praxi však nelze úhel změřit přímo a namísto toho se využívá Lambertův zákon, kterým se LED diody řídí [16]. Podle něho zářivost izotropního rovinného plošného zdroje v každém jeho bodě klesá s kosinem odklonu od kolmice k ploše zdroje záření [19]. Na základě tohoto zákona lze využít fotodiody a určit polohu přijímače. Například práce [20] se návrhem takového systému zabývá a dosahuje průměrné přesnosti 25 centimetrů.

Výhodou této techniky oproti jiným je, že polohu přijímače je možné ve 2D prostoru určit pomocí pouze dvou vysílačů a znalosti jejich polohy. Ve 3D prostoru je situace obdobná, pouze jsou potřebné vysílače 3. Dále je výhodné, že tato technika nevyžaduje

časovou synchronizaci mezi vysílači, a ani mezi vysílači a přijímačem. U systémů využívajících k určení polohy rádiové spektrum bývá obvykle složité tuto techniku využít. Naopak VLP systémy těží z toho, že se signály šíří téměř jen na přímou viditelnost, což zjednodušuje určení úhlu [14].



**Obrázek 2.5:** Základní koncept AoA. Převzato z práce [16].

### 2.4.2 Fingerprinting

Určení polohy probíhá u této metody ve dvou fázích. V první z nich jsou sbírána data, která se liší v závislosti na poloze a dohromady jednoznačně mapují prostor. Ve druhé fázi se pak porovnávají původně nasbíraná data s těmi měřenými v reálném čase a výsledná poloha se odhaduje na základě porovnávání těchto dat [21].

Jedná se o poměrně jednoduchou metodu, která obvykle používá již zmíněný indikátor síly signálu RSS. S touto metodou jsou spojené dvě velké nevýhody. První je, že se data z offline fáze stávají neplatná v případě, že je změněno prostředí. Druhou nevýhodou je, že data jsou měřena v diskrétních místech. Obvykle je tedy i poloha určována diskrétně, čímž vzniká chyba vždy, když se přijímač nenachází přesně v bodech původního měření. Chybu lze zmenšit zvýšením hustoty měřených dat v offline fázi, ale jen dokud nejsou sousední body kvůli odchylkám signálu zaměnitelné [15].

### 2.4.3 Proximity

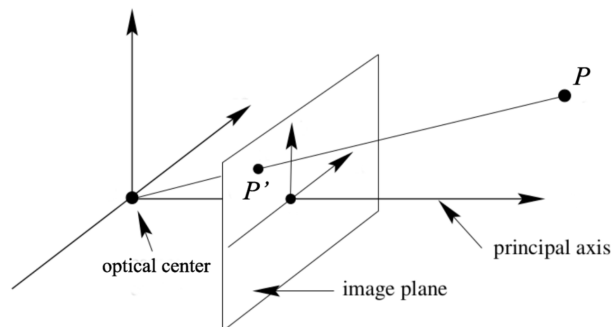
Tato metoda je, jak z názvu vyplývá, založená na určení, zda se přijímač nachází v blízkosti některého z vysílačů. Pokud přijímač zachytí signál od jednoho vysílače, lze předpokládat, že jsou zařízení blízko u sebe. V případě, že přijímač zachytí signál od více vysílačů, lze předpokládat, že je u toho, jehož síla signálu je největší. Tímto způsobem je zjištěna relativní poloha přijímače. Pro určení absolutní polohy je třeba vědět, kde přesně se nejbližší vysílač nachází [10].

Také existuje modifikace této metody s anglickým názvem Centroid. Přijímač se opět nachází v blízkosti několika vysílačů, nicméně nyní se poloha přijímače určí jako střední hodnota souřadnic vysílačů [17].

Tento způsob určování polohy má jako hlavní výhodu jednoduchost implementace. Nevýhodou naopak je, že k přesné lokalizaci přijímače je potřebná velmi hustá infrastruktura vysílačů.

#### 2.4.4 Vision analysis

Další způsob určování polohy je založen na geometrických vztazích mezi pozicí objektů zachycených na kameru a jejich skutečnou polohou v prostoru. Využívané geometrické vztahy jsou založené na dírkovém modelu kamery popsaném v práci [14], která u tohoto popisu vychází z knihy [22].



**Obrázek 2.6:** Dírkový model kamery.

Obrázek dírkového modelu ilustruje soustavu s bodem  $P$ , který se v rovině obrazu mapuje na  $P'$ . Předpokládejme, že bod  $P$  má v souřadnicovém systému kamery souřadnice  $(x', y', z')$  a v souřadnicovém systému celé soustavy  $(X, Y, Z)$ . Potom je možné se maticovým násobením transformovat ze souřadnic  $(X, Y, Z)$  na  $(x', y', z')$  pomocí rovnice

$$(x', y', z', 1)^T = \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix} (X, Y, Z, 1)^T, \quad (2.4)$$

kde  $R_{3 \times 3}$  je matice rotace a  $t_{3 \times 1}$  je vektor posunu. V knize [22] je zároveň ukázáno, že platí

$$(x, y, 1)^T = C(x', y', z', 1)^T = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} (x', y', z', 1)^T, \quad (2.5)$$

kde  $f$  je ohnisková vzdálenost kamery a  $(x, y, 1)^T$  jsou souřadnice bodu  $P'$  namapovaného do roviny kamery. Výsledný vztah mezi souřadnicemi bodu  $P$  a projekcí do roviny kamery je

$$(x, y, 1)^T = C \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix} (X, Y, Z, 1)^T. \quad (2.6)$$

Tento vztah lze využít k výpočtu polohy bodu  $P$  v prostoru a také tvoří základ pro počítačové vidění a aplikace rozšířené reality [14].

### 2.4.5 Hybrid algorithms

Tyto metody jsou založené na myšlence, že každá z dříve uvedených metod má své výhody a nevýhody a proto může být kombinace několika z nich přínosná pro zlepšení přesnosti systému. Například práce [23] navrhuje systém, u něhož je vzdálenost od světél určena kombinací RSS a TDOA. Jejich váženým součtem se následně určí poloha přijímače.

### 3 Systém pro testování metod VLP

Na základě teorie popsané v předchozí kapitole byly navrženy a implementovány dvě konkrétní metody určování polohy. Aby mohly být tyto metody implementovány a následně ověřeny, bylo nezbytné nejprve vytvořit systém, který zprostředkuje hardware a software vysílačů, hardware a software přijímače a nakonec také definuje přesné prostředí v němž budou metody testovány. Tato kapitola popisuje realizaci takového systému a zabývá se jeho jednotlivými částmi.

#### 3.1 Vysílače

Světla, která slouží jako vysílače pochází z prací [24] a [25]. Tyto práce se zabývaly návrhem a následnou konstrukcí jejich hardwaru i softwaru. Uvnitř světel se nachází napětím řízený zdroj proudu nastavující proud tekoucí skrz LED. Protože lze proud skrz LED považovat za přímo úměrný vyzařovanému světelnému toku, můžeme v tomto zapojení vysílat signály se stejným průběhem, jaké má řídicí napětí mikrokontroléru. Světla jsou konfigurovatelná pomocí internetové aplikace a umožňují vysílat signál s pulzně šířkovou modulací (PWM) nebo signál sinusového průběhu. U PWM lze nakonfigurovat frekvenci a duty cycle a sinusový průběh nabízí konfiguraci amplitudy, frekvence a fáze. Maximální nastavitelná frekvence se pohybuje v jednotkách kHz.



**Obrázek 3.1:** Světla využitá k lokalizaci.

V této práci byla světla nakonfigurována na vysílání sinusových signálů, přičemž na každém z nich byla nastavena odlišná frekvence. Konkrétně 400, 700 a 1000 Hz. Právě rozdíl vysílaných frekvencí byl využit při zpracování naměřených signálů k určování polohy, jak bude detailněji popsáno u konkrétní implementace metod.

