



**FAKULTA  
INFORMAČNÍCH  
TECHNOLOGIÍ  
ČVUT V PRAZE**

## ZADÁNÍ DIPLOMOVÉ PRÁCE

**Název:** Lokalizace senzorů v síti LoRaWAN  
**Student:** Bc. Jiří Pospíšilík  
**Vedoucí:** RNDr. Ing. Vladimír Smotlacha, Ph.D.  
**Studijní program:** Informatika  
**Studijní obor:** Počítačové systémy a sítě  
**Katedra:** Katedra počítačových systémů  
**Platnost zadání:** Do konce letního semestru 2018/19

### **Pokyny pro vypracování**

Navrhněte způsob lokalizace a určení polohy senzorů v síti LoRaWAN na principu porovnání doby šíření signálu od několika bran se známou polohou. Implementujte algoritmus, který bude využívat agregovaná data z jednotlivých bran. Specifikujte požadavky na senzor, navrhněte a vyrobte jeho prototyp. Reálné testy proveďte v experimentální síti LoRaWAN společnosti CESNET. Vyhodnoťte dosažitelnou přesnost a porovnejte s jinými metodami.

### **Seznam odborné literatury**

Dodá vedoucí práce.

prof. Ing. Róbert Lórencz, CSc.  
vedoucí katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.  
děkan

V Praze dne 11. ledna 2018





**FAKULTA  
INFORMAČNÍCH  
TECHNOLGIÍ  
ČVUT V PRAZE**

Diplomová práce

## **Lokalizace senzorů v síti LoRaWAN**

*Bc. Jiří Pospíšilík*

Katedra počítačových systémů

Vedoucí práce: RNDr. Ing. Vladimír Smotlacha, Ph.D.

4. května 2018



---

## Poděkování

Tímto bych chtěl poděkovat vedoucímu práce RNDr. Ing. Vladimíru Smotlachovi, Ph.D. za připomínky a rady v průběhu tvorby diplomové práce, kolejnímu klubu Pod-O-Lee za zpřístupnění a umožnění tisku na 3D tiskárně, Davidu Kovandovi za výpomoc při návrhu testovacího senzoru a klubu SiliconHill za dočasné připojení LoRaWAN brány. Dále své rodině a blízkým za podporu během celého trvání studia.



---

## Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Praze dne 4. května 2018

.....

České vysoké učení technické v Praze  
Fakulta informačních technologií

© 2018 Jiří Pospíšilík. Všechna práva vyhrazena.

*Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení na předchozí straně, je nezbytný souhlas autora.*

### **Odkaz na tuto práci**

Pospíšilík, Jiří. *Lokalizace senzorů v síti LoRaWAN*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2018.



---

# Abstrakt

Tato diplomová práce se zabývá lokalizací senzorů v IoT síti LoRaWAN a současně popisuje a připravuje potřebnou infrastrukturu. Řešení si klade za cíl energeticky nenáročný způsob určení pozice, který by nesnižoval životnost instalovaných senzorů. To je důvodem pro využití pasivní lokalizace, jenž využívá pouze informace získané z běžného provozu senzoru - konkrétně rozdílné doby šíření signálu k jednotlivým LoRaWAN branám. Navržené a implementované řešení je porovnáno s ostatními metodami lokalizace. Pro měření je využit senzor vybaven externím zdrojem určení pozice, který slouží jako referenční hodnota polohy. Experimentální ověření funkčnosti proběhlo na infrastruktuře sdružení CESNET v Praze. Prezentované řešení nedosahuje přesnosti technologií jako je GPS, nicméně v některých případech nabízí zajímavou a energeticky přívětivější alternativu.

**Klíčová slova** IoT, LoRaWAN, LoRa, lokalizace senzorů, TDoA, multilaterace, hyperbolické výpočty, pasivní lokalizace

---

# Abstract

This master's thesis deals with sensor localization in IoT based network called LoRaWAN and presents (prepares) necessary infrastructure. Final solution aims to be power efficient in a way that would not degrade sensor lifetime. That is reason for using passive localization which utilizes information gathered from regular sensor operation - specifically using time difference of arrival. Solution, which is proposed and implemented, is compared to other localization techniques. Measurements are made by test sensor equipped with external source of position, i. e. GNSS module, which functions as a reference location. Proper function was verified by field test conducted in Prague, utilizing LoRaWAN gateways run by CESNET association of legal entities. Presented solution does not achieve accuracy of technologies like GPS, however, it can offer an interesting and power friendly alternative, in some use cases.

**Keywords** IoT, LoRaWAN, LoRa, sensor localization, TDoA, multilateration, hyperbolic calculations, passive localization

---

# Obsah

<b>Úvod</b>	<b>1</b>
Cíle práce . . . . .	2
Struktura práce . . . . .	3
<b>1 LoRaWAN protokol</b>	<b>5</b>
1.1 LoRa vs. LoRaWAN . . . . .	5
1.2 Síťová topologie . . . . .	6
1.3 Specifikace . . . . .	8
1.4 Konkurenční technologie . . . . .	13
<b>2 Geodetický model země a geolokace</b>	<b>15</b>
2.1 Základní pojmy . . . . .	15
2.2 Referenční model . . . . .	15
2.3 Souřadnicový systém . . . . .	17
2.4 Projekce do 2D . . . . .	18
<b>3 Současný stav lokalizačních metod</b>	<b>21</b>
3.1 Globální družicový polohový systém . . . . .	21
3.2 LoRa geolokace . . . . .	24
3.3 Přehled dalších technologií . . . . .	28
3.4 Porovnání . . . . .	30
<b>4 Potřebná infrastruktura</b>	<b>33</b>
4.1 LoRaWAN brána . . . . .	33
4.2 Síťový server . . . . .	36
4.3 Testovací senzor . . . . .	38
<b>5 Návrh lokalizačního řešení</b>	<b>43</b>
5.1 Výpočet polohy . . . . .	43
5.2 Úložiště . . . . .	48

5.3	Spolupráce všech komponent . . . . .	50
<b>6</b>	<b>Implementace</b>	<b>51</b>
6.1	SensorLoc aplikace . . . . .	51
6.2	Počáteční konfigurace a logy . . . . .	55
<b>7</b>	<b>Testování a měření</b>	<b>57</b>
7.1	Mapování síly signálu . . . . .	57
7.2	Experimentální měření . . . . .	58
7.3	Zhodnocení . . . . .	62
	<b>Závěr</b>	<b>65</b>
	<b>Literatura</b>	<b>67</b>
<b>A</b>	<b>Seznam použitých zkratk</b>	<b>73</b>
<b>B</b>	<b>Fotky z instalace bran a testování</b>	<b>77</b>
<b>C</b>	<b>Testovací senzor konstrukce</b>	<b>81</b>
<b>D</b>	<b>Obsah příloženého CD</b>	<b>83</b>

---

## Seznam obrázků

1.1	Jednotlivé vrstvy - LoRaWAN stack . . . . .	6
1.2	Mapa pokrytí LoRaWAN sítí v ČR (Q3 2017) . . . . .	7
1.3	Typická LoRaWAN topologie . . . . .	8
1.4	Formát LoRa zprávy . . . . .	10
1.5	Přehled vysílacího výkonu a rychlosti přenosu . . . . .	11
2.1	Nadmořská výška - vztah mezi referenčním elipsoidem a geoidem .	16
2.2	Zeměpisné souřadnice vs. ECEF . . . . .	18
2.3	Mercatorova projekce . . . . .	19
2.4	UTM zóny . . . . .	20
3.1	GPS princip lokalizace - geometrická představa založená na pseudovzdálenostech . . . . .	23
3.2	Hyperbola a konstantní vzdálenost - základní princip TDoA . . . .	26
3.3	Lokalizační technologie - členění . . . . .	30
4.1	Testovací LoRaWAN brána . . . . .	35
4.2	Resin.io základní pohled na spravované zařízení . . . . .	36
4.3	The Things Network síťová architektura . . . . .	37
4.4	Připojení tlačítka interní pullup rezistor . . . . .	39
4.5	Sestavený testovací senzor v boxu z 3D tiskárny . . . . .	42
5.1	LoRa lokalizace a nalezení řešení pomocí pseudovzdálenosti (geometrická interpretace hyperbola) . . . . .	44
5.2	Sekvenční diagram celého procesu lokalizace a zúčastněné subjektů	48
5.3	Diagram nasazení - rozložení komponent a propojení . . . . .	50
6.1	Průběh výpočtu z aplikačního pohledu . . . . .	53
7.1	Sestavení senzoru - zkompletovaná horní část . . . . .	59
7.2	Histogram ze 131 odeslaných zpráv - test č. 1 Strahov . . . . .	60

7.3	Histogram ze 116 odeslaných zpráv - test č. 2 Parukářka . . . . .	61
7.4	Rozptyl spočtených hodnot pozice ku referenční (test č. 1) . . . . .	61
7.5	Rozptyl spočtených hodnot pozice - heat mapa (test č. 1) . . . . .	62
B.1	Instalované LoRaWAN brány . . . . .	77
B.2	Vnější pohled na LoRaWAN bránu (verze 1) . . . . .	78
B.3	Vnější pohled na LoRaWAN bránu (verze 2) . . . . .	78
B.4	Testovací kolo - mapování síly signálu . . . . .	79
C.1	Sestavení senzoru - přehled jednotlivých dílů . . . . .	81
C.2	Sestavení senzoru - spodní část . . . . .	82
C.3	Sestavení senzoru - zkompletovaná horní část . . . . .	82

---

# Seznam tabulek

1.1	LoRaWAN classes přehled . . . . .	9
7.1	Statistika z provedených testů (chyby v metrech) . . . . .	60





---

# Úvod

Digitální revoluce v 21. století pokračuje a navazuje na trendy započaté koncem minulého tisíciletí. Stále více se opírá o celosvětovou síť Internet, jenž se stala její hybnou silou, jakožto prostředník a spojující element, který umožňuje a v mnoha odvětvích iniciuje technologický vývoj a prostup technologií do běžného života; napříč všemi vrstvami lidské činnosti. Vše propojující síť stírá geografické rozdíly (ruku v ruce s vývojem v dopravě) a umožňuje průnik myšlenkových pochodů, stavicích na akcelerované globalizaci.

Vývoj a nabyté znalosti z oblasti výroby zajistily miniaturizaci integrovaných obvodů, senzorů a v konečném důsledku celých zařízení a spolu s masivním rozšířením Internetu (a jeho cenovou dostupností) následovalo rozšíření a v jistém smyslu oživení všemožných zařízení, která se do této sítě připojují a tím nabízí nové možnosti využití a interakce. S tímto trendem se objevil pojem *internet věcí* (IoT) jenž tyto chytré/propojené věci v jistém smyslu zaštiťuje a který je v poslední době stále častěji nahrazován generickým pojmem *internet všeho* neboli IoE, který se snaží pojmout rychle se rozvíjející ekosystém Internetu a všudypřítomného propojení - ať už mezi lidmi a jejich tělesnými senzory, zvířaty nebo věcmi a komplexními systémy. Částečně v sobě skrývá fakt, že si veškeré technologie a služby, které nás obklopují a které využíváme, ani neuvědomujeme; stávají se transparentními. Stávají se přirozenou a nedílnou součástí běžného života.

Informace se staly velmi cennou komoditou a nabízejí přidanou hodnotu. To znamená, že jsou data, resp. informace - od senzorů, zařízení, tedy obecně prvků IoT - sbírány a hromaděny, i když nejsou bezprostředně potřeba (jednoduše jsou hromaděna veškerá dostupná data, jelikož mohou být jistým způsobem hodnotné). Tj. nejedná se o informaci, kterou vyžaduje uživatel, případně aplikace/služba ke své primární funkci. Takovým příkladem je poloha, která je často vyžadována (navigace, sledování zásilky atp.), nicméně neméně často je sbírána čistě ze statistických či evidenčních důvodů (případně pro budoucí využití).

Poloha, získána procesem lokalizace, se stala kritickou informací (spolu

s časem, tj. „kdy“ a „kde“). Potřeba lokalizovat věci (IoT zařízení) potažmo osoby, které je nosí nebo jsou v jejich blízkosti, vznikla na základě aplikačních požadavků a požadavků společností, resp. zákazníků. Jednoduše se jedná o velice hodnotnou informaci - víme kde se daná osoba pohybuje, kde se nachází daná věc ... To má velice širokou škálu využití; od marketingových účelů a prodeje zboží po optimalizaci v dopravě, či informaci o procentuální době, kterou domácí mazlíček stráví za stěnami domu.

Pro masové rozšíření IoT senzorů je nutný autonomní (na baterii, bez potřeby externího připojení/napájení, obsluhy) provoz bez nutnosti příliš časté výměny (řádově jednotky let). To jde ovšem proti vlastnostem standardních metod pro lokalizaci jako jsou GNSS/GPS, GSM, Bluetooth nebo WiFi. V těchto případech se jedná o aktivní způsob lokalizace, kdy využíváme externí zdroj určení pozice, avšak za cenu vysoké energetické náročnosti. Existuje spousta aplikací, které nevyžadují vysokou přesnost GNSS modulů, na druhou stranu ovšem vyžadují nízkou spotřebu. Z aplikačního pohledu je přesnost relativní. Pokud chceme ve skladu umístit položku do boxu o velikosti 1 m, znamená pro nás „přesný“ řádově 10 cm. Ovšem pokud chceme vědět zdali je nějaká osoba (např. potomek) ve městě, znamená „přesný“ řádově 1 km. Z toho plyne, že pro uspokojení dnešních velmi rozmanitých požadavků, je nutné volit vhodnou kombinaci dostupných metod pro lokalizaci - v konečném důsledku jakési hybridní řešení.

Tato práce se zabývá lokalizací IoT senzorů, za využití energeticky nenáročných metod. Tedy bez externího zdroje určení pozice (typicky GNSS), navzdory pravděpodobné ztrátě přesnosti. To mj. znamená, že lze tento postup aplikovat na všechny taková zařízení. Téma práce je motivováno absencí lokalizačních nástrojů, které by byly otevřené a zdarma přístupné. Experimentálních testů, ve větších městech (i obecně), je zatím také pramálo. Konkrétně bude využita LoRaWAN síť, která je - v Praze - experimentálně provozovaná sdružením CESNET.

## Cíle práce

Hlavním cílem práce je implementovat řešení, které bude schopno určit/aproximovat absolutní pozici senzoru, který komunikuje pomocí LoRa modulace, resp. LoRaWAN protokolu. Dílčími úkoly jsou:

- Návrh řešení uplatňující pasivní lokalizaci - na základě rozdílné doby šíření signálu k jednotlivým LoRaWAN branám (GW - gateway), neboli TDoA (Time Difference of Arrival). Pojem „pasivní“ je používán ve smyslu neúčasti samotného senzoru na procesu lokalizace, tj. získání pozice není aktivně vyžadováno, a ani nemůže být - senzor si celého procesu není vědom. Jedná se o čistě vedlejší produkt nějaké hlavní funkce (typicky odesílání sensorických dat).