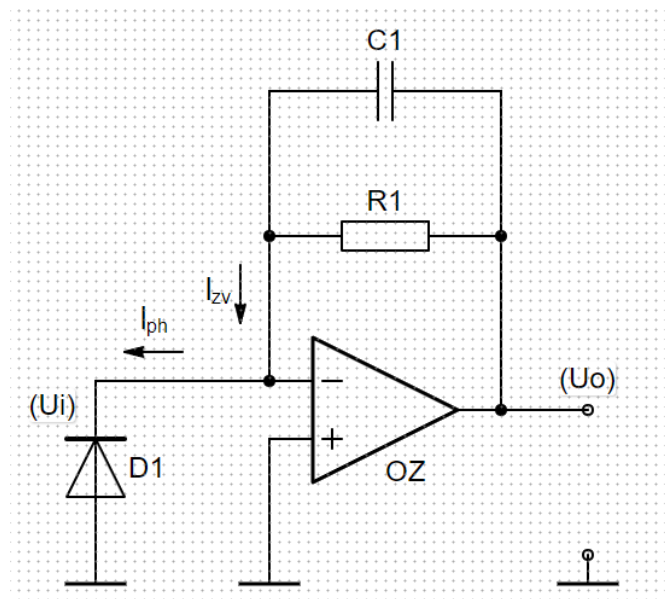
Nakonec je nutné zmínit, že světla nejsou synchronizovaná. Kvůli drobným odchylkám ve frekvencích se neustále mění fázový posun přijatých signálů. Z tohoto důvodu metody pro určování polohy nepracují přímo s navzorkovaným signálem, ale s amplitudovým spektrem diskrétní Fourierovy transformace. Při využití amplitudového spektra DFT nemají změny fáze vliv na vyhodnocení, díky čemuž není třeba se složitou synchronizací světel zabývat.

## 3.2 Robot s přijímačem

Dálkově ovládaný robot slouží k vyhodnocení funkčnosti implementovaných metod v reálném čase a také ke snímání signálů ze světla. Návrh a konstrukce podvozku, napájení a H-můstku pro řízení motorů robota jsou, stejně jako hardware světla, výsledkem práce [24]. Tyto části robota tedy nebudou dále diskutovány. Naopak obvod pro měření signálů ze světla a software mikrokontroléru vznikly jako součást této práce a budou popsány v následujícím textu.

### 3.2.1 Zpracování světelného signálu

Průběh světelného signálu je třeba nějakým způsobem zpracovat. V případě viditelného světla k tomuto účelu slouží převážně fotodiody a fototranzistory. V této práci byla využita fotodioda BPW34 [26]. Fotodiody generují elektrický proud, který bývá ve velké většině rozsahu přímo úměrný intenzitě osvětlení dopadající na oblast prostorového náboje. Tato závislost platí i pro použitou fotodiodu a díky ní zůstává průběh generovaného proudu shodný s průběhem dopadajícího signálu. Jedná se však o proud a ten je pro zpracování AD převodníkem mikrokontroléru nutné převést na napětí. K tomuto účelu byl navržen následující obvod.



Obrázek 3.2: TIA.

Jedná se o transimpedanční zesilovač (TIA), tedy obvod převádějící vstupní proud na výstupní napětí. Z předchozího popisu fotodiody je zřejmé, že v tomto zapojení je právě fotodioda stará o dodání vstupního proudu do obvodu. Výstup je připojený k AD převodníku mikrokontroléru ESP32. Pro celkovou impedanci ve zpětné vazbě obvodu lze psát

$$Z_{zv} = R_1 \parallel \frac{1}{j\omega C_1} = \frac{R_1 \frac{1}{j\omega C_1}}{R_1 + \frac{1}{j\omega C_1}} \quad [\Omega], \quad (3.1)$$

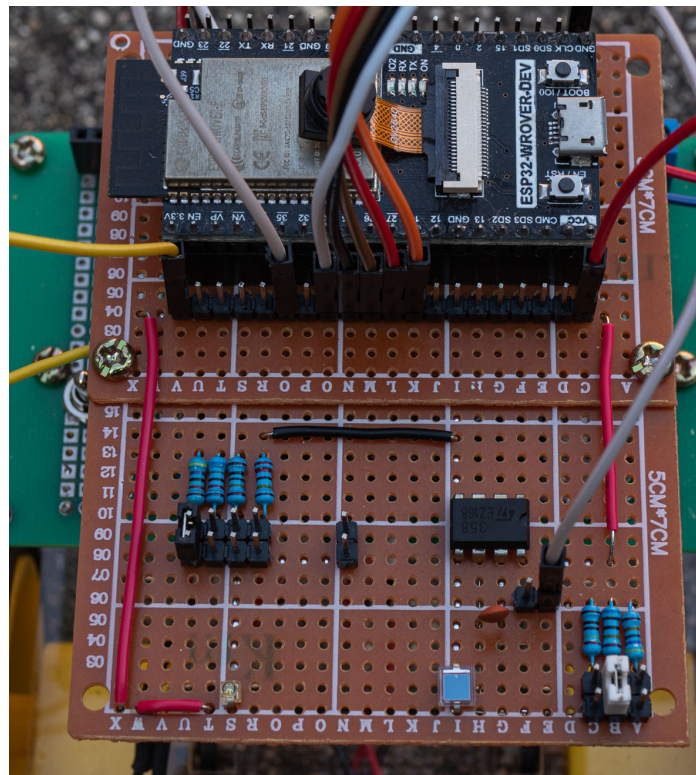
kde  $R_1$  je zpětnovazební odpor,  $C_1$  je zpětnovazební kapacita a  $\omega$  je úhlová frekvence. Dále z důvodu, že je zpětná vazba záporná platí

$$I_{ph} = I_{zv} \text{ [A] ,} \quad (3.2)$$

S využitím předchozích vztahů lze výstupní napětí  $U_o$  vyjádřit v podobě

$$U_o = I_{ph} \frac{R_1}{j\omega R_1 C_1 + 1} \text{ [V] .} \quad (3.3)$$

Kapacitor umístěný ve zpětné vazbě slouží k zajištění stability operačního zesilovače. Bez tohoto kapacitoru mívají transimpedanční zesilovače tendenci kmitat, což je způsobeno kapacitou PN přechodu diody a parazitních kapacit na vstupu operačního zesilovače. Matematický popis vzniku kmitání z pohledu frekvenčních charakteristik a jak se s ním vypořádat popisuje článek [27]. Zpětnovazební kapacitor dále slouží k omezení vysokofrekvenčního šumu, který by byl bez jeho přítomnosti, významně zesílen. Příčinou tohoto zesílení je, že kapacita na vstupu obvodu tvoří velký zisk na vysokých frekvencích [28]. Finální realizaci obvodu ukazuje obrázek 3.3.



**Obrázek 3.3:** Transimpedanční zesilovač připevněný na podvozku robota.

Popsaný obvod se nachází na pravé straně obrázku. Je zde vidět více zpětnovazebních rezistorů z nichž je možné si jeden zvolit pomocí jumperu. Pro stejnosměrné signály se kapacitor chová jako rozpojené svorky a tím pádem tento rezistor spolu se vstupním proudem přímo určují výstupní napětí. Velikost rezistoru musí být volena s ohledem na rozsah napětí AD převodníku, který se za obvodem nachází. V této práci



to byl AD převodník mikrokontroléru ESP32 jehož rozsah je 0–3.3 V. Přímou v dokumentaci knihoven, které implementují různé funkce pro práci s tímto AD převodníkem se však uvádí, že v horní a spodní části tohoto rozsahu není převodník lineární. Kvůli tomu byla hodnota zpětnovazebního rezistoru zvolena 680 k $\Omega$ , pro níž se signál většinu času pohyboval v rozsahu 0.5–2.5 V. Je nutné dodat, že při extrémní intenzitě denního světla by výstupní napětí vlivem takto velkého rezistoru mohlo dosáhnout velmi vysokých hodnot. To by mohlo vést až ke zničení AD převodníku. Ideálně by měl systém být schopný takovou situaci řešit například automatickou změnou velikosti ZV rezistoru. Prostředí v němž bylo testování prováděno má však poměrně malá okna a napětí úroveň 3.3 V nikdy nepřesáhlo. Dále je na obrázku vidět operační zesilovač LM358 [29]. U tohoto OZ výrobce udává, že lze dosáhnout nulového výstupního napětí i při napájení jedním zdrojem. Díky tomu nebylo třeba řešit dodatečné napájecí obvody a místo toho bylo využito napájení 0–5 V dostupné na vývojové desce s mikrokontrolérem ESP32. Nakonec je možné si všimnout keramického kondenzátoru o velikosti 22 pF, který se nachází ve zpětné vazbě. Tato velikost se na základě testování ukázala být nejvhodnější.

Na levé straně obrázku je vidět i druhý obvod s fototranzistorem OSRAM BPX 81 [30]. Tento fototranzistor má směrovější charakteristiku, díky níž měl být celý systém citlivější na změny polohy. Při měření však bylo zjištěno, že naměřený signál je závislý na směru natočení robota. Snaha připevnit tranzistor tak, aby k tomuto jevu nedocházelo byla neúspěšná. Z toho důvodu nebyl obvod při testování implementovaných metod využit a nebude dále diskutován.

### 3.2.2 Software mikrokontroléru

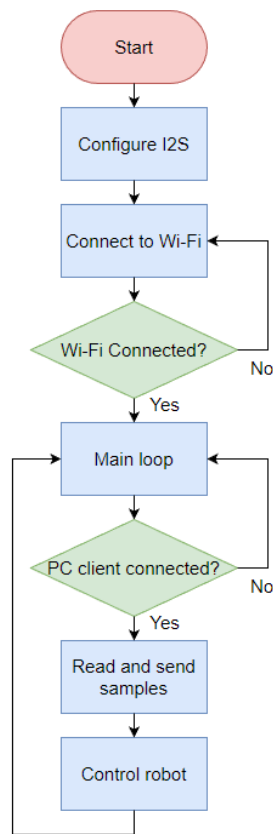
V této práci byl použitý mikrokontrolér ESP32 a to primárně kvůli WiFi konektivitě. Hlavní tři úlohy o něž se mikrokontrolér stará jsou vzorkování signálu na výstupu transimpedančního zesilovače, přenos dat do PC a řízení robota.

Pro vzorkování je na ESP32 možné použít I2S sběrnici nakonfigurovanou do ADC režimu [31]. V této konfiguraci je pro přenos dat využitý přímý přístup do paměti (DMA), díky čemuž je možné dosáhnout velkých vzorkovacích frekvencí, aniž by byl processor tímto vzorkováním zatěžován. Přenos dat do PC probíhá ve vytvořeném softwaru pomocí TCP serveru běžícího na ESP32. Poté, co se klient připojí, začne server posílat naměřené signály. Klient serveru naopak posílá data sloužící k řízení robota.

Celkový program pracuje následovně. Nejprve se inicializují proměnné, vytvoří WiFi server a nakonfiguruje I2S. Následně se mikrokontrolér připojí k WiFi síti a čeká na připojení klienta. Po jeho připojení se vyčtou vzorky signálu z DMA bufferu do paměti, pošlou se ve formě bytů do PC a dále se program po dobu 3 sekund stará o řízení robota na základě přijatých příkazů. Celý cyklus se opakuje dokud je klient připojený. Zjednodušené fungování znázorňuje vývojový diagram 3.4.

Hlavním nedostatkem tohoto programu je, že robot nemůže být ovládán ve chvíli kdy ESP32 čte vzorky z DMA bufferu a posílá je přes WiFi. Vzhledem k malému počtu odesílaných dat to však na jeho odezvu nemá velký vliv. V případě zvýšení množství posílaných dat by bylo vhodné zvážit jiné řešení.





Obrázek 3.4: Software běžící na ESP32.

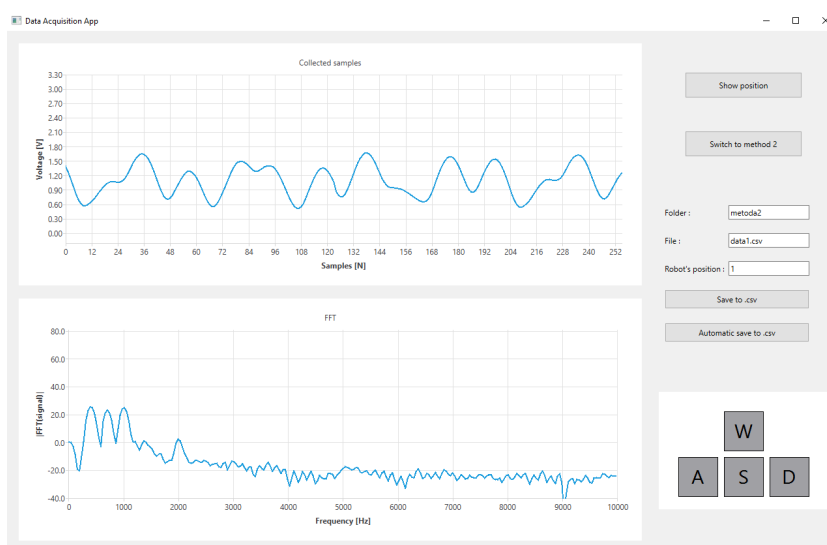
### 3.3 Aplikace vyhodnocující polohu

Pro vyhodnocování polohy byla vytvořena počítačová aplikace. Hlavním důvodem k rozhodnutí, že poloha bude určována v počítači místo mikrokontroléru je jazyk Python. U tohoto jazyka existuje mnoho veřejně dostupných knihoven, jejichž využití významně zjednodušilo celou implementaci. V této podkapitole bude krátce nastíněno jak vzniklá aplikace funguje a následně bude popsáno co aplikace nabízí z pohledu uživatele. Celý kód je dostupný na [32].

Po spuštění se aplikace připojí k TCP serveru běžícímu na ESP32 a začne přijímat vzorky signálu. Vždy po obdržení celé jedné posloupnosti vzorků zobrazí momentálně naměřený signál, jeho DFT a také vypočítá momentální polohu robota pomocí metod popsaných v následujících kapitolách. V průběhu tohoto opakujícího se děje může uživatel ukládat změřená data. K tomu je využita knihovna Pandas [33]. Pandas obecně slouží k analýze dat, ale obsahuje také funkce `read_csv` a `save_csv`, pomocí nichž lze data načíst ve formátu CSV nebo je v tomto formátu ukládat. Dále může uživatel písmeny W, S, A a D na klávesnici ovládat robota. V daný moment však může být aktivní pouze jeden směr. Grafické rozhraní je vytvořené pomocí frameworku Qt pro Python [34]. V jeho levé části se nachází 2 grafy. Horní z nich zobrazuje navzorkovaný signál a spodní amplitudové DFT spektrum tohoto signálu. Pravá část grafického rozhraní obsahuje tlačítka a editovatelné řádky s následujícími funkcemi.

- **Show position/Show collected samples** – Slouží k přepínání mezi dvěma zobrazeními aplikace. U prvního z nich je v aplikaci zobrazen poslední naměřený signál a u druhého okno s poslední odhadnutou polohou robota. Vzhled okna pro zobrazení polohy se liší podle použité metody a jsou proto ukázána až u jednotlivých implementací.
- **Switch to method 1/Switch to method 2** – Umožňuje přepínat mezi tím, jaká metoda je momentálně používána k určení polohy.
- **Folder** – Editovatelný řádek vpravo slouží k určení do jakého adresáře se data uloží. Pokud adresář neexistuje, bude vytvořen.
- **File** – Editovatelný řádek vpravo udává jméno souboru do něhož se naměřené vzorky uloží. Pokud soubor neexistuje, bude vytvořen a vyplněn jedním řádkem naměřených vzorků. Pokud soubor existuje, bude za poslední data přidán nový řádek s momentálně změřenými vzorky signálu.
- **Robot's position** – Přidá do posledního sloupce ke vzorkům informaci o poloze v níž se robot nachází. Tento poslední sloupec sloužil primárně u metody založené na strojovém učení, u níž je pro trénování a testování modelu třeba znát správné výsledky.
- **Save to .csv** – Uloží poslední naměřený signál do souboru ve formátu CSV.
- **Automatic save to .csv** – Automaticky ukládá data dokud uživatel ukládání nezruší. V kódu lze také nastavit konstantu `AUTO_MEASUREMENT_RANGE`, která určí maximální množství uložených řádků.

Ve spodní polovině pravé části aplikace se nakonec nachází widget zobrazující momentální směr pohybu robota. Písmeno odpovídající momentálnímu směru pohybu robota je barevně zvýrazněno.



**Obrázek 3.5:** Hlavní okno aplikace vyhodnocující polohu.

### 3.4 Testovací prostředí

Poslední nezbytnou komponentou k otestování navržených metod je jasně stanovené testovací prostředí. Testování probíhalo v garáži se 4 malými okny. Celkový vliv venkovního světla byl tedy o něco menší než v běžné místnosti. Na garážovou konstrukci byly zavěšeny dva trámy k jimž se světla následně připevnila pomocí stahovacích pásek. Každé z nich bylo zavěšeno ve výšce 212 cm nad zemí. Vzhledem k dostupnosti tří světél bylo zvoleno jejich rozmístění do rovnostranného trojúhelníku se vzdáleností 1 metr mezi vrcholy. Nakonec byl pod světly vytyčen další rovnostranný trojúhelník, který ohraničoval oblast, v níž všechno měření probíhalo. Jeho vrcholy se nacházely přímo pod světly a délka strany byla také přesně 1 m.