- Implementace algoritmu, který bude využívat metadata (primárně časovou známku) z jednotlivých bran (které vyslaný paket přijaly) a případně provede jejich filtraci, pro určení absolutní pozice (zeměpisná šířka a délka). Spočtenou pozici uložit do DB spolu s původním datovým paketem na aplikačním serveru (bude využit virtuální server poskytnutý sdružením CESNET).
- Popsání a připravení potřebné infrastruktury (brány, síťový server, aplikační server), na které bude možné toto řešení vystavět.
- Vytvoření senzoru, který bude sloužit k testování a vývoji. Bude odesílat referenční hodnotu polohy (za pomoci GNSS/GPS modulu), jenž umožní následné porovnání s vypočtenou pozicí.
- Terénní testy - experimentální ověření a vyhodnocení z reálného prostředí. Případné doporučení na zlepšení přesnosti.

## Struktura práce

První část je teoretická a začíná kapitolou o LoRaWAN protokolu, typické topologii, používaných pojmech a konkurenčních technologiích. Druhá kapitola představuje modely používané při geometrickém popisu země, souřadnicové systémy či konverze mezi jednotlivými modely. Třetí kapitola popisuje současné lokalizační metody a jejich principy v porovnání s LoRa přístupem.

Druhá část je praktická a začíná přípravou potřebné infrastruktury, tj. nastavením LoRaWAN serveru, konfigurací bran a navržením testovacího senzoru. Následující kapitoly prezentují návrh a implementaci samotného řešení (použité přístupy a algoritmy, technologie, nástroje). Poslední kapitola obsahuje testování a ověření funkčnosti navrženého řešení.



---

# LoRaWAN protokol

LoRaWAN je jedním z komunikačních IoT protokolů a technologií, které jsou v poslední době na vzestupu a z tzv. stavu „proof of concept“ a vytváření nových UC (Use Case) přechází do fáze produkční. Technologie jako taková dozrává a pomalu si hledá místo na trhu i v reálných projektech. [1]

Na fyzické vrstvě využívá bezdrátovou RF technologii LoRa<sup>1</sup>, kterou vyvinul Semtech právě pro podporu IoT a M2M (Machine-to-machine) aplikací (jedním z prvních využití bylo odečítání dat z vodoměru). [2] V následujících sekcích jsou tyto pojmy rozvedeny podrobněji, je uvedena specifikace a podoba typické topologie. V závěru jsou uvedeny podobné/konkurenční technologie.

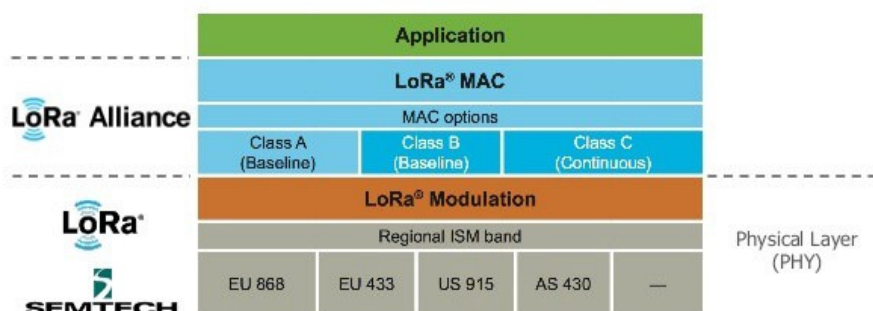
## 1.1 LoRa vs. LoRaWAN

LoRa je v úzkem slova smyslu proprietární Chirp modulační schéma, které je odvozeno od CSS (Chirp Spread Spectrum) modulace - jedná se o fyzickou vrstvu určenou pro rádiovou komunikaci (díky tomu je vhodnou technologií pro lokalizaci viz 3.2). Zároveň umožňuje změnu přenosové rychlosti na úkor citlivosti v pevně daném pásmu. Vysílací výkon - a tím daný dosah a spotřeba - je proměnný/dynamický a určuje přenosovou rychlost. Z toho plyne, že je možné optimalizovat výkon podle toho jak je vysílač daleko od přijímače, využívat odlišnou šířku pásma, dobu přenosu (neboli air-time), kterou jsou data vysílána a v konečném důsledku energetickou durabilitu (jenž je pro tyto zařízení kritická). Jednou z vlastností LoRa je schopnost lineárně odlišit frekvenci a časové chyby, což z ní činí ideálního adepta pro rádiové aplikace (potažmo lokalizační služby).

Někdy je pod pojmem LoRa označovaná technologie jako celek, často zahrnující i vyšší programovou nadstavbu (protocol stack) v podobě LoRaWAN

---

<sup>1</sup>LoRaWAN navíc podporuje standardní FSK modulaci, nicméně tato kombinace není běžně užívána.



Obrázek 1.1: Jednotlivé vrstvy - LoRaWAN stack [2]

protokolu. Samotná LoRa je ovšem agnostická vůči implementacím na vyšších vrstvách.

LoRaWAN je otevřenou specifikací, vyvíjenou neziskovou organizací *LoRa Alliance*<sup>™</sup> [3], která tímto krokem sledovala zvýšení interoperability a masovější rozšíření v rámci států a dalších kontinentů. Zároveň poskytuje certifikaci, která zaručuje, že dané zařízení vyhovuje právním regulacím daného státu a specifikaci protokolu. Mezi členy aliance patří mj. Semtech, Actility, IBM, Microchip nebo Cisco. Typově spadá do kategorie LPWAN (Low-Power, Wide-Area Network) sítí, které se vyznačují nízkou spotřebou a širokým rozsahem pokrytí. S tím také souvisí nižší datový tok - typicky přenášenými informacemi jsou data senzorská o velikosti v jednotkách bajtů (obecně binárně - co nejušporněji - zakódované číslo/hodnota). Využívány jsou volná pásma ISM (Industrial, Scientific and Medical), které podléhají jistým omezením viz sekce 1.3.2.

Kooperace mezi LoRa a LoRaWAN (celý stack) je možné vidět na obrázku 1.1. Při návrhu byl kladen důraz na vysokou míru podobnosti s IEEE 802.15.4, tak aby byly zachovány vyšší vrstvy (ve smyslu kompatibility).

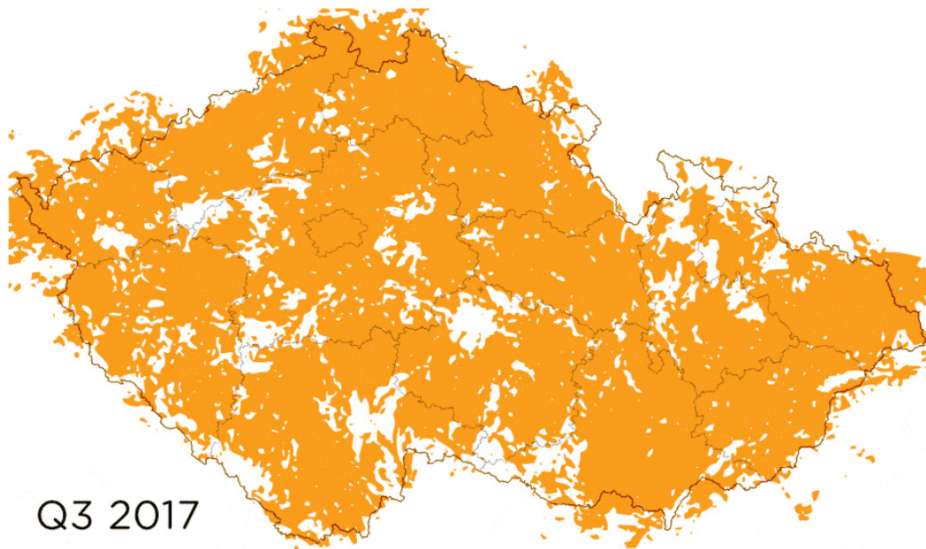
V České Republice provozuje LoRaWAN síť společnost České Radiokomunikace, mapu pokrytí lze vidět na obrázku 1.2.

Dalším příkladem protokolu na vyšší vrstvě, který využívá LoRa modulaci je *Symphony Link* od společnosti Link Labs, který implementuje některé pokročilé prvky, jenž nachází uplatnění v průmyslu a velkých instalacích (např. aktualizace firmwaru „Over-the-Air“) [5].

## 1.2 Síťová topologie

V této sekci jsou popsány základní stavební prvky LoRaWAN sítě, jenž typicky používá hvězdicovou (z globálního pohledu hvězda hvězd) topologii. Těmi jsou:

- *Brána* (gateway) - neboli koncentrátor či základní stanice. V hvězdicové topologii představuje ústřední prvek, který přijímá datové pakety

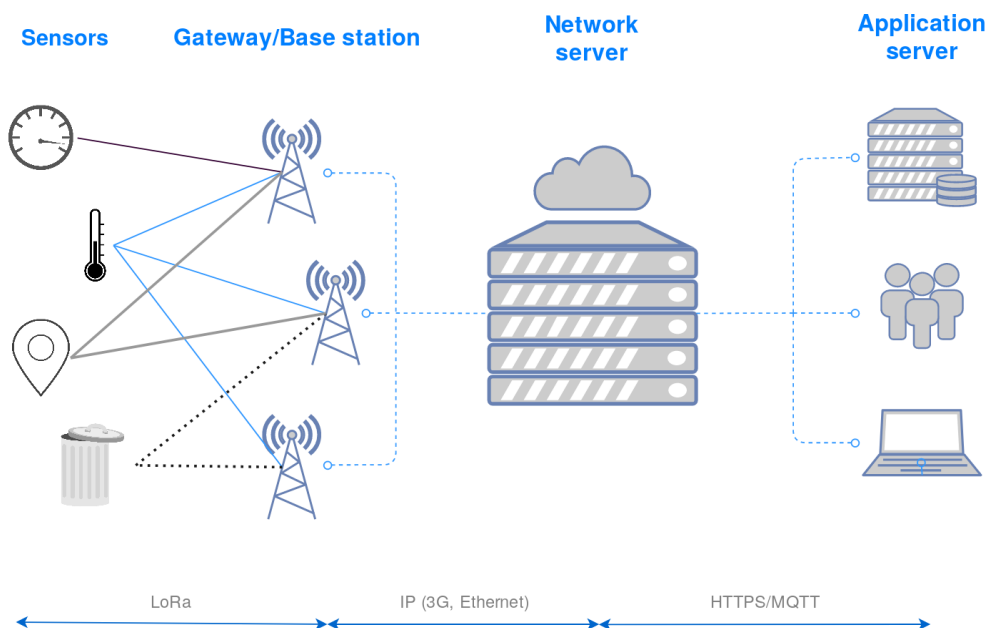


Obrázek 1.2: Mapa pokrytí LoRaWAN sítí v ČR (Q3 2017) [4]

od všech zařízení v okolí - podobně jako např. BTS (Base Transceiver Station) v mobilních sítích GSM (Global System for Mobile Communications). Tyto data pak přeposílá na centrální síťový server (backend), zpravidla po IP (Internet Protocol) síti - Ethernet, WiFi, 3G/LTE. Představuje poměrně jednoduchou roli prostředníka, brána samotná o ničem nerozhoduje ani žádné úkony neinicuje; pouze k datové struktuře, kterou odesílá síťovému serveru přidá určitá metadata (např. čas přijetí paketu nebo použité pásmo).

- *Koncová zařízení* (end-devices) - nebo (LoRa) uzly. Jedná se o samotné senzory a prvky IoT. Vyslaný paket může být přijat více branami najednou, tj. z pohledu senzoru se jedná o komunikaci typu „one-to-many“.
- *Síťový server* (network server) - centrální prvek, který hraje důležitou roli. Komunikuje s jednotlivými branami (hvězdicová topologie) a agreguje data (pokud jsou přijaty od více bran) od nich získaná. Provádí úkony jako jsou detekce redundantních paketů, bezpečnostní kontrola a plánování tzv. *downlink*<sup>2</sup> paketů. Musí splňovat LoRaWAN specifikaci a má řídicí úlohu (např. může měnit vysílací výkon na koncovém zařízení). Agregovaná data přeposílá na aplikační server, případně aplikuje požadované integrační a transformační funkce.

<sup>2</sup>Data odesílány aplikačním serverem směrem ke koncovému zařízení. Jedná se o méně častý typ komunikace a často nežádáný (vyšší energetická spotřeba, podrobněji sekce 1.3).



Obrázek 1.3: Typická LoRaWAN topologie

- *Aplikační server* (application server) - operuje nad samotnými daty. Často už spravován zákazníkem, který si přijatá data uloží do DB a dále využívá k naplňování obchodních cílů (statistiky, grafy).

Názorný příklad LoRaWAN sítě lze vidět na obrázku 1.3.

### 1.3 Specifikace

*Tato sekce čerpá především z LoRaWAN specifikace verze 1.0.2 vydané LoRa Alliance [6].*

Verze 1.0.2 není poslední vydanou verzí - tou je nově verze 1.1 (zpětně kompatibilní), nicméně jedná se o verzi, která je běžně podporovaná a provozovaná na síťových serverech (na rozdíl od nejnovější verze) - mj. i na serveru používaném pro účely této práce (viz kapitola 4). Oproti předchozí verzi se mírně změnilo názvosloví a některé požadavky (přidány relační klíče, nové parametry, a jiné naopak odebrány; u některých termínů došlo k upřesnění).

Zařízení implementující LoRaWAN stack mohou pracovat ve třech režimech/úrovních - tzv. LoRaWAN classes. Základním režimem, který musí podporovat každé zařízení je class A. Následují class B a C, které nabízejí další funkce a cílí na odlišný scénář užití. V následujícím textu bude popsána primárně základní class A (zmíněná pravidla platí i pro ostatní operační režimy). Krátké srovnání v tabulce 1.1.



Tabulka 1.1: LoRaWAN classes přehled

Class A	Obousměrná komunikace, downlink je plánován na základě dvou časových slotů (okének), která jsou alokována (zařízení naslouchá a je připraveno přijmout data) po odeslání uplink paketu. Má nejnižší spotřebu, tj. typicky malé senzory s integrovanou baterií.
Class B	Časových slotů pro příjem - downlink - je plánováno více. To probíhá na základě synchronních Beacon zpráv od brány (server ví, kdy zařízení naslouchá a může odeslat např. řídicí příkaz). Nižší latence downlink zpráv.
Class C	Přijímá neustále - vyjma samotného odesílání. Nejnižší latence downlink zpráv za cenu vyšší spotřeby.

### 1.3.1 Formát rámce

Formát fyzické zprávy (kompletní struktura přenášená rádiovým modulem) je odlišný pro uplink<sup>3</sup> a downlink zprávu - ta se liší absencí kontrolního součtu CRC (cyclic redundancy check) pro vlastní data přenášená na vyšších vrstvách neboli *PHYPayload*. Samotná data jsou přenášena ve struktuře *FRMPayload* viz obr. 1.4. Kromě dat tam mohou být přenášeny také příkazy na MAC vrstvě (např. změna frekvence, vysílacího výkonu, okének pro příjem); taková situace je indikována nastavením *FPort* hodnoty na 0, v opačném případě tato hodnota slouží pro odlišení různých aplikací, případně testovacích scénářů.

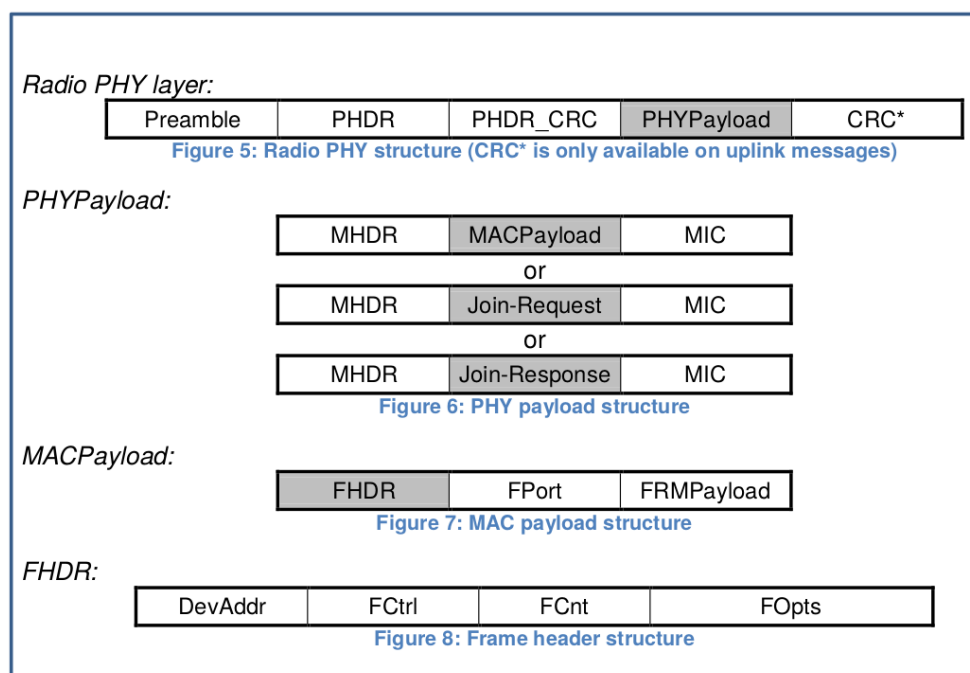
### 1.3.2 Kmitočtové pásmo a ADR

LoRaWAN k přenosu využívá ISM pásmo, které je v regionech jako EU, USA nebo Čína odlišné. V EU a potažmo ČR jsou využívány 868MHz frekvence, které jsou regulovány evropským úřadem *ETSI* (European Telecommunications Standards Institute), resp. *ČTU* (Český telekomunikační úřad) v ČR. Např. je nutné dodržovat vysílací výkon: typicky do 25 mW ERP (Effective radiated power) a klíčovací poměr<sup>4</sup>, jenž udává procentuální poměr po který je možné okupovat dané pásmo/podpásmo: typicky méně než 1 nebo 0.1%. [7] Specifikace nevyužívá alternativní přístup k médiu LBT (Listen Before Talk).

DR (Data rate) - koresponduje s tzv. SF (Spreading factor) - určuje použitou rychlost přenosu, a v návaznosti na to také vysílací výkon viz obrázek 1.5. DR tedy ovlivňuje kolik bajtů/oktetů aplikačních dat je možné přenést v jedné zprávě, maximálně to je 51 - 222 oktetů (DR0 - DR7). V EU863-

<sup>3</sup>Data odesílány koncovým zařízením směrem k aplikačnímu serveru. Jedná se o standardní a často jediný typ komunikace. Downlink je pravým opakem.

<sup>4</sup>Klíčovací poměr (duty-cycle) označuje procentuální podíl vysílací doby ku celkové době v daném pásmu. Je regulována a typické hodnoty mohou být 1% nebo 0.1% (ČR), liší se pro jednotlivé frekvence (jistě omezení ISM pásma, aby nebylo zneužíváno a zahlceno).



Obrázek 1.4: Formát LoRa zprávy [6]

870 MHz pásma je nutná podpora alespoň těchto frekvenčních kanálů (šířka pásma 125 kHz) 868.1, 868.3 a 868.5 (DR0-DR5) a klíčovací poměr  $< 1\%$ . Další běžně používané frekvence 867.1 - 867.9 (rozteč 200 kHz) a klíčovací poměr  $< 0.1\%$ .

*ADR* (Adaptive Data Rate) představuje důležitý mechanismus, který má za následek snížení vysílacího času (tedy i zarušení pásma) a energetické náročnosti (delší životnost senzoru). Toho je dosaženo dynamickou změnou DR na základě historicky přijatých zpráv. Algoritmus běžící na síťovém serveru vyhodnotí řadu ukazatelů, jako je síla signálu nebo vysílací čas, a na základě těchto rozhodne zdali je možné navýšit nebo naopak nutné snížit DR (senzor/zařízení je blízko nějaké brány, nebo se naopak oddaluje a je příhodné DR snížit a zvýšit vysílací výkon).

### 1.3.3 Aktivace a přihlášení do sítě

Každé koncové zařízení musí - před samotným odesláním dat - projít procesem přihlášení do místní LoRaWAN sítě. Probíhá ve dvou krocích *personifikace* a *aktivace*. Specifikace definuje dvě různé metody, které je možné použít - **OTAA** (Over the Air Activation) a **ABP** (Activation by Personalition). Odlišují se ve způsobu nastavení/získání identifikačních údajů a bezpečnostních klíčů. To ovlivňuje flexibilitu (snadnost) nasazení, míru zabezpečení, ale také komunikační nutnost směrem k bráně (což může být určující při špatné síle

DataRate	Configuration	Indicative physical bit rate [bit/s]
0	LoRa: SF12 / 125 kHz	250
1	LoRa: SF11 / 125 kHz	440
2	LoRa: SF10 / 125 kHz	980
3	LoRa: SF9 / 125 kHz	1760
4	LoRa: SF8 / 125 kHz	3125
5	LoRa: SF7 / 125 kHz	5470
6	LoRa: SF7 / 250 kHz	11000
7	FSK: 50 kbps	50000
8..15	RFU	

TXPower	Configuration (ERP)
0	20 dBm
1	14 dBm
2	11 dBm
3	8 dBm
4	5 dBm
5	2 dBm
6..15	RFU

Obrázek 1.5: Přehled vysílacího výkonu a rychlosti přenosu [6]

signálu, resp. asymetrickém zisku přijímače a vysílače). Podrobněji v následujícím textu.

Nastavované a v zařízení ukládané hodnoty jsou následující.

- *DevAddr* - adresa koncového zařízení. Jednoznačně identifikuje senzor v síti. Využívá 32 bitů a z toho je 7 bitů využito jako identifikátor sítě (daného operátora) a zbylých 25 bitů představuje síťovou adresu, která může být svévolně přidělena operátorem (podobně jako u IP).
- *AppEUI* - globální identifikátor aplikace (IEEE EUI64), jednoznačně identifikující entitu schopnou vyřizovat požadavky o přihlášení (typicky síťový server operátora, který nám službu poskytuje).
- *NwkSKey* - síťový relační klíč (specifický pro dané zařízení). Slouží pro zajištění integrity a důvěrnosti (pouze pro užitná data tzv. MAC-only zpráv) dat mezi zařízení a síťovým serverem.

- *AppSKey* - aplikační relační klíč (specifický pro dané zařízení). Podobně jako síťový klíč, šifruje ovšem aplikační data mezi zařízením a aplikačním serverem. Síťový server tedy není schopný data odposlouchávat, ale může je pozměnit (neslouží pro ověření integrity - předpoklad, že síťový server je důvěryhodný).
- *DevEUI* - globální identifikátor zařízení (IEEE EUI64). Unikátní - slouží pro jednoznačnou identifikaci.
- *AppKey* - kořenový AES-128 klíč<sup>5</sup> koncového zařízení. Slouží k odvození *NwkSKey* a *AppsKey*, a tedy zajištění integrity a důvěrnosti mezi síťovým, resp. aplikačním serverem.

### 1.3.3.1 OTAA

Tato metoda vyžaduje aktivaci na základě přihlašovací procedury, která - pokud je úspěšná - končí vygenerováním relačních informací (z toho plyne nutnost opakovaného přihlášení, např. při resetu - ztrátě relace). Těmi jsou zejména relační klíče *NwkSKey* a *AppSKey*. Tento proces probíhá mezi síťovým serverem a zařízením, které musí být nejprve personifikováno a to nastavením *DevEUI*, *AppEUI* a *AppKey*.

Samotná aktivaci proběhne ve dvou krocích:

1. *Join-request* zpráva - zařízení odešle žádost, která zahrnuje *AppEUI* a *DevEUI* společně s *DevNonce* (standardní mechanismus proti tzv. replay útokům).
2. *Join-accept* zpráva - síťový server ověří, zdali je dané zařízení (personifikované informace) validní a odpoví zprávou, která obsahuje dodatečné informace pro nastavení zařízení. Konkrétně *DevAddr* a *AppNonce*, jenž slouží, spolu s několika dalšími údaji, pro odvození relačních klíčů na straně zařízení. Tyto informace musí být šifrované pomocí *AppKey* (používá se AES-128 dešifrovací operace v ECB módu, tak aby na zařízení dostačovala implementace šifrovací operace AES-128).

### 1.3.3.2 ABP

Specifikací méně doporučovaná metoda, která je ovšem hojně používaná (viz další odstavce). Jak název napovídá, jsou kroky personifikace a aktivace slouženy v jeden, to znamená, že nastavením, resp. personifikací zařízení, zároveň provedeme aktivaci. Neprobíhá žádná komunikace se serverem a rovnou jsou odeslány aplikační data. Z toho plyne, že je nutné ručně nastavit *DevAddr*, *NwkSKey* a *AppSKey*.

---

<sup>5</sup>Po vyzrazení klíče je možné kompromitovat pouze postižené zařízení (klíče na ostatních zařízeních jsou na sobě nezávislé).

Význam relačních klíčů je tímto statickým nastavením de facto redukován (neplní funkci relačně specifických a tedy proměnných klíčů). Pokud jsou vy-zrazeny, je veškerá budoucí komunikace kompromitována, a navíc může vést i ke kompromitaci zařízení ostatních (při nalezení vzorce pro generování těchto klíčů - hlavně u masového úniku klíčů).

Další nevýhodou je „napevno“ nastavená DevAddr, která odpovídá adrese v rámci sítě specifické danému operátorovi. Pokud výrobce zařízení nebo SW síťového serveru neposkytuje možnost změny této adresy, je problém takové zařízení v jiné síti, než pro kterou bylo určeno používat (typicky operátor země v které bylo zařízení vyrobeno).

Ačkoliv je ABP technicky méně bezpečná metoda, přesto je v praxi použí-vaná. Důvodem je zpravidla horší přijímací schopnost zařízení, než brány. To v praxi znamená, že odeslání uplink paketu je úspěšné, avšak downlink zprávy transportované z brány k zařízení nikoliv (pravděpodobněji je, že projde ale-spoň jeden paket, než že projdou oba). Tento fakt vytváří asymetrickou situaci a znemožňuje OTAA aktivaci, která tuto oboustrannou výměnu vyžaduje. [8]

## 1.4 Konkurenční technologie

V této kapitole je zmíněno několik dalších technologií, které přímo konku-rují LoRa/LoRaWAN protokolu - mají podobné vlastnosti, resp. případy užití (UC) a zároveň jsou nasazovány v celonárodním měřítku (mají ambice globál-ního pokrytí). Např. v ČR vyvinutá technologie IQRF má trochu jiný model nasazení, i když je ve spoustě aspektů velice podobná [9].

### 1.4.1 Sigfox

Sigfox je na rozdíl od LoRaWAN uzavřenou technologií a proprietárním ře-šením. Vlastní síťové servery a veškeré řídicí systémy. V některých státech (Francie, USA) zároveň vykonává roli operátora a pokrytí v dané zemi za-jišťuje sám. V ostatních případech uzavře smlouvu s operátorem (pro danou zemi získá výhradní licenci), který instalaci infrastruktury zajistí. V České Republice se jedná o společnost SimpleCell. Za využívání Sigfox sítě uživatelé platí poplatek (vždy za každý modul). Výrobu koncových zařízení (referenční design) Sigfox - na základě jistých obchodních podmínek - poskytuje. Např. Atmel nebo Texas Instruments produkují Sigfox rádiové moduly. [10]

Na fyzické vrstvě využívá UNB (Ultra Narrow Band) technologii v kombi-naci s modulací DBPSK (Differential BPSK) pro uplink a GFSK pro downlink komunikaci. Užší spektrum (100 Hz) má za následek menší rušení a vyšší odol-nost proti šumu. Není tak flexibilní jako LoRaWAN - v případě adaptabilní přenosové rychlosti či frekvencí, ale na druhou stranu může poskytnout lepší dosah a nižší spotřebu - maximální množství přenášených dat je velmi striktní (12 bytů pro uplink a 8 bytů pro downlink). [11]

Sigfox komunikace je asymetrická a provoz směrem k sensorům je omezený (ještě více než u LoRaWAN, kde funguje plánování downlinku a různé třídy koncového zařízení viz sekce 1.3). To je potřeba zohlednit při analýze požadavků a nasazení.

Jednou z velkých výhod je vyřešený roaming<sup>6</sup> (vše je provozováno na systémech Sigfox). To je dáno samotnou technologií a obchodním modelem nasazení. Na rozdíl od LoRaWAN sítí, které provozují různé soukromé společnosti a možnost migrace zařízení od jednoho k druhému není garantována.

### 1.4.2 NB - IoT

Narrowband IoT neboli LTE Cat NB1 je LPWAN technologie specificky upravená pro přenos dat (generovaných IoT zařízeními) za pomoci (frekvenčních pásem) mobilních sítí, spravovaných operátory (možné zakomponování do existující LTE infrastruktury). Prvotně standardizován 3GPP (Third Generation Partnership Project).