Obrázek 3.6: Zavěšení světél.



Obrázek 3.7: Uspořádání světél.



## 4 Lokalizace algoritmem KNN

Tato kapitola se zabývá implementací první metody pro určování polohy pomocí komunikace ve viditelném světle. Z hlediska teoretického rozdělení lokalizačních metod uvedeného v 2. kapitole by se tato dala zařadit do kategorie fingerprinting. Je založená na algoritmu K-Nearest Neighbors, který spadá pod strojové učení.

Nejprve je strojové učení krátce představeno, dále je vysvětleno jak funguje algoritmus K-Nearest Neighbors a nakonec je metoda implementována a otestována.

### 4.1 Strojové učení

Umělá inteligence nabývá v posledních letech na své popularitě, protože dokáže řešit mnohé úlohy, jejichž řešení by jinak bylo velmi obtížné nebo zdlouhavé.

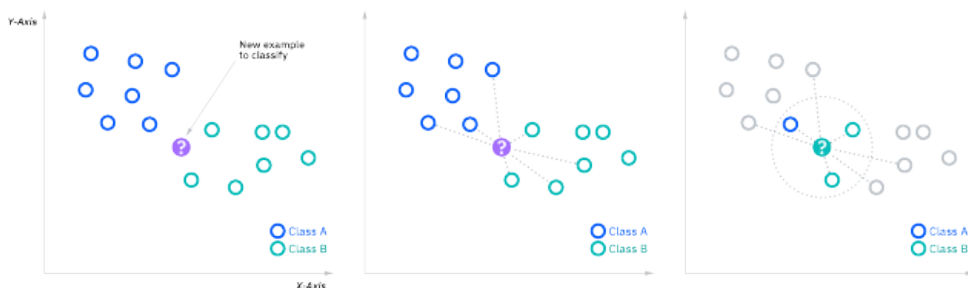
Strojové učení, o němž je tato kapitola, je podmnožinou umělé inteligence. Jedná se o algoritmy, které jsou schopné se učit z nasbíraných dat pomocí metod statistické analýzy a na základě toho hledat souvislosti a vyvozovat závěry. Nejčastěji se strojové učení rozděluje podle způsobů učení na Supervised, Unsupervised, Semi-Supervised a Reinforced. Následující popis vychází z článku [35].

- **Supervised learning (učení s učitelem)** – Jedná se o způsob, u něhož jsou při učení známé jak vstupní, tak i výstupní data. Odhadované výsledky jsou při učení porovnávány s těmi korektními a podle toho je upravován výsledný model. Po natrénování je metoda schopná sama určovat výsledná data i bez znalosti správných výsledků.
- **Unsupervised learning (učení bez učitele)** – Způsob učení, kde jsou k dispozici pouze vstupní data. Algoritmy se snaží v datech hledat strukturu danou podobnostmi nebo rozdíly a podle toho je určitým způsobem třídít.
- **Semi-Supervised learning** – Algoritmy, které kombinují učení s učitelem a učení bez učitele. Obvykle je k dispozici malé množství dat, u nichž jsou předem známé výstupní hodnoty, ale u zbytku výstupy neznáme.
- **Reinforced learning (zpětnovazební učení)** – Učení, založené na zpětné vazbě. Do určitého prostředí je přidán počítačový program nazývaný agent. Agent s prostředím interaguje a vykonává určité úkoly. Na základě úspěchů a neúspěchů je následně odměňován nebo trestán. Cílem agenta je dosažení co nejvyšších odměn.

Celá problematika je velmi rozsáhlá, kvůli čemuž se tato práce dále zaměří pouze na dříve zmíněný algoritmus K-Nearest Neighbors, který patří do skupiny učení s učitelem. Bude vysvětleno jak tento algoritmus pracuje a následně popsán konkrétní způsob jeho využití pro určování polohy.

## 4.2 Algoritmus KNN

K-Nearest Neighbors (KNN) je algoritmus klasifikující objekt na základě třídy jeho nejbližších sousedů. Lze ho sice použít i pro úlohu regrese, ale typicky se jedná o klasifikátor. Mějme množinu bodů v prostoru, u níž platí, že body nacházející se blízko sebe patří do stejné třídy. Předpokládejme, že získáme nový bod, u něhož známe polohu v prostoru, ale neznáme třídu do níž patří. S využitím znalosti, že body stejné třídy se nachází blízko sebe, jsme schopni odhadnout třídu tohoto bodu podle jeho polohy [36]. Popsaná situace je ukázána na obrázku 4.1, který demonstuje využití tří nejbližších sousedů pro určení třídy.



**Obrázek 4.1:** Myšlenka KNN. Převzato z [36].

Pro co nejlepší odhad třídy, do níž bod spadá, jsou důležité dva parametry. Prvním z nich je množství nejbližších sousedů, na základě jichž odhad provádíme. Výsledná volba je dána třídou s největším zastoupením mezi sousedy. Tento způsob volby je v anglické literatuře nazýván „plurality voting“ a nevyžaduje nadpoloviční většinu hlasů. Klasifikace na základě 6 nebo 20 nejbližších sousedů může dávat velmi odlišné výsledky a vhodné množství závisí na konkrétních datech. Druhým důležitým parametrem je metrika definující způsob výpočtu vzdálenosti. Rovnice 4.1 popisuje Minkowského metriku

$$D(X, Y) = \left( \sum_{i=1}^n |x_i - y_i|^p \right)^{\frac{1}{p}}, \quad (4.1)$$

kde  $X = (x_1, x_2, \dots, x_n)$ ,  $Y = (y_1, y_2, \dots, y_n)$  jsou body v  $n$ -dimenzionálním prostoru a  $p \geq 1$  je reálné číslo. Pro  $p = 2$  tato metrika odpovídá běžně známé Euklidovské metrice

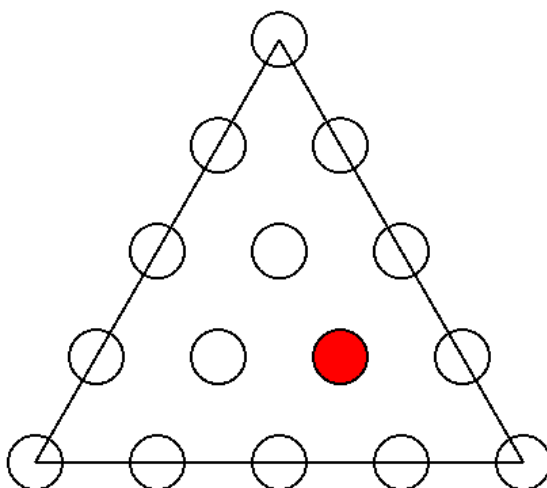
$$D(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}. \quad (4.2)$$

## 4.3 Implementace a testování

### 4.3.1 Implementace

Při implementaci metody byly využity veřejně dostupné knihovny Scikit-learn [37], SciPy [38], a Numpy [39]. Scikit-learn je zaměřená na strojové učení. Z hlediska této metody je důležité, že poskytuje implementovaný KNN klasifikátor. Knihovna SciPy zase obsahuje různé matematické algoritmy, jedním z nich je i algoritmus FFT využitý pro transformaci do frekvenční domény. Poslední z knihoven Numpy slouží pro práci s maticemi a vícedimenzionálními poli. Pomocí ní byla data předzpracována.

Z teoretického popisu je zřejmé, že klasifikace určuje diskretní třídu. Kvůli tomu je tedy i poloha určována pouze v diskretních místech. Pro natrénování modelu bylo zvoleno 15 bodů v prostoru. Ty byly rozmístěny tak, aby vzdálenost každého z bodů byla ke všem jeho sousedům co nejrovnoměrnější. Umístění zároveň vychází z dříve popsaného uspořádání měřicího prostředí. Z 15 bodů se 12 nachází po obvodu rovnostranného trojúhelníku a zbylé 3 uvnitř. Obrázek 4.2 zobrazuje přesné polohy bodů. Zároveň tento obrázek pochází z počítačové aplikace, v níž slouží jako okno zobrazující odhad momentální polohy robota.



**Obrázek 4.2:** Body v nichž byl měřen signál pro trénování metody. Zároveň se jedná o okno z aplikace zobrazující polohu.