Na rozdíl od LoRa sítě využívá o něco širší pásmo (180 KHz) a svými vlastnostmi zahrnuje mírně odlišnou cílovou skupinu. Je synchronní (z toho plyne vyšší spotřeba) a operuje v licencovaném pásmu (lze aplikovat QoS atp.). Tedy je vhodnější pro aplikace, které vyžadují nižší latenci a častější typ komunikaci. Další výhodou je rychlost přenosu, který je v případě NB-IoT průměrně 200 Kbps. Což je více než 20krát více v porovnání s LoRa (tam je rychlost typicky ještě nižší v závislosti na použitém DR). Jde o vhodnou technologii pro aplikace s požadavkem na vyšší propustnost.

Není možné aplikovat privátní model nasazení jako u LoRa sítí, vždy se jedná o veřejnou službu poskytovanou místním operátorem vlastním infrastrukturou a frekvenční pásma. Pro lokalizační účely využívá techniky popsaných v 3.3.1.

Jedná se o novější technologii než je LoRa a teprve se rozšiřuje (prvotně nasazena ve Španělsku, a poté v několika dalších státech Evropy). V ČR je síť provozována společností Vodafone, která deklaruje pokrytí 100% území. Celostátní nasazení chystá T-Mobile v USA, a to s příznivější cenou než dosud dostupné roční předplatné na LTE [12]. Moduly jsou vyráběny mj. společností u-blox (byla první), která na tuto technologii - spolu s LTE CATM1 - vsadila a doplňuje tím své řady GNSS modulů.

---

<sup>6</sup>Schopnost zařízení přecházet mezi různými provozovateli - napříč různými státy.

---

# Geodetický model země a geolokace

V této kapitole jsou uvedeny základní principy lokalizace na zemi, popisné systémy a způsoby vykreslení do 2D prostoru (typicky mapy) neboli 2D projekce. Pro účely orientace na zemském povrchu je zapotřebí nějaký matematický model, který zemi co nejpřesněji popisuje, a také souřadnicový systém, jenž umožní navigaci a určení konkrétních míst.

## 2.1 Základní pojmy

Lokalizace je proces při kterém dochází k určení pozice; přidaná hodnota takové informace může být nulová, není-li znám kontext, který je určující pro většinu (nejen) senzorických dat. Pozice může být absolutní, nebo relativní. Absolutní pozicí jsou např. souřadnice získané z GPS modulu a relativní např. informace o vzdálenosti od nějakého bodu (závislé na kontextu).

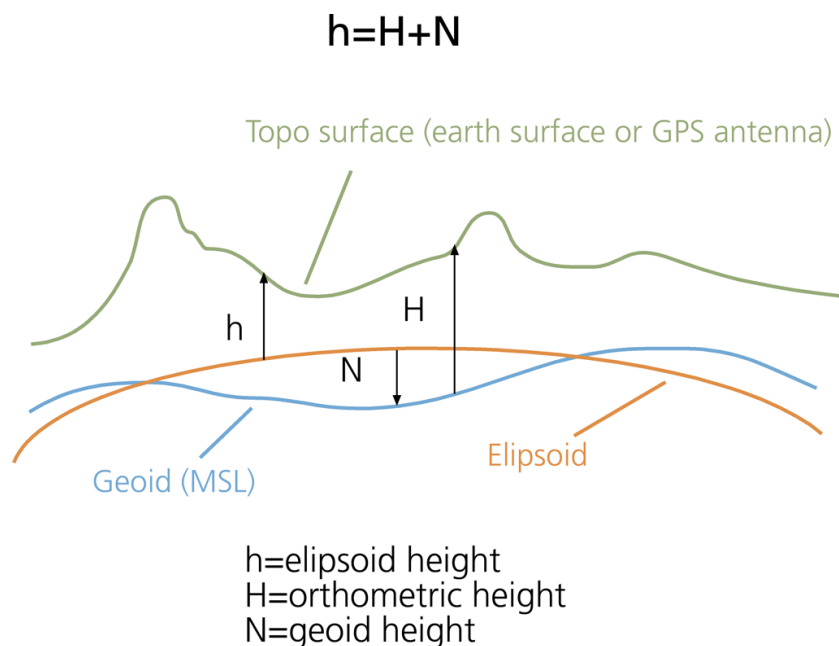
Geolokace jednoznačně reprezentuje geografickou pozici na zemském povrchu; typickým představitelem jsou např. GPS souřadnice (viz následující sekce).

## 2.2 Referenční model

Země - ačkoliv připomíná kouli - je nepravidelná a působením rotace zploštělá (v oblasti rovníku je větší poloměr než v oblastech zemských pólů). Masy hornin a nerovnoměrné rozložení zemské masy má za následek odlišný gravitační potenciál, který ovlivňuje rozložení oceánů na zemském povrchu. Přitom je střední hladina vody - definována jako MSL<sup>7</sup> (Mean Sea Level) - používána

---

<sup>7</sup>Výpočet probíhá dlouhodobým (např. časový horizont 19 let) měřením mořské výšky a provedením váženého průměru; to eliminuje výkyvy způsobené gravitačním působením měsíce a slunce.



Obrázek 2.1: Nadmořská výška - vztah mezi referenčním elipsoidem a geoidem [14]

k modelování zemského povrchu. Až nedávno (s příchodem přesných geodetických satelitů) bylo zjištěno, že jsou tyto výkyvy větší než se očekávalo a vodní povrch netvoří gravitací vytvořený rovnoměrný obal země. Výskyt mírných kopců a údolí je tedy běžný, nicméně o mnoho mírnější. [13, 14]

Geoid je model povrchu země, který odpovídá MSL (přesněji je aproximován MSL), pokud by oceány, resp. voda procházela pod kontinenty - to je provedeno na základě matematických výpočtů (např. EGM2008). Pro výpočty a navigaci je vhodnější geometrický referenční model - zploštělí rotační elipsoid neboli zploštělí sféroid.

Nejčastěji používaným modelem je WGS 84 (Google mapy). WGS 84 je zadán osou hlavní  $a = 6378137$  m, vedlejší  $b = 6356752.314245$  m a inverzním zploštěním  $1/f = 298.257223563$ . Tento model - ačkoliv není nejpřesnější - je jeden z nejpoužívanějších díky využití v GPS. Obsahuje definici souřadnicového systému (viz další sekce) a mezi jeho parametry patří i gravitační model země EGM-96 (Earth Gravity Model), jenž je používán pro korekci výšky - dříve GPS moduly vracely pouze výšku nad referenčním elipsoidem (tj. počítanou od referenčního elipsoidu WGS 84); ta je ovšem v některých místech výrazně odlišná od reality, resp. geoidu aproximovaném pomocí MSL. Největší rozdíl byl naměřen v jihovýchodní Indii, kde je výška nad elipsoidem o 105 výše než výška nad MSL. Korekci nadmořské výšky je možné provést pomocí tzv. geoid výšky, která je definovaná jako rozdíl mezi výškou nad elipsoidem a nadmořskou, resp. ortometrickou:  $N = h - H$ , viz obrázek 2.1.



## 2.3 Souřadnicový systém

V této kapitole jsou shrnuty dva základní typy souřadnicových systémů, které jsou dále využívány v (nejen) praktické části práce. Každý typ má jiné využití dle svých vlastností.

### 2.3.1 Kartézská soustava

Každý bod je identifikován po sobě jdoucími souřadnicemi, které určují vzdálenost od počátku souřadnicového systému v dané dimenzi (směru) - reprezentované vzájemně kolmými a orientovanými přímkami (osy). V případě 2D a 3D prostoru se jedná o uspořádanou dvojici, resp. trojici. Je tedy nutné určit počátek směr v jednotlivých dimenzích a jednotku v které jsou jednotlivé souřadnice uváděny.

Při výpočtech jsou používány souřadnice *ECEF* (Earth Centered, Earth Fixed), jelikož umožňují přesně zvolit bod na zemském povrchu, bez závislosti na použitém modelu země (např. zmíněný WGS 84). Definovány jsou následovně: počátek soustavy je umístěn v centru země<sup>8</sup>; rovina x-y je určena rovníkem; osa *x* je vedena ve směru nultého poledníku<sup>9</sup>; osa *y* je umístěna ortogonálně na osu *x* ve východním směru (tak, aby byl systém orientován doprava); poslední osa *z* je vedena skrze severní pól viz obr. 2.2.

### 2.3.2 Sférická soustava

Jedná se o systém úhlový, který se častěji využívá pro prezentaci - jako snadno představitelný údaj o poloze. Obsahuje informaci o úhlové vzdálenosti (velikost svíraného úhlu) od dvou pevně definovaných os a vzdálenost od počátku.

Z pohledu práce jsou předmětné hlavně zeměpisné (geodetické) souřadnice, konkrétně zeměpisná šířka a délka (vzdálenost od počátku se neuvádí, počítá se s poloměrem země - v případě elipsoidu je náležitě upraven).

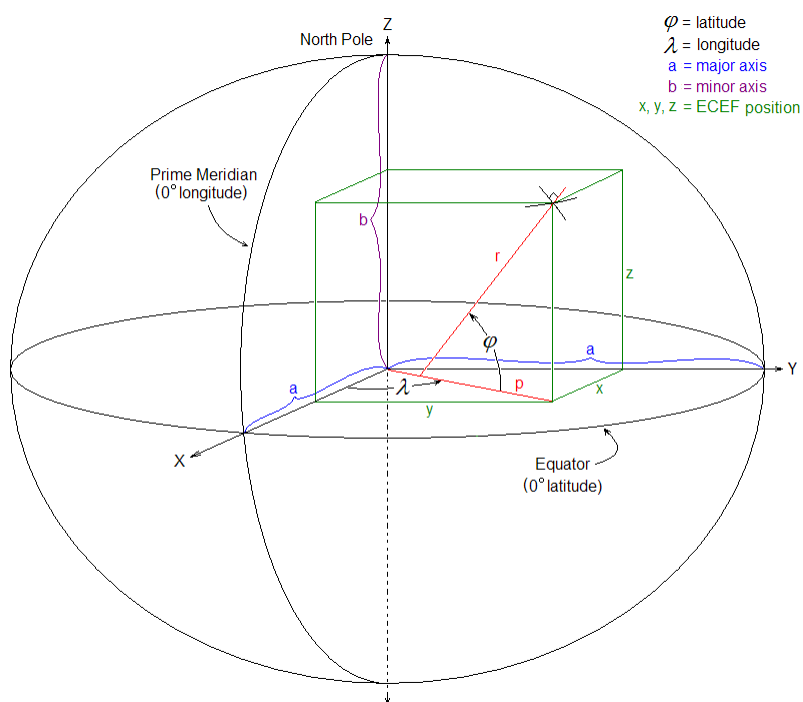
Zeměpisná šířka je definována jako úhlová vzdálenost mezi rovinou určenou rovníkem a k danému tělesu kolmou přímkou, která protíná daný bod (rovnoběžky). Existuje další tzv. geocentrická šířka, která určuje úhel mezi rovinou rovníku a přímkou, která prochází daným bodem a středem země. V případě koule jsou obě definice ekvivalentní, nicméně u elipsoidu nikoliv, tj. příмка by středem země neprošla viz obrázek 2.2. Rozsah hodnot je 0–90° s orientací na sever (severně od rovníku) záporné hodnoty směrem jižním; značí se řeckým znakem  $\varphi$ . Pro snazší porozumění (čtení) se užívá označení „severní šířka“ - znakem N (North), resp. „jižní šířka“ - znakem S (South).

Zeměpisná délka je stejná jak u koule, tak u elipsoidu; jedná se o úhlovou vzdálenost mezi rovinou určenou nultým poledníkem (protíná severní a

<sup>8</sup>Gravitační centrum/těžiště - spočteno pomocí (geodetických) satelitů

<sup>9</sup>Neboli základní poledník; rozděluje zemi na dvě polokoule (hemisféry). GPS využívá IRM (International Reference Meridian) definici nultého poledníku - ta se mírně liší od Greenwich definice (cca 100 m).

## 2. GEODETICKÝ MODEL ZEMĚ A GEOLOKACE



Obrázek 2.2: Zeměpisné souřadnice vs. ECEF [14]

jižní pól) a poledníkem protínajícím daný bod. Používá se pravá orientace, tj. kladné hodnoty směrem na východ v rozsahu  $0 - 360^\circ$ , případně negativní ve směru na západ; značí se řeckým znakem  $\lambda$ . Podobně jako u šířky je možné použít značení „východní délky“ - E (East), resp. „západní délky“ - W (West).

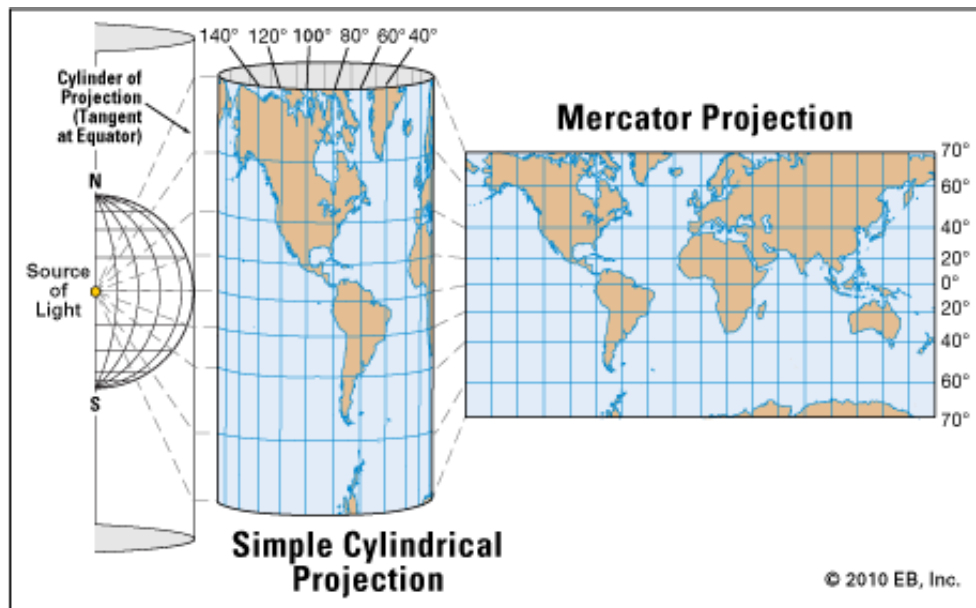
Souřadnice mohou být zapsány více způsoby; těmi nejčastějšími jsou:

- *Úhlové jednotky* - rozdělení na stupně, minuty, sekundy a určení orientace (sever/jih, východ/západ); např.  $50^\circ 04' 47.2''\text{N } 14^\circ 25' 47.1''\text{E}$ .
- *Stupně v desetinné podobě* - orientace je určena pozitivním (neuvádí se) nebo negativním znaménkem, např.  $50.079768, 14.429735$ .

Dále je nutné přidat informaci o výšce, která polohu upřesní - takto ucelená informace se označuje *LLH* (Latitude, Longitude, Height). Výška může být definovaná různými způsoby; GPS, resp. WGS 84 využívá elipsoidní výšku viz předchozí sekce 2.2.

### 2.4 Projekce do 2D

Pro některé účely je vhodné převést prostorové 3D souřadnice do 2D roviny - typicky mapové zobrazení (projekce). Dalším důvodem jsou výpočty v rovině, např. při určení pozice. K tomu dochází pokud není dostatek informací, resp.



Obrázek 2.3: Mercatorova projekce [16]

rovnic pro získání všech proměnných v prostoru nebo je některý rozměr nepotřebný (např. výška). Nutno podotknout, že každá projekce má za následek mírné zkreslení; je tedy vhodné zvolit takovou projekci, která pro danou aplikaci nabízí - alespoň v užitném rozsahu - přijatelnou odchylku (požadovaných vlastností).

Zobrazení se odlišují polohou konstrukční osy a zobrazovací plochou; např. v normální (pólové), příčné (transverzální) nebo v obecné (šikmé) poloze, resp. jednoduchá (azimutální, válcová, kuželová) nebo obecná (nepravá, polykónická, víceplošná) zobrazení. Klasifikace dle zkreslení je následující: délkojevná (ekvidistantní; v daném směru délky nedochází ke zkreslení), plochojevná (ekvivalentní; nezkruslují se plochy), úhlojevná (konformní; nezkrusluje úhly - zachovává tvary) a vyrovnávací (kompenzační; navrženo tak, aby zkreslení úhlů a ploch bylo v rovnováze). [15]

### 2.4.1 Mercatorovo zobrazení

Jedná se o válcové zobrazení s normálně vedenou konstrukční osou. To znamená, že válec, který je základem tohoto zobrazení, má totožnou konstrukční osu jako je ta zemská (kolmá na rovník) viz obrázek 2.3.

Směrem od rovníku se zkreslení prohlubuje a je výrazná disproporce ve velikostech (např. Grónsko je velké jako Afrika). Při větším přiblížení (menší oblast) jsou úhly a tvary zachovány (např. na úrovni ulic), jedná se tedy o konformní zobrazení. [17] Tento fakt je možné pozorovat i v Google mapách, které používají mírnou obměnu tohoto zobrazení - tzv. webovou verzi.

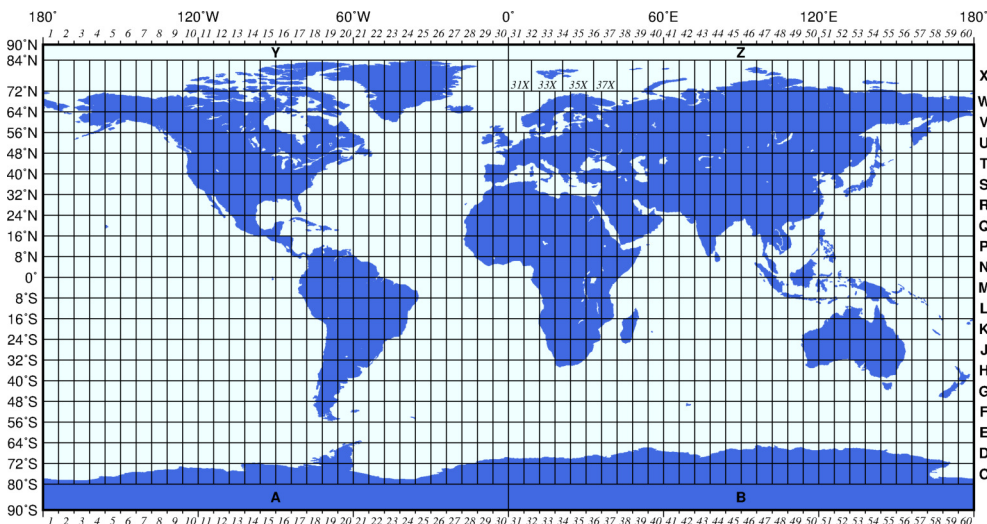
## 2. GEODETICKÝ MODEL ZEMĚ A GEOLOKACE

Webová verze (Web Mercator) se stala velmi běžnou právě kvůli využití v Google mapách a postupně i v dalších, např. Bing, OpenStreetMap nebo Esri. Oproti klasickému Mercatorovu zobrazení využívá rovnice po kouli a ne pro elipsoid; zakládá však na WGS 84 souřadnicích (převádí souřadnice z elipsoidu na sférické - chová se k nim jakoby byly definovány na kouli). Rozdíl je viditelný až při větším přiblížení a dále od rovníku. Póly jsou projektovány jako nekonečno a proto je není možné zobrazit (cca kolem 85° severní a jižní šířky). [18] Nejedná se o standart a jednotlivé implementace/API (Application Programming Interface) se mohou lišit.

### 2.4.2 UTM

UTM (Universal Transverse Mercator) neboli Univerzální transversální Mercatorův systém souřadnic. Transverzální projekce umožňuje zobrazit oblasti s větším rozsahem sever-jih na úkor možností východ-západ. To je dáno polohou konstrukční osy - leží v rovině rovníku (na rozdíl od normální polohy). Zkreslení velikosti, vzdálenosti, směru a plochy se navyšuje směrem od centrálního poledníku. Rozumná přesnost je pouze v okolí 15°. [17]

UTM problém minimalizuje horizontálním rozdělením zemského povrchu do 60 zón po 6° zeměpisné délky (zúžení šířky projekce). Každá z těchto zón má vlastní středový poledník a počátek, který je definován jako průnik s rovníkem. Pohyb po ose x je kladný směrem na východ od daného poledníku a pohyb po ose y je kladný směrem sever od rovníku. Zkreslení se začíná navyšovat v oblastech severně od 84° a jižně od 80°, kde se UTM nahrazuje tzv. UPS (Universal Polar Stereographic) projekcí, jež zabraňuje konvergenci poledníkových čar. A také v hraničních oblastech, kde zóna přechází v jinou. Počátek jednotlivých zón a rozdělení zemského povrchu viz obrázek 2.4.



Obrázek 2.4: UTM zóny [19]

## Současný stav lokalizačních metod

Dříve se pod pojmem lokalizace myslela převážně tzv. *outdoor* lokalizace (pro venkovní prostory), nicméně s nástupem rozlehlých obchodních center, skladů, letišť aj., se začala rozmáhat a ve větší míře prosazovat také tzv. *indoor* lokalizace (pro vnitřní prostory). Současně probíhající robotizace a automatizace zapříčinila diverzifikaci trhu a začaly přibývat nové způsoby umožňující/podporující proces lokalizace - např. proximitní senzory.

Tato kapitola shrnuje hlavní technologie a metody, které mohou být chápány jako stavební kameny pro konečné a komplexní (hybridní) řešení, vyhovující specifickým - aplikačním - požadavkům jednotlivých zákazníků. Navržený lokalizační systém se snaží, co nejvíce, vyhovět konkrétnímu UC; ať už z pohledu ceny, spotřeby, přesnosti, komfortu užívání či snadnosti a rychlosti při nasazení.

### 3.1 Globální družicový polohový systém

GNSS (Global Navigation Satellite System) označuje uspořádaný systém satelitů, jenž z vesmíru (oběžných drah kolem země) vysílá informace nutné pro určení pozice a přesného času. GNSS přijímač naslouchá na daných frekvencích, tyto informace přijme a za pomoci matematických algoritmů vypočítá polohu a synchronizuje svůj čas<sup>10</sup>. Mezi nevýhody patří vysoká energetická náročnost, a z toho plynoucí nízká životnost (zařízení s omezeným zdrojem energie - akumulátor/baterie). Uvnitř budov je problematický kvalitní příjem a musí být doplněn/nahrazen jinými technologiemi. Pro zpřesnění jsou používány systémy jako je GBAS (ground-based augmentation systems) a SBAS (satellite-based augmentation systems). Další možností je kombinovat jednot-

<sup>10</sup>Zároveň se jedná o velmi přesný zdroj času používaný k synchronizaci např. v počítačových sítích (NTP server).

livé GNSS - běžně užívanou kombinací je GPS a GLONASS (využívané i ve smartphone zařízeních).

#### 3.1.1 Současné GNSS

Z definice GNSS by se mělo jednat o systém s opravdu globálním pokrytím (existují i regionální systémy, např. NAVIC, QZSS). BeiDou se - vypuštěním dalších satelitů - mezi GNSS zařadil nedávno [20] a navýšil tím jejich počet:

- *GPS (USA)* - podrobněji v podsekcí 3.1.2,
- *GLONASS (Rusko)* - po GPS druhý plně operativní systém,
- *Galileo (EU)* - plně operativní v roce 2020, aktuálně 18 satelitů na orbitální dráze,
- *BeiDou (Čína)* - stále ve fázi nasazení (další satelity budou vypuštěny).

GPS byl prvním funkčním a pro širokou veřejnost přístupným GNSS systémem; také proto se stal de facto synonymem (GNSS přijímače jsou často nazývány jako GPS moduly, i když umožňují přijímat a zpracovávat další GNSS), a to nejen pro GNSS, ale také pro outdoor lokalizaci (široké a univerzální využití). Proto jsou základní principy popsány právě na GPS systému, viz následující podsekcce.

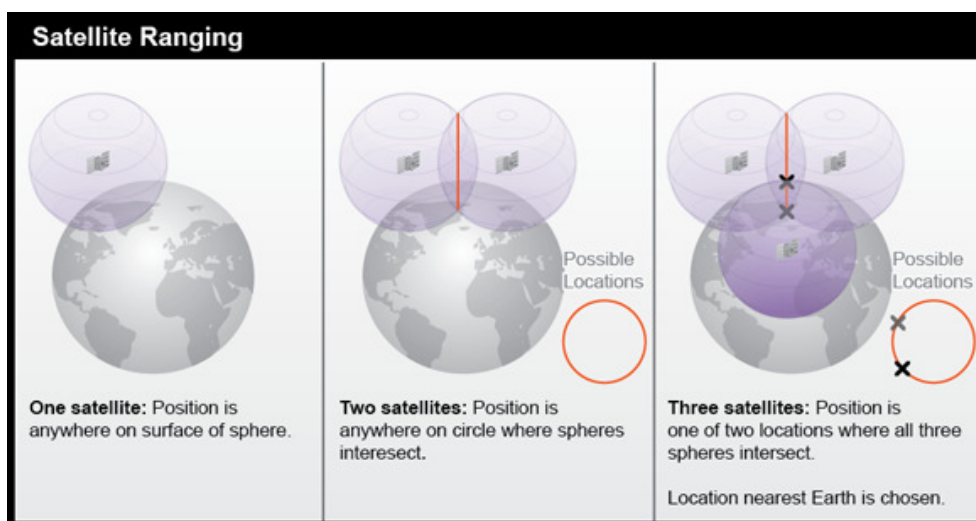
#### 3.1.2 Princip GPS

GPS disponuje soustavou 24 satelitů, instalovaných spolu se zdrojem velmi přesného času (rubidiové, cesiové hodiny), který je navíc každý den upravován (korekce) s referenčními pozemními hodinami (jednotlivé satelity jsou synchronizovány jak mezi sebou, tak se zemí). Pozice těchto satelitů (spolu s rychlostí, výškou) je také známa - jedná se o základní informace nutné pro určení pozice na zemském povrchu. Jako referenční model je využíván WGS 84 a při kalkulacích ECEF souřadnicový model viz kapitola 2. Zpřesňující metody (A-GPS, DGPS) nejsou předmětem této podsekcce.

Principiálně je GPS založena na TOA (Time of Arrival) a metodě nazývané *trilaterace* (někdy nesprávně zaměňována s triangulací<sup>11</sup>). Tedy na čase uplynulém od odeslání k přijetí, resp. vzdálenosti mezi vysílačem (satelit) a přijímačem (GPS modul). Satelity slouží jako referenční body; vždy ve stejný čas odešlou signál s informací, která mj. obsahuje přesný (řádově v nanosekundách - při rychlosti světla cca  $3 \times 10^8$  m/s se jedna nanosekunda rovná 30 cm) čas odeslání. GPS modul si zaznamená čas přijetí a pokud by měl čas přesně synchronizovaný s ostatními satelity, mohl by vypočítat vzdálenost od vysílače (reprezentovanou koulí - 3D prostor). Při přijmutí signálu od dalšího satelitu

---

<sup>11</sup>Primárně využívá úhlů, nikoliv vzdáleností.



Obrázek 3.1: GPS princip lokalizace - geometrická představa založená na pseudovzdálenostech [21]

by získal další kouli - průnikem je kružnice, průnik s další koulí určí dva body v prostoru (z těch už nebývá problém určit ten správný, jelikož musí ležet na zemském povrchu, modelovaném elipsoidem; případně pomůže minulá poloha, resp. historická data) viz 3.1.

Předpoklad, že je čas synchronizovaný s přijímačem je - typicky - nesplnitelný, a proto je do výpočtu zahrnut časový offset, který vyjadřuje odchylku (je stejná pro všechny naměřené hodnoty) od přesného satelitního času (UTC). Nejsou tedy počítány pravé vzdálenosti, ale tzv. *pseudovzdálenosti*. Pro výpočet přesného času je zapotřebí další pseudovzdálenost (satelit), tj. pro určení 3D polohy je zapotřebí 4 satelitů. Pokud by byl čas přesný, zajistila by 4 hodnota průnik v jednom bodě. GPS přijímač pozná, že je čas nepřesný a pokusí se o korekci, pomocí odečtení časového offsetu - tj. hledá hodnotu, která bude po odečtení od všech měření vyhovovat jedinému bodu (průnik 4 koulí). [22, 23]

Matematický zápis problému - použity jdou osy  $x$ ,  $y$ ,  $z$  - je následující<sup>12</sup> [24, 22]:

$$\delta_i = c(t_{r,i} - t_{t,i} - t_c) = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}$$

pro  $i \in \hat{n}$ , kde

$n$  = počet satelitů,  
 $\delta_i$  = vzdálenost od  $i$  satelitu,

<sup>12</sup>Rovnice slouží pro ukázkou principu a jsou zde vynechány externí podmínky, které výsledek ovlivňují (průchod signálu atmosférou, pod jakým úhlem vstoupil do ionosféry, reflexe atp.).