Pro natrénování bylo naměřeno 50 průběhů signálu v každém ze zvolených bodů. Všechny tyto signály byly pomocí počítačové aplikace, která byla popsána ve 3. kapitole, uloženy do CSV souboru. Soubor se získanými daty je k dispozici na [32]. Po nasbírání potřebných dat probíhalo zpracování. Nejprve proběhlo odstranění stejnosměrné složky signálu. Tato složka se mění s intenzitou denního světla a model, který by jí využíval k určení polohy by nefungoval za různých světelných podmínek. Dále byly všechny průběhy signálu transformovány pomocí FFT z časové domény do frekvenční. Důvodem pro vyhodnocování polohy na základě FFT je, že světla nejsou navzájem synchronizovaná. Vlivem drobných změn frekvence řídicích mikrokontrolérů světla dochází ke změnám fázového posunu. Signály naměřené v jedné poloze byly kvůli tomuto

jevu zcela odlišné a přesné vyhodnocení algoritmem KNN poskytovalo nepoužitelné výsledky. Využití amplitudového spektra diskrétní Fourierovy transformace tento problém vyřešilo a umožnilo jednoznačně určit polohu robota. Po transformaci signálu probíhalo samotné trénování modelu KNN. Při trénování byl použit klasifikátor s Euklidovskou metrikou. Z hlediska dosažené přesnosti se ukázala být nejlepší volba 6 nejbližších sousedů. Pomocí nich tedy byla poloha odhadována.

### 4.3.2 Dosažené výsledky

První způsob testování zjišťoval, jak spolehlivě dokáže model určit polohu, pokud se robot nachází v jednom z 15 bodů, v nichž byl model natrénován. K vyhodnocení přesnosti byly využity tři sady dat. První z nich vznikla při rozdělení původně naměřených dat do trénovací a testovací kategorie. Tato sada měla oproti dalším dvěma výhodu, že vznikla za stejných světelných podmínek jako sada trénovací. Zbylé dvě sady byly naměřeny v odlišných dnech a v různých hodinách. Důvodem pro měření v různou dobu byla právě snaha otestovat systém za odlišných světelných podmínek, které by mohly mít velmi významný negativní vliv na správné určení polohy. Nakonec se však ukázalo, že rozdílné světelné podmínky na odhad polohy vliv nemají, kvůli odečtení stejnosměrné složky. Výsledky tohoto měření ukazuje následující tabulka.

Měření	Úspěšnost odhadu polohy [%]
č.1	0.99555
č.2	1.0
č.3	0.99333

**Tabulka 4.1:** Tabulka s dosaženými výsledky.

Natrénovaný model dosáhl tímto testováním přesnosti 25 cm s téměř 100% úspěšností. Vzhledem k dosažení vysoké přesnosti na testovanou vzdálenost by bylo vhodné zhustit síť bodů, v nichž byl signál natrénován. To však vyžaduje velké množství času a nebylo provedeno.

Druhý způsob testování hodnotil vytvořený model spíše subjektivně. Nezkoumal konkrétní číselné výsledky, ale snažil se zhodnotit funkčnost na základě pozorování. Výsledkem je video dostupné na [40].



## 5 Lokalizace metodou DFT triangulace

Druhá z implementovaných metod je z teoretického pohledu založená na triangulaci. Jako v předchozím případě, je i zde nejprve popsán princip na němž metoda stojí a následně její implementace.

### 5.1 Princip metody

Myšlenka metody je založená na předpokladu, že signál klesá se vzdáleností od světél. Dále metoda vyžaduje, aby tento pokles nebyl závislý na směru. Využívaná světla splňují oba tyto předpoklady. Problém by mohl nastat v případě, že se světla umístí blízko předmětu, od něhož by se signál odrazil. To však vzhledem ke zvolenému testovacímu prostředí nehrozilo, protože v okolí světél se žádné stěny či jiné předměty nenacházely.

Ze znalosti, že pokles signálu nezávisí na směru vyplývá, že se robot musí nacházet na kružnici, jejíž poloměr závisí na síle přijatého signálu. Rovnice kružnice má nejčastěji podobu

$$(x - a)^2 + (y - b)^2 = r^2, \quad (5.1)$$

kde  $[a, b]$  jsou souřadnice středu kružnice,  $r$  je její poloměr a  $[x, y]$  jsou body v prostoru ležící na kružnici pokud je rovnost splněna. Pomocí přijatého signálu složeného ze tří sinusových průběhů lze určit tři kružnice, na nichž se robot nachází. Tyto kružnice by se v ideálním případě protuly v jednom bodě a tím by byla zjištěna přesná poloha robota. Tento případ však téměř nikdy nenastává. Typický je vznik situace, kdy se kružnice protnou ve více bodech, nebo naopak v žádném. Pokud se kružnice protnou, pak je možné využít blízké průsečíky a pomocí nich odhadovat polohu. Nicméně, když se kružnice neprotnou, tak není k dispozici žádný průsečík a neexistuje jednoduchý způsob určení polohy. Z toho důvodu bude využit jiný postup, který pochází z [41]. Nejprve zavedeme funkci

$$f(x, y) = (x - a)^2 + (y - b)^2 - r^2. \quad (5.2)$$

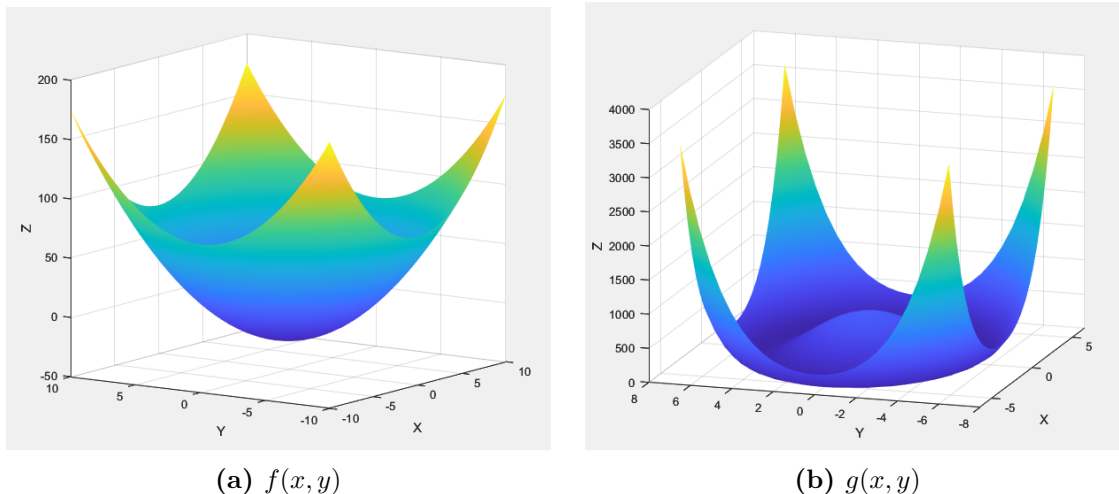
V ní je původní kružnice reprezentována řezem 3D grafu rovinou  $z = 0$ . Graf této funkce je uveden na obrázku 5.1. Dále funkci  $f(x, y)$  umocníme a získáme

$$g(x, y) = ((x - a)^2 + (y - b)^2 - r^2)^2. \quad (5.3)$$

I v této funkci je původní kružnice reprezentovaná řezem skrz rovinu  $z = 0$ . Graf se z pohledu určení polohy změnil v tom, že nyní se tato kružnice nachází v minimu funkce  $g(x, y)$ . Robot tedy nachází někde v minimu této funkce. Na základě toho definujeme

$$E(x, y) = \sum_{i=0}^n (x - a_i)^2 + (y - b_i)^2 - r_i^2, \quad (5.4)$$

kde  $[a_i, b_i]$  jsou polohy, v nichž se světla nachází a  $r_i$  jsou vzdálenosti od jednotlivých světél, tj. poloměry kružnic a  $[x, y]$  je neznámá poloha robota.