$t_{t,i}$	= čas odeslání z $i$ satelitu,
$t_{r,i}$	= čas přijetí od $i$ satelitu,
$t_c$	= časová odchylka přijímače od UTC,
$(x_i, y_i, z_i)$	= souřadnice (ECEF) $i$ satelitu,
$(x, y, z)$	= hledané souřadnice GPS modulu,

pro pseudovzdálenosti  $\rho_i$  tedy platí (při 4 satelitech)

$$\begin{aligned}\rho_1 &= \sqrt{(x_1 - x)^2 + (y_1 - y)^2 + (z_1 - z)^2} + ct_c, \\ \rho_2 &= \sqrt{(x_2 - x)^2 + (y_2 - y)^2 + (z_2 - z)^2} + ct_c, \\ \rho_3 &= \sqrt{(x_3 - x)^2 + (y_3 - y)^2 + (z_3 - z)^2} + ct_c, \\ \rho_4 &= \sqrt{(x_4 - x)^2 + (y_4 - y)^2 + (z_4 - z)^2} + ct_c.\end{aligned}$$

Výsledná soustava je nelineární a obsahuje 4 neznámé  $(x, y, z, t_c)$  - možné řešit explicitním nebo iteračním přístupem. Tedy provést linearizaci (Taylor) a poté iterační algoritmus (např. LS - Least Square) aplikovat na soustavu lineárních rovnic. [22]

Matematický princip je podobný jako u LoRa lokalizace, pouze inverzní z pohledu přijímače/vysílače viz 3.2. Vhodnou (algebraickou) úpravou rovnic lze získat odlišnou geometrickou interpretaci:

- (a) *Koule* - pokud je uvažována zjednodušená situace pseudovzdáleností (obsahujících časovou chybu).
- (b) *Dvojdílný hyperboloid* - po odečtení vzdáleností mezi  $i$  a  $j$  satelitem získáme rozdíl, který odpovídá polohám na hyperboloidu [25] - princip *TDoA* (Time Difference of Arrival) a *multilaterace*, více viz sekce 3.2.
- (c) *Světelný kužel* - při představě 4D prostoru, kde čtvrtý rozměr představuje čas a vrcholy jsou jednotlivé satelity . [26, 27]

## 3.2 LoRa geolokace

LoRa neboli LoRaWAN geolokace se řadí mezi LPWAN lokalizační metody, které fungují na stejných základech (tj. např. Sigfox nabízí podobné služby). Název sekce je LoRa jakožto specifikace L1 vrstvy - na implementaci L2-L3 nezáleží (princiálně se neliší). V kontextu této práce se ovšem uvažuje LoRaWAN protokol, proto tedy LoRaWAN lokalizace (v podsekcí 3.2.3 je uveden příklad, jenž využívá pouze LoRa). Podobnost s radarovými systémy (odvozená od CSS modulace) umožňuje přesnou detekci přijímaného signálu a přiřazení časové značky. Navzdory tomu stále trpí negativními aspekty, jenž plynou z fyzikální podstaty rádiových technologií (podrobněji v podsekcí 3.2.2).



### 3.2.1 Princip LoRa lokalizace

Lokalizace v LoRa sítích je založena na podobných principech jako GPS (případně mobilní sítě GSM, WiFi viz dále) avšak v obráceném směru. V případě GPS známe polohu satelitů, které ve stejný čas vysílají nějaký kód, resp. signál, a samotné koncové zařízení - přijímač / GNSS modul - signál přijme, data vyhodnotí a vypočte polohu. U LoRa, jakožto IoT sítě, je nemyslitelné, aby složité výpočty prováděl - z konstrukčního, ale i funkčního hlediska - senzor. Tato odpovědnost je převedena na stranu sítě (aplikačního serveru), která disponuje výpočetním výkonem a potřebnými metadaty, jenž slouží jako vstup do lokalizačních algoritmů.

Senzor odešle libovolná data a ty jsou v podobě rádiového signálu přijaty několika LoRa branami neboli GW (Gateway). Brány mají synchronizovaný čas a známou polohu (zpravidla zajištěnou GNSS modulem). Následně se provede přiřazení časové značky (spolu s informacemi jako SNR či RSSI<sup>13</sup>) a v podobě metadat se tyto údaje odešlou uplinkem na síťový server, který provede agregaci (viz 1.2) a celou strukturu dat předá aplikačnímu serveru.

Aplikační server má k dispozici sadu časových značek, které po odečtení vyjadřují vzdálenost, o kterou je jedna (ze dvou bran, jejichž časové značky od sebe byly odečteny) z bran blíže, případně dále od senzoru. Tato technika se nazývá *TDoA* (Time Difference of Arrival) a metoda, která s touto hodnotu pracuje *multilaterace*. Body které vyhovují této konstantní vzdálenosti, o kterou jsou vždy blíže/dále k jedné bráně (relativně vzhledem k senzoru), tvoří v 2D geometricky hyperbolu a v 3D dvojdílný hyperboloid. Proto jsou výpočty hyperbolické a celá metoda někdy nazývána jako *hyperbolická lokalizace*. [28]

Tento fakt plyne přímo z definice, která říká, že rozdíl vzdáleností od jakéhokoliv bodu hyperboly k ohnisku je konstantní viz obrázek 3.2, neboli:

$$|r_1 - r_2| = 2a$$

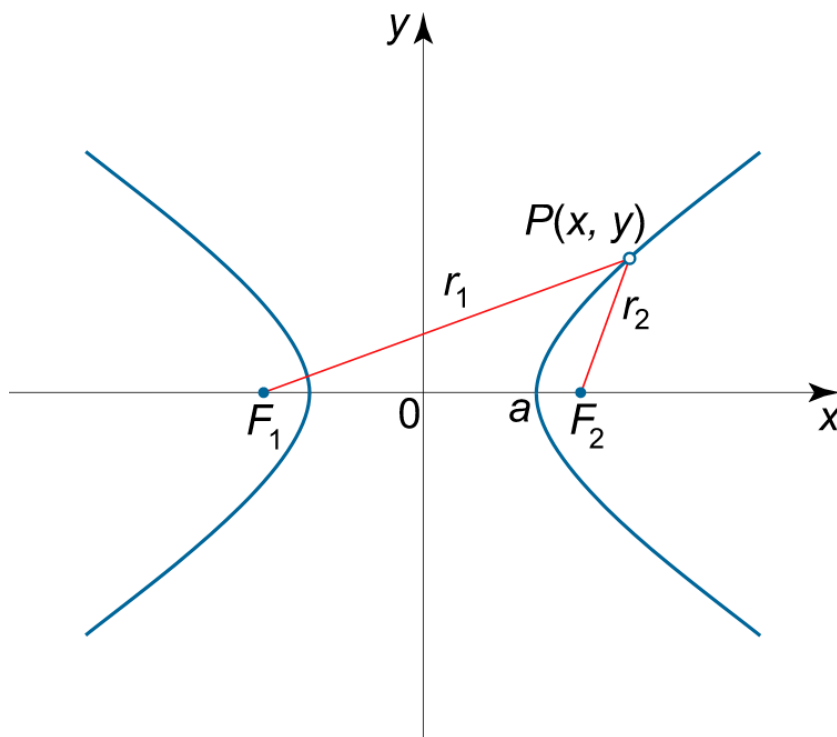
Při odstranění absolutní hodnoty vznikne pouze jedna větev („polovina“) hyperboly, tj. jedna z bran je vždy blíže - vztah mezi nimi není obousměrný. Pro dvojdílný hyperboloid v 3D prostoru je situace obdobná.

Pro určení pozice v 2D prostoru je potřeba, aby signál přijaly alespoň tři brány (získáme dvě nelineární rovnice - dvě TDoA hodnoty relativní vzhledem k jedné společné). V 3D je zapotřebí minimálně čtyř bran (tři rovnice pro vyřešení třech proměnných - souřadnic v prostoru). Konkrétní algoritmy a výpočty v sekci 5.1.

### 3.2.2 Negativní dopady

LoRa má, podobně jako ostatní lokalizační metody, založené na rádiových vlnách, jisté nedostatky a fyzikální mantinely (mimo jiné).

<sup>13</sup>Signal-to-Noise Ratio a Received Signal Strength Indication.



Obrázek 3.2: Hyperbola a konstantní vzdálenost - základní princip TDoA [29]

- *Vícecestné šíření* - jén málokdy je v zabydlelých oblastech (velká města) možné přijmout signál přímou cestou a dochází k odrazům. Pokud je signál přijat přímou cestou a odraženou, je ten přímý ovlivněn (sečtou se a dojde ke změně amplitudy) a zpožděn. Schopnost rozlišit odražený signál je určen funkcí přenosové šířky  $\frac{c}{125\text{Khz}} = 2398$  m. V takto úzkém pásmu (125 Khz) je obtížné (oproti širšímu pásmu viz podsekcce 3.3.4) odrazy rozpoznat; konkrétně by musel být rozdíl v uražené vzdálenosti cca 2.4 km. [30]
- *Odražený signál* - doba přenosu se může mnohonásobně navýšit (odrazy mezi panelovými domy, mrakodrapy) a je obtížné, až nemožné rozpoznat, že se jedná o odraz (v některých případech lze použít RSSI, SNR a statistické metody).
- *Malý výkon* - pásmo je regulované a z principu LPWAN je vysílací výkon nízký. Schopnost přijmout přímý signál je funkcí

$$(TIME_{onAIR}) \times BANDWIDTH \times POWER,$$

vysílací čas (může být dynamický viz sekce o ADR) je dostatečný, ale výkon nikoliv. Je tedy obtížné přijmout signál bez odrazu, pokud se nejedná o přenos za přímé viditelnosti (line-of-sight). [30]

- *Špatná geometrie/umístění bran* - nevhodně rozmístěné brány (v jedné rovině, přísmce) komplikují matematické výpočty (potřebná matice bude singulární atp.). Výška umístění a typ zvolené antény (příliš velký zisk a vysoko umístěná brána - vznikají „hluchá“ místa) mají také značný vliv.

### 3.2.3 Reálné výsledky a využití

LoRa lokalizace se uplatní tam, kde není vyžadována vysoká absolutní přesnost (řádově jednotky až desítky metrů). Typicky aplikace geofencingu, prvotní odhad pozice, mapování na předem známe objekty v databázi - stačí rozeznávat několik stavů, např. zdali je se objekt nachází na pobočce nebo je na cestě, nebo pokud známe trasu (doprava, nákladní auta), tak je možné provést spárování s nejbližším místem na trase (mnohdy stačí přibližná poloha a odhad času do příjezdu atp.). Další aplikací je doplnění stávajících řešení, pro potřeby výpadku či selhání (záloha) nebo jako součást komplexního/hybridního systému pro lokalizaci (např. doplnění nějaké indoor metody).

Nejvíce se geolokaci věnuje samotný Semtech. Pro tento účel navrhl referenční design tzv. druhé generace (verze 2.0) LoRaWAN brány, který zajišťuje dostatečně přesné časové značky - timestamps - přijatých paketů [31] (řádově 10-100 nanosekund). Jedná se o vylepšení (mj. přidán specifický FPGA a DSP<sup>14</sup> obvod) oproti první verzi, která pro tento účel nebyla (primárně) navrhována a časové rozlišení je řádově v jednotkách  $\mu s$  (což představuje limitující faktor). Pro obě verze platí nutnost společné časové báze, tedy přítomnost GNSS/GPS modulu. Vylepšené brány prozatím poskytuje např. společnost Kerlink a Cisco, spolu s vlastním systémem pro lokalizaci (Kerlink vyvinul Wanesty™ Geolocation) [32, 33].

Semtech jako první prezentoval tzv. jednorázový „nano-tag“, který disponuje miniaturní tištěnou baterií a otevírá prostor pro nové UC a aplikace [34]. Dále provedl vlastní testy ve městě a v odlehlých oblastech, které dosáhly přesnosti 125 m, resp. 20 m (střední hodnota při čtyřech odeslaných paketech) [35].

LoRa geolokace je využívána v Tanzanii (Mkomazi National Park) pro ochranu ohrožených černých nosorožců. Mále rozměry umožňují lokátor implantovat přímo do rohů. Lokátory jsou imunní proti GPS rušičkám, které jsou používány pytláky a zároveň umožňují vyšší periodu odesílání (dáno nižší spotřebou). Lokátory jsou používány i v kombinaci se solárními panely, např. na autech personálu. Přesnost dosahovala 50 m. V plánu je nasazení do dalších rozlehlých oblastí/parků. [28]

Dalším případem je využití při procesech stavebních prací v Paříži. Lokátor disponuje několika senzory a detekuje pohyb (pohybový senzor) stavebního auta a nakládání/vykládání nákladu (gyroskop), celý proces je vylepšen NFC

<sup>14</sup>Field-programmable gate array a digital signal processor.

(Near-field Communication) rozhraním, které slouží pro identifikaci lokátoru a nahrání dat jako je typ materiálu a množství. [36, 28]

Sledování v dopravě se věnuje společnost *imec*, která vyvinula technologii kombinující LoRa a DASH7 pro outdoor, resp. indoor lokalizaci. Řešení je doplněno RFID (Radio Frequency Identification) metodou, která objekt identifikuje při vstupu do vnitřních prostor (sklad, parkoviště) a aktivuje indoor lokalizační metodu. Všechny tyto senzory a rádiové moduly jsou integrovány v jednom lokátoru. [37]

V Nizozemsku společnost *Allsetra* používá miniaturní LoRa lokátor (Villoc LoRa tag) pro sledování malých objektů (jako záložní řešení v případě krádeže). To umožňuje - ve světě první - celostátní LoRa síť nasazená národním operátorem KPN [38]. Pomocí LoRa geolokace bylo, v časovém horizontu několika hodin, nalezeno vybavení v hodnotě 31,000 EUR. [39]

## 3.3 Přehled dalších technologií

### 3.3.1 Mobilní sítě

Sítě GSM (Global System for Mobile Communications)/EDGE/GPRS, 3G, LTE (NB1, CAT1, CATM1) aj. Označovány jako LCS (Location Services) dle standardů 3GPP.

Primárně jsou využívány technologie *U-TDOA* (Uplink-Time Difference of Arrival), *E-OTD* (Enhanced Observed Time Difference), *OTDOA* (Observed Time Difference Of Arrival) - podobné LoRa, které jsou založeny na technikách *multilaterace* (využívá *TDoA*). Dále se využívá triangulace - *BTS* (Base Transmit Station) někdy využívají směrové antény (např. 3 x 180°). RSSI se používá pouze pro hrubý odhad pozice, případně pro počáteční odhad pokud není známa poslední poloha. Pro snížení energetické spotřeby jsou využívány techniky pasivního skenování.

S každou novou generací 2G/3G/4G jsou dostupná vylepšení a hybridní metody lokalizace zvyšující přesnost (principy jsou ovšem podobné).

### 3.3.2 Wifi

Existují databáze s namapovanými souřadnicemi a jednotlivými WiFi AP (Access Point), resp. BSSID (Basic Service Set Identification) neboli MAC (Media Access Control) adresa síťové karty. Proces mapování je zpravidla autonomní - uživatel má na zařízení aktivovaný GNSS (případně jiný zdroj určení pozice) a po připojení nebo jen oskenování okolních WiFi SSID jsou získané informace spárovány a odeslány. Jednu z prvních DB vlastnil Skyhook [40]. Postupně si vlastní DB vytvořil Google (za pomoci Androidu), Apple (iOS), Microsoft (Windows Phone) aj. Takových DB existuje více, ale nejsou tak rozsáhlé (např. komunitní DB).