**Obrázek 5.1:** Funkce  $f(x, y)$  a  $g(x, y)$ .

Funkce  $E(x, y)$  vyjadřuje součet kvadrátů chyb určené polohy. Kdyby nastala situace, že se všechny kružnice protnou právě v jednom bodě, pak tato funkce nabývá hodnoty 0 a v tomto průsečíku se nachází robot. Jak již bylo zmíněno, toto nenastane téměř nikdy. Pokud však budou odhadnuté kružnice nabývat takových poloměrů, aby se od ideálního průniku o moc nevzdalovaly, lze předpokládat, že minimum této funkce bude odhadovat skutečnou polohu robota s minimální chybou. Minimum této funkce lze nalézt vyřešením soustavy rovnic

$$\frac{\partial E}{\partial x}(x, y) = 0, \quad \frac{\partial E}{\partial y}(x, y) = 0. \quad (5.5)$$

## 5.2 Implementace metody

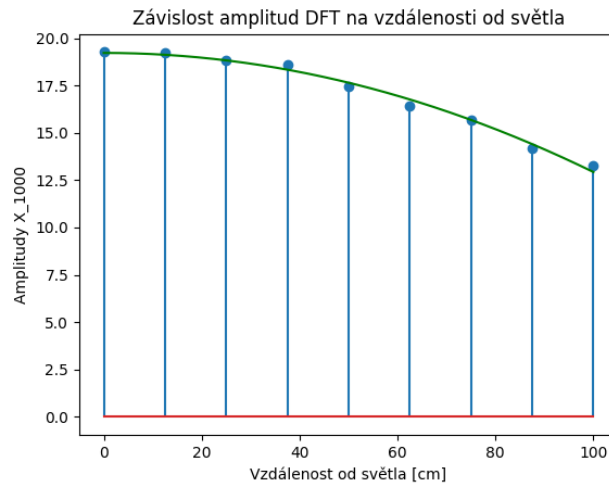
Metoda byla opět implementována v jazyce Python a zakomponována do počítačové aplikace. Znovu bylo také využito amplitudového spektrum diskretní Fourierovy transformace z důvodu nesynchronizovaných vysílačů. Z teoretického popisu této metody vyplývá, že prvním úkolem bylo zjistit, jaký je průběh poklesu amplitud spektrálních čar v závislosti na horizontální vzdálenosti robota od světla. Světla jsou nakonfigurována na vysílání sinusového signálu, takže by se po odstranění stejnosměrné složky měl ideálně veškerý výkon od konkrétního světla nacházet pouze v jedné spektrální čáře. Toto však určitě není zcela pravda s ohledem na přítomnost šumu, spektrálního prosačování a drobných nelinearit v celém systému. Platit ale bude, že se velká část výkonu nachází na spektrální čáře odpovídající frekvenci vysílajícího světla a právě změny její velikosti budou určovat vzdálenost robota od světla.

Bylo tedy provedeno měření, při kterém byl robot postupně vzdalován od jednotlivých světla s krokem 12.5 cm. V každé vzdálenosti byl naměřen signál, z něj vypočteno DFT spektrum a nakonec vyčtena spektrální čára na frekvenci světla od něhož se robot vzdaluje. Výsledkem pro světlo vysílající na frekvenci 1000 Hz jsou hodnoty amplitud DFT spektra zobrazené v grafu 5.2. Velmi podobné grafy vznikly i pro zbylá dvě světla.

Jejich průběhy byly téměř totožné, nicméně maximální a minimální hodnoty se lehce lišily. Dále bylo třeba najít funkci, která bude procházet skrz získané grafy. Jako nejlepší volba se ukázala být funkce

$$X = -aR^2 + b \quad [-], \quad (5.6)$$

kde  $X$  je hodnota amplitudy DFT spektra na vysílací frekvenci některého ze světél,  $R$  je horizontální vzdálenost robota od světla a  $a$ ,  $b$  jsou neznámé koeficienty této funkce. K nalezení těchto koeficientů byla využita funkce `curve_fit`, kterou implementuje knihovna Scipy. Tato funkce využívá nelineární metodu nejmenších čtverců, pomocí níž dokáže neznámé koeficienty určit tak, aby funkce 5.6 co nejlépe odpovídala naměřeným hodnotám. Průběh vypočtené funkce pro světlo vysílající na frekvenci 1000 Hz je s již vypočtenými koeficienty opět zobrazen v grafu 5.2.

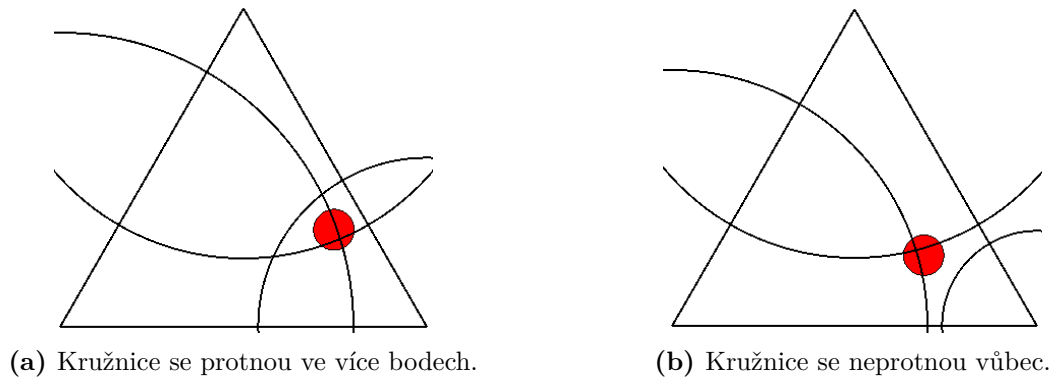


**Obrázek 5.2:** Závislost amplitudy DFT vzdalujícího se světla na vzdálenosti.

Naše situace je však obrácená. Hodnoty amplitud DFT spektra známe, ale vzdálenost od světél ne. Předchozí funkci tedy musíme pro výpočet odhadu polohy upravit do podoby

$$R = \sqrt{\frac{b - X}{a}} \quad [\text{cm}]. \quad (5.7)$$

Rovnici 5.7 s příslušnými koeficienty je dále třeba aplikovat na amplitudy získané ze všech tří světél, čímž získáme kružnice o různých poloměrech. Jak popisuje teoretická část, tyto kružnice se téměř nikdy neprotnou v jednom bodě a k určení polohy robota je využito minimum funkce  $E(x, y)$ . O nalezení tohoto minima se stará funkce `minimize` z knihovny Scipy. Obrázek 5.3 pochází z dříve popsané počítačové aplikace, v níž slouží k zobrazení polohy robota. Zde zobrazuje dvě situace, které mohou nastat. Vlevo je vidět případ, v němž se kružnice protnou ve více bodech. Vpravo zase případ, kdy se kružnice neprotnou vůbec. Lze vidět, že metoda založená na nalezení minima funkce  $E(x, y)$  si umí dobře poradit i se situací, kdy kružnice žádný průnik nemají, což by metodou využívající blízké průsečíky kružnic nebylo možné.



**Obrázek 5.3:** Odhady poloh.

Tímto měla být navržená metoda dokončená. Nicméně se ukázalo, že vypočtená hodnota amplitudy DFT spektra se mění i v případě, že se robot nachází na stále stejném místě. Tyto změny nebyly zanedbatelné a z toho důvodu bylo ještě dodatečně implementováno průměrování dvou vypočtených amplitud. Ideální by bylo průměrování provádět z více hodnot, to však při implementaci vedlo k dlouhé době přenosu dat z mikrokontroléru do počítače. Pokud se robot při přenosu pohyboval, pak byla poloha určena nejen z dat, kde se robot skutečně nachází, ale také z dat naměřených při cestě do této polohy. Výsledky byly dobré pouze pokud se robot nepohyboval vůbec, což je u systému pro určování polohy velmi nevhodné. Průměrování dvou získaných amplitud bylo tedy zvoleno jako kompromis mezi těmito problémy.

### 5.3 Dosažené výsledky

Po dokončení implementace byla navržená metoda testována. Testování probíhalo tak, že byl robot postaven do 30 náhodně zvolených míst, která se nacházela uvnitř měřicího trojúhelníku. V každém z nich byla změřena skutečná poloha robota a také zapsána poloha vypočtená implementovanou metodou. Kompletní tabulka naměřených dat je uvedena v příloze. Ze získaných hodnot byly vypočítány euklidovské vzdálenosti mezi skutečnými a odhadnutými polohami robota a z nich určena průměrná a maximální odchylka polohy.

Průměrná odchylka [cm]	Maximální odchylka [cm]
6.27	13.77

**Tabulka 5.1:** Velikost chyby určení polohy.

Nakonec byla metoda ověřena v reálném čase. Stejně jako u předchozí metody vzniklo video dostupné na [42]. Zhodnocení metody na základě videa je subjektivní, ale nezdá se, že by dosažená přesnost byla tak vysoká, jak jí uvedlo měření. Problémem by mohlo být chvění fotodiody způsobené pohybem robota nebo již dříve zmíněné průměrování.