Ve větších městech se jedná o poměrně přesnou metodu s nízkou spotřebou. Využívána je síla signálu RSSI (Received Signal Strength Indication), doba letu ToF (Time of Flight), úhel (triangulace - MIMO AP) nebo fingerprinting. Existují přesné indoor lokalizační systémy, např. od společnosti Cisco.

### 3.3.3 BLE

Bluetooth Low Energy neboli Bluetooth Smart. Součástí Bluetooth specifikace (od verze 4.0). Díky svému rozšíření a relativně nízké ceně je poměrně oblíbený. Nové specifikace umožňují větší dosah. Používá např. Tracko [41] v kombinaci s tzv. „Bluetooth beacon“ zařízeními (proximitní lokalizace).

### 3.3.4 UWB

Vyžívané ve spoustě proprietárních řešení. Je možné použít i ve zhoršených podmínkách v průmyslu - vysoká přesnost (řádově 10 cm). Navíc lze detekovat odrazy (široké pásmo umožňuje efektivnější oddělení odražených signálů). Využívá např. Sewio [42] a Pozyx [43].

### 3.3.5 RFID

Např. NFC. Doplnující technologie. Např. pro detekci pohybu v určitém místě (náklad projede dveřmi do terminálu a lokalizační systém se přepne na indoor verzi).

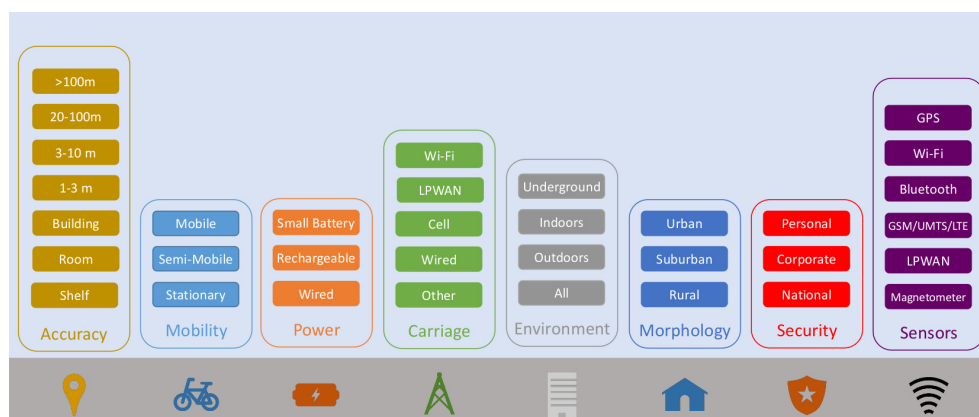
### 3.3.6 Pohybové a směrové senzory

Zástupci jsou akcelerometr, gyroskop a magnetometr. Slouží jako doplňující metody - nejsou příliš přesné a udávají relativní změnu polohy. Tento proces se v navigaci nazývá *dead reckoning*, případně *PDR*<sup>15</sup> (Pedestrian dead reckoning). Dobře poslouží např. při výpadku primární metody, jako tzv. hrubý odhad (pokud není žádná jiná možnost). Typickou situací je dočasný vstup a pohyb sledovaného objektu do oblasti, kde není signál (auto v tunelu). Dalším využitím je zpřesnění ostatních metod lokalizace nebo nabídka dalších funkcionalit (např. orientace při navigaci). Kombinací akcelerometru, gyroskopu a magnetometru vznikají tzv. 6-osové, resp. 9-osové pohybové senzory.

Dráha je spočtena pomocí dvojného integrálu ze zrychlení, gyroskop slouží pro určení rotace, a k doplnění informace o absolutním směru lze využít magnetometr. Akcelerometr zároveň poskytuje informaci o poloze vůči středu země, to je dáno informací o tíhovém zrychlení (gravitace), tj. neposkytuje údaj o rotaci, pokud se zařízení otáčí kolem gravitace - typicky osa  $z$  u gyroskopu. Existuje řada filtrů, které tyto hodnoty kombinují pro zpřesnění a vyhlazení průběhu.

---

<sup>15</sup>Využívá principu krokometru, k určení směru slouží magnetometr.



Obrázek 3.3: Lokalizační technologie - členění [46]

### 3.3.7 Nové technologie

#### 3.3.7.1 LED

Společnost *Philips* (Lighting) přišla se způsobem jak rozšířit svůj obchodní model a nabídku služeb, pomocí stávajících produktů. Jednotlivé LED (Light-emitting Diode) vysílají unikátní světelné kódy, zařízení přijme více takových signálů a na základě toho je schopno dopočítat svou polohu. Záložní řešení využívá BLE. Jedná se tedy o indoor lokalizační metodu. Výhodou je zabudování do existující infrastruktury (při výměně starých žárovek za nové), a z toho plynoucí nižší cena. [44]

#### 3.3.7.2 Obrazové zpracování

Je využíváno tzv. „computer vision“. Nejdříve musí proběhnout namapování daných prostor neboli fáze učení (na základě vizuálních značek/tagů - např. QR kód). Toto řešení implementuje a nabízí společnost *Accuware*. [45]

## 3.4 Porovnání

Vždy nemusí platit „nejlepší“ = „nejpřesnější“. Faktorů, které zasahují do výběru lokalizační metody je spousta a nejprve je nutné provést kvalitní analýzu. Z té je následně možné vyvodit možná řešení při současných technologických možnostech a stanovit minimální průnik s požadavky. Obrázek 3.3 znázorňuje základní kategorie do kterých lze jednotlivé techniky a technologie umístit.

Obecně si najde (LoRa) uplatnění v odlehlých oblastech (s např. stíženým přístupem), geofencingu, dopravě (může probíhat mapování na známou trasu - není potřeba přesně vědět kde se nachází), nebo ve městech kde bude síť instalovaných bran hustší. Jedná se o otevřený standard a síť může provozovat každý, to je výhodou i nevýhodou (kompatibilita, roaming). Není potřeba SIM

karta a lokalizace může být v podstatě „zadarmo“. Úspěch bude silně závislý na dalším vývoji a konkurenci ze strany mobilních (od stávajících operátorů) technologií jako je LTE-CATM1.

LoRa by při komerčním (a kritickém) nasazení, v některých směrech, zůstávala. Konkrétně v těchto oblastech: *bezpečnost*, *aktualizace* (patch), *spolehlivost a QoS* (Quality of Service), *dostupnost* (pouze class B a C), *latence*, *mobilita* a *globální pokrytí*. Některé z těchto bodů souvisí s nelicencovaným pásmem (nelze nic vynucovat, pouze dodržovat), v tom mají výhodu konkurenční LTE varianty.





## Potřebná infrastruktura

V této kapitole je popsána příprava infrastruktury, umožňující reálné otestování lokalizace pomocí testovacího senzoru (spolu s použitým nastavením a technologiemi). Tedy celý řetězec LoRaWAN sítě, jenž se účastní lokalizačního procesu.

### 4.1 LoRaWAN brána

LoRaWAN brána v celé síti sehrává transparentní roli. Paket přijatý přes rádiové rozhraní demoduluje a přiřadí mu jistá metadata. Z pohledu geolokace je kritickým údajem přesný čas, který lze nejjednodušeji (s rozumnou nejistotou) získat z GNSS/GPS modulu - ten zároveň poskytuje informaci o poloze<sup>16</sup>, jenž je taktéž esenciální. Jedná se o návrh brány tzv. první generace, která neumí tak přesné přiřazení časových značek (demodulovaným paketům), jako brána druhé generace viz 3.2.3. S touto branou je dosažitelná přesnost v jednotkách  $\mu\text{s}$ .

Z důvodu absence (u většiny dostupných LoRaWAN bran) GNSS/GPS a možnosti synchronizace přes PPS (Pulse-per-second signal) výstup, byla v CESNETu sestavena vlastní brána.

#### 4.1.1 Konstrukce

Základ brány se skládá z následujících komponentů:

- *Raspberry Pi 3 (RPi3)* - kontrolér celé brány. Pomocí IP (Internet Protocol) sítě přeposílá data z rádiového rozhraní na síťový server. Slouží pro připojení dalších senzorů a monitoring - správu (úprava konfigurace). Univerzální a rozšiřitelné což je pro testovací účely vhodné.

<sup>16</sup>Polohu lze přiřadit i staticky (centrálně, např. na aplikačním serveru podle identifikátoru brány), čímž se minimalizuje riziko chybných údajů (pokud brána poskytne nepřesnou polohu, bude tím výpočet ovlivněn a znehodnocen). Při masovém nasazení ovšem nepraktické.

- *Rádiový koncentrátor iC880A* - od společnosti IMST GmbH. Umožňuje přiřazení časové značky na základě GNSS/GPS PPS vstupu a příjem až 8 paketů paralelně (automaticky detekuje použitý DR neboli SF). Využívá Semtech SX1301 signálový procesor a dva rádiové moduly SX1257. Komunikace s RPi3 probíhá pomocí SPI (Serial Peripheral Interface) protokolu.
- *iC880A LoRaWAN shield* - zajišťuje spolehlivé propojení RPi3 a koncentrátoru nejkratší cestou (větší vzdálenost může být u SPI problém). Disponuje napěťovým regulátorem, který snižuje napětí z PoE (Power over Ethernet) splitteru (12V) na požadovaných 5 V, a senzorem měřící teplotu a vlhkost (pro účely monitoringu).
- *GNSS modul* - poskytuje přesný čas a informaci o pozici (geodetické souřadnice). PPS pulz připojen na samotný koncentrátor (rádiová část) a Tx/Rx přes UART (Universal Asynchronous Receiver-Transmitter) do RPi, kde jsou NMEA (National Marine Electronics Association) věty parsovány klientským programem. Modul je nastaven do stacionárního módu což může zlepšit přesnost polohy a času.

Konektivita a napájení jsou zajištěny aktivním PoE (802.3at) připojením. Vše je umístěno ve voděodolném boxu, který je doplněn filtrem pro kompenzaci tlaku (vlhkost) a panelovým N konektorem pro připojení externí antény (868 MHz). Využívány jsou tři modely antén: Procom CXL 900-3LW-NB/868 MHz, SIRIO GP 868 C a IK-Telecom Oy A10-868; disponují odlišným ziskem, dle umístění a lokality. Vnitřní pohled na bránu lze vidět na obrázku 4.1.

#### 4.1.2 Nasazení

Z důvodu snadného a bezpečného nasazení bylo zvoleno řešení využívající *resin.io* [47] programovou nadstavbu. Jedná se o projekt, který cílí na Linuxová - převážně - IoT zařízení, a zjednodušuje monitoring, správu, zabezpečení a celý proces nasazení.

*Resin.io* pro danou aplikaci vytváří oddělené virtualizované prostředí (využívá Docker) a při startu vytvoří VPN (Virtual Private Network) spojení tak, aby bylo zařízení de facto odpojené od Internetu (a nebylo možné ho zneužít). Všechny porty jsou vypnuty, lze povolit SSH (Secure Shell), ale ten by umožňoval přístup pouze k virtualizované aplikaci - případně zneužití by neohrozilo celého hosta. Při nějaké chybě a selhání aplikace supervizor provede automatický restart.

Vytvoření aplikace a přidávání jednotlivých zařízení probíhá pomocí webového rozhraní. Pro vytvoření stačí zvolit cílovou platformu (RPi3, BeagleBone atd.) a jméno (ID) aplikace. Nasazení kódu je založeno na *git commitu*, tj. z lokálního repositáře se provede *push* na vzdálený přípojný bod, který byl vytvořen pro danou aplikaci. Podle cílové platformy se na serverech *resin.io*



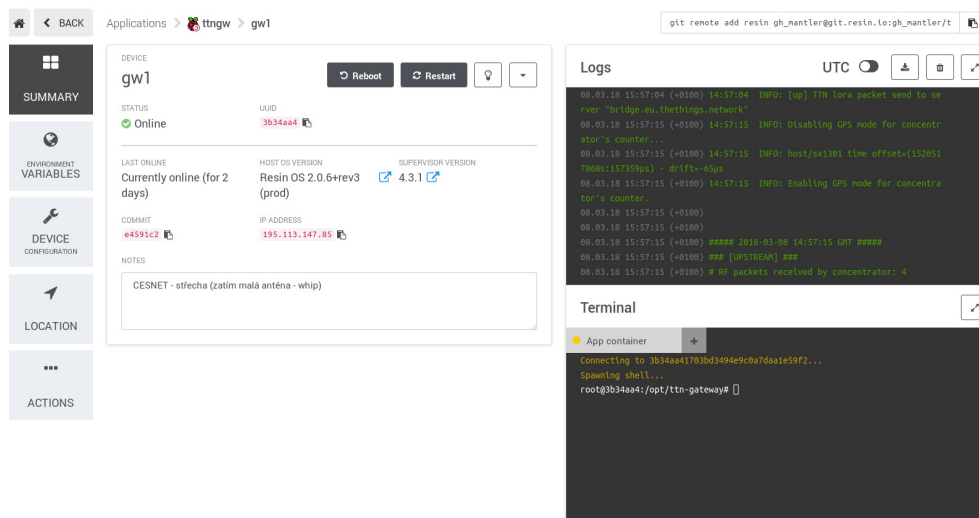
Obrázek 4.1: Testovací LoRaWAN brána

provede build aplikace, která je následně připravená jako Docker kontejner ke stažení do nového zařízení. To probíhá tak, že se stáhne OS image svázaný (přednastavený) s *resin.io* platformou (nastavený identifikátor, API klíče atp.) a zařízení se po startu pomocí šifrovaného spojení připojí k serverům projektu. Od vyšších verzí supervizoru je možné se připojit (přes webové rozhraní) jak do aplikačního kontejneru, tak do hostujícího systému. Součástí grafického rozhraní jednotlivých zařízení je výstup s logy (konkrétní URL lze změnit na veřejnou a implementovat např. hromadný monitoring). Příklad rozhraní u jedné z LoRaWAN bran je možné vidět na obrázku 4.2.

Samotná aplikace je nastavena pomocí Docker template vytvořeným [48]. Ten využívá tzv. Multi-protocol-packet-forwarder, tj. fork verzi původního packet forwarderu<sup>17</sup> od Semtechu. Ke své funkci potřebují ještě driver/HAL vrstvu pro samotný koncentrátor (*lora\_gateway* [49]). Multiprotokolová verze podporuje jak původní UDP (User Datagram Protocol) verzi forwarderu, tak novější verzi The Things Network (TTN) gateway-connector protokolu, která používá TCP (Transmission Control Protocol). TTN verze podporuje autentifikaci (používá se *GW Key* vygenerovaný při registraci brány), šifrování a na rozdíl od UDP verze je spolehlivá. Zprávy jsou serializovány pomocí gRPC, případně zabezpečenou TLS (Transport Layer Security) verzí MQTT (Message Queuing Telemetry Transport).