## 6 Závěr

Práce představila a vysvětlila existující metody určování polohy pomocí komunikace ve viditelném světle. Dále byl popsán hardware vysílačů, hardware robota sloužícího pro otestování metod, navržen obvod pro příjem světelného signálu a nakonec naprogramován software, včetně dvou metod pro určení polohy, které jsou založené na strojovém učení a triangulaci. První z metod využila algoritmus K-Nearest Neighbors (KNN). V první fázi byl v měřícím prostředí naměřen soubor dat s jejichž pomocí byl natrénován KNN model. S využitím natrénovaného modelu následně probíhalo testování. Ukázalo se, že metoda funguje spolehlivě a není ovlivněna měnící se intenzitou denního světla. Dosažená přesnost je 25 cm s více než 99% pravděpodobností správného vyhodnocení polohy. Vzhledem k vysoké úspěšnosti na zvolenou vzdálenost by mohlo být zajímavé zavést hustší síť bodů měření, natrénovat nový model a zkoumat jak se přesnost vyhodnocení změní. Sběr dat je však časově náročný, kvůli čemuž nebyl tento pokus proveden a zůstává pouze jako návrh pro budoucí výzkum. Druhá z implementovaných metod je založená na triangulaci. Byla nalezena funkce popisující závislost amplitud DFT spektra na vzdálenosti. Na základě toho se určuje vzdálenost od všech tří světél a nalezením minima představené funkce  $E(x, y)$  se odhaduje poloha. Maximální naměřená chyba odhadu byla 13.77 cm a průměrná chyba 6.27 cm.

Práce dosáhla velmi dobrých výsledků a demonstrovala, že komunikace ve viditelném světle je velmi dobrý kandidát pro určování polohy v budovách. Existuje však několik důležitých slabin, které by bylo třeba před zavedením obdobného systému do praxe doladit. Hlavní slabinou je, že poloha je určována pouze ve 2D. To nemusí vadit v případě využití u robota, který má přijímač nainstalovaný v konkrétní výšce nad zemí, ale pro využití lidmi to bude problém. Každý člověk bude mít mobilní telefon v jiné výšce, a tím pádem naměří i jinak silný signál. S tím také souvisí předpoklad, že člověk neponese mobilní telefon rovnoběžně se zemí, čímž bude opět ovlivněna síla přijatého signálu. Další problém plyne z toho, že se světelný signál šíří téměř výhradně na přímou viditelnost. I malá překážka signál zablokuje nebo alespoň silně utlumí. Pro praxi by bylo vhodné, aby se systém uměl přizpůsobit výpadku jednoho nebo více příchodících signálů a správně určoval polohu i bez nich. Řešení těchto problémů představuje výzvu pro budoucí výzkum, který by na tuto nebo podobné práce mohl navázat.



## Použitá literatura

- [1] 27 Use Cases for Indoor Mapping You Might Not Know About. In: *Mappedin* [online]. Waterloo, Ontario: Mappedin, Jan 13, 2023 [cit. 2023-05-11]. Dostupné z: <https://www.mappedin.com/blog/product/indoor-mapping/use-cases-for-indoor-mapping/>
- [2] NAKAJIMA, Madoka a Shinichiro HARUYAMA. New indoor navigation system for visually impaired people using visible light communication. *EURASIP Journal on Wireless Communications and Networking* [online]. 2013, 19 February 2013, 2013(1) [cit. 2023-05-15]. ISSN 1687-1499. Dostupné z: doi:10.1186/1687-1499-2013-37
- [3] RENAUDIN, Valérie, Okan YALAK, Phillip TOMÉ a Bertrand MERMINOD. Indoor Navigation of Emergency Agents. *European Journal of Navigation*. 2007, 5(3), 36-45.
- [4] *GPS.gov* [online]. Washington, D.C.: National Coordination Office for Space-Based Positioning, Navigation, and Timing [cit. 2023-05-11]. Dostupné z: <https://www.gps.gov/>
- [5] KOS, Tomislav, Ivan MARKEZIC a Josip POKRAJCIC. Effects of multipath reception on GPS positioning performance. In: *ELMAR-2010* [online]. Zadar, Croatia: IEEE, 2010, s. 399-402 [cit. 2023-05-17]. ISBN 978-1-4244-6373-2. ISSN 1334-2630. Dostupné z: <https://ieeexplore.ieee.org/document/5606130>
- [6] VAN DIGGELEN, F. Indoor GPS theory & implementation. In: *2002 IEEE Position Location and Navigation Symposium (IEEE Cat. No.02CH37284)* [online]. Palm Springs, California: IEEE, 2002, s. 240-247 [cit. 2023-05-15]. ISBN 0-7803-7251-4. ISSN 2001095311. Dostupné z: doi:10.1109/PLANS.2002.998914
- [7] KRIZ, Pavel, Filip MALY a Tomas KOZEL. Improving Indoor Localization Using Bluetooth Low Energy Beacons. *Mobile Information Systems* [online]. 2016, 2016, 1-11 [cit. 2023-05-15]. ISSN 1574-017X. Dostupné z: doi:10.1155/2016/2083094
- [8] CHENG, Yun a Taoyun ZHOU. UWB Indoor Positioning Algorithm Based on TDOA Technology. In: *2019 10th International Conference on Information Technology in Medicine and Education (ITME)* [online]. Qingdao, China: IEEE, 2019, 2019, s. 777-782 [cit. 2023-05-17]. ISBN 978-1-7281-3918-0. ISSN 2474-3828. Dostupné z: doi:10.1109/ITME.2019.00177
- [9] LEE, Sangjae, Seungwoo CHAE a Dongsoo HAN. ILoA: Indoor Localization Using Augmented Vector of Geomagnetic Field. *IEEE Access* [online]. 2020, 8, 184242-184255 [cit. 2023-05-17]. ISSN 2169-3536. Dostupné z: doi:10.1109/ACCESS.2020.3029281
- [10] ZHUANG, Yuan, Luchi HUA, Longning QI, et al. *A Survey of Positioning Systems Using Visible LED Lights* [online]. 2018, 20(3), 1963-1988 [cit. 2023-05-17]. ISSN 1553-877X. Dostupné z: doi:10.1109/COMST.2018.2806558

- [11] KARUNATILAKA, Dilukshan, Fahad ZAFAR, Vineetha KALAVALLY a Rajendran PARTHIBAN. *LED Based Indoor Visible Light Communications: State of the Art* [online]. 2015, 17(3), 1649-1678 [cit. 2023-05-17]. ISSN 1553-877X. Dostupné z: doi:10.1109/COMST.2015.2417576
- [12] JUNG, Soo-Yong, Swook HANN a Chang-Soo PARK. TDOA-based optical wireless indoor localization using LED ceiling lamps. *IEEE Transactions on Consumer Electronics* [online]. 2011, 57(4), 1592-1597 [cit. 2023-05-17]. ISSN 0098-3063. Dostupné z: doi:10.1109/TCE.2011.6131130
- [13] FARAHNEH, Hasan, Christopher MEKHIEL, Ala KHALIFEH, Wisam FARJOW a Xavier FERNANDO. Shadowing effects on visible light communication channels. In: *2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)* [online]. Vancouver, BC, Canada: IEEE, 2016, 2016, s. 1-5 [cit. 2023-05-17]. ISBN 978-1-4673-8721-7. Dostupné z: doi:10.1109/CCECE.2016.7726717
- [14] DO, Trong-Hop a Myungsik YOO. An in-Depth Survey of Visible Light Communication Based Positioning Systems. *Sensors* [online]. 2016, 16(5) [cit. 2023-05-17]. ISSN 1424-8220. Dostupné z: doi:10.3390/s16050678
- [15] ZAFARI, Faheem, Athanasios GKELIAS a Kin K. LEUNG. *A Survey of Indoor Localization Systems and Technologies* [online]. IEEE, 2019, 21(3), 2568-2599 [cit. 2023-05-17]. ISSN 1553-877X. Dostupné z: doi:10.1109/COMST.2019.2911558
- [16] CHEN, Pengpeng, Mingzhi PANG, Di CHE, Yuqing YIN, Donghai HU, Shouwan GAO a Ghufuran AHMED. A Survey on Visible Light Positioning from Software Algorithms to Hardware. *Wireless Communications and Mobile Computing* [online]. 2021, 2021, 1-20 [cit. 2023-05-17]. ISSN 1530-8677. Dostupné z: doi:10.1155/2021/9739577
- [17] ISAIA, Constantina a Michalis P. MICHAELIDES. A Review of Wireless Positioning Techniques and Technologies: From Smart Sensors to 6G. *Signals* [online]. 2023, 4(1), 90-136 [cit. 2023-05-17]. ISSN 2624-6120. Dostupné z: doi:10.3390/signals4010006
- [18] Object Tracking Using Time Difference of Arrival (TDOA). In: *MathWorks* [online]. The MathWorks, c1994-2023 [cit. 2023-05-17]. Dostupné z: <https://www.mathworks.com/help/fusion/ug/object-tracking-using-time-difference-of-arrival.html>
- [19] Fotometrie a radiometrie. In: *Fyzikální ústav UK* [online]. Praha [cit. 2023-05-17]. Dostupné z: [http://fu.mff.cuni.cz/biomolecules/media/files/courses/Fotometrie\\_a\\_radiometrie.pdf](http://fu.mff.cuni.cz/biomolecules/media/files/courses/Fotometrie_a_radiometrie.pdf)
- [20] YASIR, Muhammad, Siu-Wai HO a Badri N. VELLAMBI. Indoor Positioning System Using Visible Light and Accelerometer. *Journal of Lightwave Technology* [online]. 2014, 32(19), 3306-3316 [cit. 2023-05-17]. ISSN 0733-8724. Dostupné z: doi:10.1109/JLT.2014.2344772



- [21] GU, Yanying, Anthony LO a Ignas NIEMEGEREERS. *A survey of indoor positioning systems for wireless personal networks* [online]. IEEE, 2009, 11(1), 13-32 [cit. 2023-05-17]. ISSN 1553-877X. Dostupné z: doi:10.1109/SURV.2009.090103
- [22] HARTLEY, Richard a Andrew ZISSERMAN. *Multiple View Geometry in Computer Vision*. Cambridge: Cambridge University Press, 2004;2011; ISBN 9780521540513;0521540518;
- [23] BIAGI, Mauro, Stefano PERGOLONI a Anna Maria VEGNI. LAST: A Framework to Localize, Access, Schedule, and Transmit in Indoor VLC Systems. *Journal of Lightwave Technology* [online]. IEEE, 2015, 33(9), 1872-1887 [cit. 2023-05-17]. ISSN 0733-8724. Dostupné z: doi:10.1109/JLT.2015.2405674
- [24] BOSÁK, Štěpán. *Mobile-Robot and Platform for VLC Indoor Navigation*. Praha, 2022. Diplomová práce. České vysoké učení technické v Praze.
- [25] SUDA, Martin. *Infrastructure of Indoor VLP System*. Praha, 2022. Diplomová práce. České vysoké učení technické v Praze.
- [26] Vishay Semiconductors [online]. *BPW34, BPW34S*. ©2023 [cit. 2023-05-17]. Dostupné z: <https://www.vishay.com/docs/81521/bpw34.pdf>
- [27] Stabilize Your Transimpedance Amplifier. In: *Analog Devices* [online]. Wilmington, MA: Analog Devices, c1995 - 2023 [cit. 2023-05-17]. Dostupné z: <https://www.analog.com/en/technical-articles/stabilize-transimpedance-amplifier-circuit-design.html>
- [28] Small feedback capacitor in photodiode amplifier. In: *StackExchange* [online]. New York, United States: Stack Exchange, c2023 [cit. 2023-05-18]. Dostupné z: <https://electronics.stackexchange.com/questions/552307/small-feedback-capacitor-in-photodiode-amplifier>
- [29] Texas Instruments [online]. *LMx58-N Low-Power, Dual-Operational Amplifiers*. ©2023 [cit. 2023-05-18]. Dostupné z: [https://www.ti.com/lit/ds/snosbt3j/snosbt3j.pdf?ts=1684348761793&ref\\_url=https%253A%252F%252Fwww.google.com%252F](https://www.ti.com/lit/ds/snosbt3j/snosbt3j.pdf?ts=1684348761793&ref_url=https%253A%252F%252Fwww.google.com%252F)
- [30] ams OSRAM [online]. *BPX 81*. 2018 [cit. 2023-05-18]. Dostupné z: <https://look.ams-osram.com/m/49978f9fd76cd195/original/BPX-81.pdf>
- [31] Inter-IC Sound (I2S). In: *ESP-IDF Programming Guide* [online]. Shanghai: Espressif Systems, c2016 - 2023 [cit. 2023-05-18]. Dostupné z: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/i2s.html>
- [32] ONDŘEJ, Šimon. Algoritmy určování polohy pomocí komunikace ve viditelném světle. In: *GitHub* [online]. San Francisco: GitHub, c2023 [cit. 2023-05-20]. Dostupné z: <https://github.com/ondresim/BP>
- [33] *Pandas* [online]. Austin, TX: NumFOCUS, c2023 [cit. 2023-05-20]. Dostupné z: <https://pandas.pydata.org/>

- [34] Qt for Python. *Qt Group* [online]. The Qt Company, c2023 [cit. 2023-05-20]. Dostupné z: <https://www.qt.io/qt-for-python>
- [35] Types of Machine Learning. In: *JavaTpoint* [online]. India: JavaTpoint, c2011-2021 [cit. 2023-05-20]. Dostupné z: <https://www.javatpoint.com/types-of-machine-learning>
- [36] What is the k-nearest neighbors algorithm?. In: *IBM* [online]. New York [cit. 2023-05-21]. Dostupné z: <https://www.ibm.com/topics/knn>
- [37] *Scikit-learn* [online]. c2007 - 2023 [cit. 2023-05-20]. Dostupné z: <https://scikit-learn.org/stable/>
- [38] *SciPy* [online]. c2023 [cit. 2023-05-20]. Dostupné z: <https://scipy.org/>
- [39] *NumPy* [online]. c2023 [cit. 2023-05-20]. Dostupné z: <https://numpy.org/>
- [40] Šimon Ondřej, 2023, *Visible Light Positioning - KNN*, YouTube video. [cit. 2023-05-20]. Dostupné z: <https://youtu.be/s0o3o1e-UI0>
- [41] Closest point to 3 (or more) circles. In: *StackExchange* [online]. New York: Stack Exchange, c2023 [cit. 2023-05-20]. Dostupné z: <https://math.stackexchange.com/questions/366894/closest-point-to-3-or-more-circles>
- [42] Šimon Ondřej, 2023, *Visible Light Positioning - DFT triangulation*. YouTube video. [cit. 2023-05-20]. Dostupné z: <https://youtu.be/kJZK7cJT50Q>

## A Naměřená data

Skutečná poloha		Vypočtená poloha		Vzdálenost poloh
$x$ [cm]	$y$ [cm]	$x$ [cm]	$y$ [cm]	$D_{\text{Euklidovská}}$ [cm]
-50.000	0.000	-47.460	-2.960	3.900
-25.000	0.000	-19.975	-1.300	5.190
0.000	0.000	6.910	-3.270	7.645
25.000	0.000	30.390	-12.675	13.773
50.000	0.000	48.700	-3.140	3.398
-37.500	-21.650	-32.770	-19.540	5.179
-12.500	-21.650	-3.530	-20.590	9.032
12.500	-21.650	17.220	-29.440	9.108
37.500	-21.650	38.190	-18.850	2.884
-25.000	-43.300	-19.960	-48.260	7.071
0.000	-43.300	5.825	-52.490	10.881
25.000	-43.300	22.200	-50.060	7.317
-12.500	-64.950	-9.190	-67.470	4.160
12.500	-64.950	16.930	-70.780	7.322
0.000	-86.600	-0.420	-85.780	0.921
0.000	-12.000	2.210	-9.760	3.147
0.000	-30.000	0.310	-30.010	0.310
0.000	-61.000	3.180	-68.410	8.064
-25.000	-17.000	-24.220	-16.040	1.237
25.000	-20.000	23.430	-23.880	4.186
25.000	-32.000	28.670	-32.580	3.716
-25.000	-29.000	-24.510	-42.090	13.099
-12.000	0.000	-5.520	-1.320	6.613
12.000	-14.000	14.860	-16.860	4.045
15.000	-54.000	10.450	-56.260	5.080
-21.000	-53.000	-14.130	-61.260	10.744
-13.000	-32.000	-16.510	-30.160	3.963
10.000	-46.000	22.990	-47.090	13.036
0.000	-51.000	5.510	-56.730	7.949
0.000	-23.000	1.010	-28.130	5.228
		Maximum	Průměr	
		13.773 cm	6.273 cm	

**Tabulka A.1:** Data naměřená při testování metody DFT triangulace. Bod (0,0) byl zvolen jako střed spodní strany trojúhelníku měřícího prostředí. Pro body nalevo od něj platí  $x < 0$ . Pro body nahoru od něj platí  $y < 0$ .