<sup>17</sup>Má na starosti pouze komunikaci se síťovým serverem a přeposílání zpráv nižší vrstvy.

## 4. POTŘEBNÁ INFRASTRUKTURA



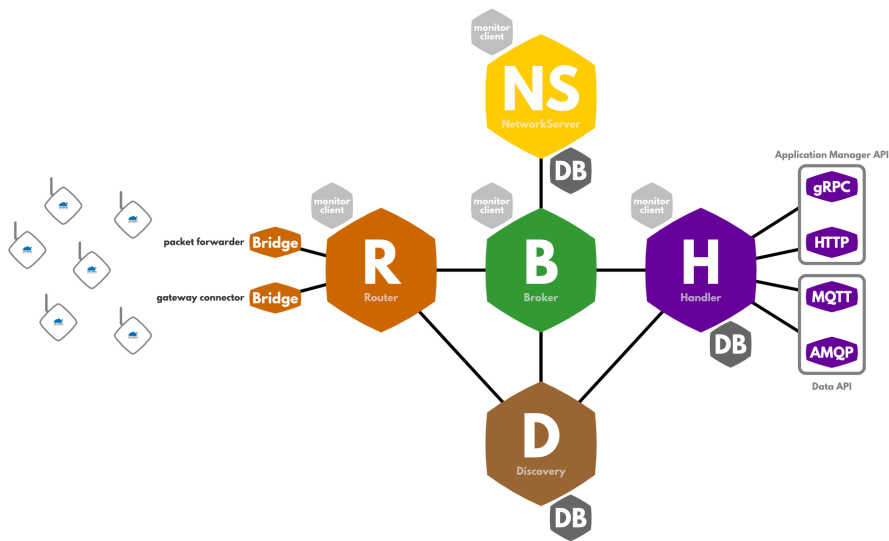
Obrázek 4.2: Resin.io základní pohled na spravované zařízení

### 4.2 Síťový server

V testovací topologii je využit open-source komunitní projekt provozovaný The Things Network [50] (TTN). V Evropě je značně rozšířený (cca 3 000 bran) a poskytuje IoT připojení senzorům skrze LoRaWAN síť - komukoliv a bez poplatku, stačí když je v oblasti instalována nějaká brána připojená (využívající) TTN síťový server. Tj. je možné připojit vlastní bránu nebo využívat brány ostatních uživatelů.

TTN architektura (viz 4.3) se skládá z těchto základních bloků.

- *Bridge* - zajišťuje kompatibilitu (překládá na interní protokol) mezi TTN a protokoly používanými na branách pro komunikaci s NS (síťový server). Těch může být obecně více, např. TTN vyvinul vlastní verzi tzv. *gateway connector*, který pracuje nad TCP (původní *packet forwarder* od Semtechu nad UDP).
- *Router* - směrování na základě DevAddr (konkrétně prefixu specifického pro jednotlivé operátory), plánování downlink paketů (která brána bude využita k odeslání - počítá tzv. skóre, podle síly signálu, dostupnosti atp.).
- *Broker* - provádí de-duplikaci paketů. Ty mohou být přijaty od více bran, což je důležité (žádoucí) např. pro účely této práce, tj. geolokaci. Data jsou ponechány (pouze jednou) a metadata jsou rozšířena o informace ze všech zúčastněných bran. Pro detekci takových paketů je využíváno MD5 hashe aplikačních dat (payload) ve specifickém časovém úseku (jednotky sekund). Měření provedené TTN ukazují, že se průměrná prodleva od prvního k poslednímu duplikovanému paketu pohybuje pod 100 ms a



Obrázek 4.3: The Things Network síťová architektura [50]

maximálně cca 300 ms (pravděpodobnost, že bude v tak krátkém úseku přijat odlišný paket se stejným hashem je velmi nízká). Dále provádí přiřazení paketu konkrétnímu zařízení (DevAddr nemusí být unikátní) v konkrétní aplikaci.

- *Handler* - samostatná komponenta, která uživateli umožňuje zvolit místo a způsob, kterým je paket dešifrován (na aplikační úrovni), poté je zpráva předána aplikaci. Data ve zprávě je často zapotřebí dekodovat (např. z binární podoby), případně provést konverzi. Handler tuto funkcionalitu nabízí pomocí tzv. payload funkcí. Využívá např. MQTT, HTTP nebo gRPC.
- *Network server* - aktualizuje stav zařízení a odesílá MAC příkazy (změna ADR atp.) ve vzorové šabloně downlink zprávy, která je předána handlerovi.
- *Discovery server* - TTN je distribuovaná síť a je zapotřebí zjistit kam odeslat data od jednotlivých aplikací a služeb.

V současné době je v projektu zaregistrováno necelých 3000 bran, které jsou k dispozici všem ostatním zájemcům/uživatelům.

Pro účely této práce byla vytvořena aplikace s identifikátorem *tnttest* pod kterou byl přidán testovací senzor (ID *tst\_tmp*) viz následující sekce. U nového zařízení je potřeba zvolit metodu aktivace (OTAA, ABP) a na základě této volby jsou buď automaticky vygenerovány další informace jako jsou relační klíče, nebo jsou nastaveny uživatelem (použity jsou klíče a identifikátory dodané se zařízením). Dále bylo zapotřebí zaregistrovat testovací brány ze sekce

4.1, které již využívají *gateway connector* protokol nad TCP (pro registraci a spárování se využívá uživatelsky zvolené Gateway ID nikoliv DevEUI jako u starší Semtech verze).

### 4.3 Testovací senzor

Testovací senzor slouží pro experimentální ověření výsledků této práce, tedy schopnosti lokalizovat. Dalším jeho účelem je mapování pokrytí, resp. síly signálů v oblastech vnitřní Prahy. Výsledkem je mapa, která graficky znázorňuje jednotlivé brány a lokace, z kterých tato brána přijala nějaký paket - ten je barevně odlišen podle hodnoty síly signálu RSSI.

Ze zmíněných plynou následující požadavky na senzor:

- *Přenosné provedení* - neboli malá velikost, aby bylo možné se senzorem obcházet Prahu, případně senzor připevnit např. na kolo. S tím souvisí použití akumulátoru/baterie s dostatečnou kapacitou (aby při terénním testu vydržela).
- *Externí zdroj určení pozice* - poskytne referenční hodnotu pro vyhodnocení přesnosti.
- *Externí spínač* - odeslání paketu na základě stisku tlačítka, přepnutí spínače.
- *Změna vysílacího režimu* - změna mezi režimem periodickým (např. při vyhodnocení přesnosti na základě více odeslaných paketů) a neperiodickým (externího spuštění, resp. odeslání na základě stisku tlačítka).
- *Vizuální indikace stavu* - znázornit kdy je paket odeslán, zdali bylo odeslání úspěšné, změna režimu.

Pro řídicí část byla zvolena Waspnote platforma osazená komunikačním LoRaWAN (Microchip RN2483) modulem a GNSS/GPS. Použitý akumulátor disponuje kapacitou 6600mA/h. Tlačítko je panelové a uchycené skrze vnější kryt senzoru. Jedná se o spínací ON-OFF typ bez aretace. Sestavený senzor lze vidět na obrázku 4.5.

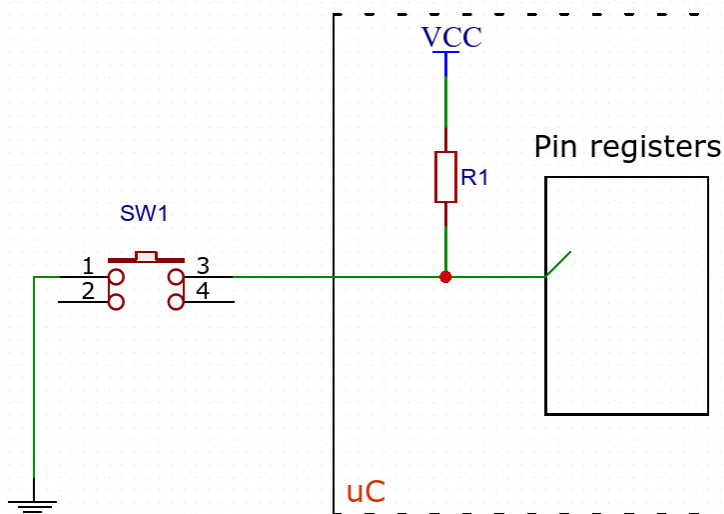
#### 4.3.1 Programové řešení

Waspnote (jeho mikrokontrolér) obsahuje interní pull-up rezistory, které lze aktivovat následujícím nastavením pinu (není v dokumentaci) [51]:

---

```
pinMode(DIGITAL3, INPUT);  
digitalWrite(DIGITAL3, HIGH);
```

---



Obrázek 4.4: Připojení tlačítka - interní pull-up rezistor

Tlačítko (SW1) je tedy možné připojit přímo na zem (viz schéma 4.4) - z libovolného digitálního vstupu/výstupu.

Program ve smyčce čeká až bude tlačítko stisknuto (zahrnut je mechanismus proti zpětnému rázu) a zjistí zdali byl nalezen (GNSS/GPS) fix (poloha) - pokud ano, odešla informace o poloze (zeměpisná šířka, délka a výška) pomocí LoRaWAN modulu. Odesílán je také HDOP (Horizontal Dilution of Precision) údaj, který je využíván při mapování pokrytí, resp. síly signálu po Praze viz sekce 7.1. Vnitřní smyčka kontroluje zdali bylo tlačítko zmáčknuto vícekrát, a pokud ano, změní aktuální režim. Režimy jsou pouze dva (periodicky a neperiodický), přepnutí je tedy možné provést pomocí stejného vzorce - dvojitý stisk tlačítka v předem definovaném časovém okénku (např. 2 sekundy). Jednotlivé stavy a úspěšnost operací jsou světelně - k dispozici jsou 2 LED diody, a to zelená a červená - indikovány následujícím způsobem:

- *Blokující stav* - konstantně svítící červená LED; indikuje stavy ve kterých nelze používat tlačítko (čekání na GNSS polohu, odesílání paketu).
- *Změna režimu* - výchozí je režim normální, tedy neperiodický. Přechod je signalizován rozsvícením obou LED (červené i zelené) po dobu 2 sekund a jedním, resp. dvojitým bliknutím při přechodu do neperiodického, resp. periodického módu.
- *Úspěšné/Neúspěšné nalezení polohy (fix)* - čtveré bliknutí zelené, resp. červené LED diody.

#### 4. POTŘEBNÁ INFRASTRUKTURA

---

- *Odesláni zprávy* - úspěšnost je signalizována dvojitým probliknutím obou LED v krátkém intervalu.

Souřadnice jsou zakódovány tak, aby se minimalizoval počet přenášených bajtů - latitude (3 B), longitude (3 B), altitude (2 B), HDOP (1 B). Zeměpisná šířka a délka jsou převedeny na float hodnotu (stupně v desetinné podobě). Nejprve je nutné zajistit, aby hodnoty nebyly záporné, tj. k šířce a délce přičíst hodnotu 90, resp. 180. Pro potřeby této práce - a očekávané přesnosti - stačí uchovávat čísla na 4 desetinná místa (přesnější hodnota uložená ve 3 B by vyžadovala složitější operace dělení a zpětně násobení při kterých dochází k zaokrouhlování a nepřesnostem) - to odpovídá cca jednotkám až desítkám metrů. Hodnoty jsou tedy roznásobeny 10000 a uloženy v uint32\_t proměnných. Pomocí binárního posunu a součinu je 24 LSB (Least Significant Bit) bitů (hodnoty nikdy nebudou větší), vždy po jednom bajtu, uloženo do unsigned char zásobníku *data*. Podobně je převedena nadmořská výška a HDOP (udáván v metrech, tj. stačí jej zachovat na jedno desetinné místo). Posledním krokem je převedení na HEX čísla, která jsou následně předána LoRaWAN funkci k odeslání.

```
uint32_t lat = (latitude + 90) * 10000;
data[0] = (lat >> 16) & 0xFF;
data[1] = (lat >> 8) & 0xFF;
data[2] = lat & 0xFF;
uint32_t lon = (longitude + 180) * 10000;
data[3] = (lon >> 16) & 0xFF;
data[4] = (lon >> 8) & 0xFF;
data[5] = lon & 0xFF;

data[6] = (altitude >> 8) & 0xFF;
data[7] = altitude & 0xFF;
data[8] = accuracy
```

Pro dekodování je použita integrační funkce na síťovém TTN serveru viz následující JavaScript kód

```
1 function Decoder(bytes, port) {
2   var decoded = {};
3   decoded.lat = ((bytes[0]<<16)) + ((bytes[1]<<8)) + bytes[2];
4   decoded.lat = (decoded.lat / 10000) - 90;
5   decoded.lat = Math.round( decoded.lat * 1e4 ) / 1e4;
6   decoded.lon = ((bytes[3]<<16)>>>0)+((bytes[4]<<8)>>>0)+bytes[5];
7   decoded.lon = ((decoded.lon / 10000) - 180);
8   decoded.lon = Math.round( decoded.lon * 1e4 ) / 1e4;
9   decoded.alt = (bytes[6]<<8) + bytes[7];
10  decoded.hdop = bytes[8] / 10;
11  return decoded; },
```



který hodnoty dekóduje a uloží do JSON struktury pod klíčem *payload\_fields*

```
"payload_fields": {  
  "alt": 274,  
  "hdop": 1.1,  
  "lat": 50.0851,  
  "lon": 14.4601  
}
```

Operace na řádce číslo 5 a 8 slouží pro korektní zaokrouhlení na 4 desetinná místa.

### 4.3.2 Konstrukce uzavíracího boxu

Pro jednotlivé díly senzoru (tlačítko, externí baterie, GNSS anténa, samotný modul) bylo zapotřebí vytvořit uzavíratelnou krabičku a konstrukční prvky, které všechno zafixují tak, aby bylo snadné senzor používat.

Univerzální krabička nenabízí optimální uchycení a stejně by vyžadovala další úpravy. Z toho důvodu byla zvolena cesta vlastní výroby - vytištění konstrukčních dílů na 3D tiskárně, konkrétně na české dvoutryskové Poseidon DUO [52].

Při tisku byly použity celkem tři druhy materiálu (struny neboli filament), a to ABS, PETG a PLA (dle vlastností). ABS je pevný a odolný materiál, ale trpí horší adhezí a při větších dílech má tendenci kroužit/deformovat se (jednotlivé vrstvy mají odlišnou teplotu a rychlost tuhnutí). To byl problém převážně u vnější částí boxu, která uchovává celý senzor a je rozměrově největší. PETG tento problém vyřešil, ale zase nevykazoval tak dobré výsledky při tisku „do vzduchu“ - např. tisk zarážek či průchodů a otvorů. Konečnou úpravou bylo vyhlazení zasouvacího krytu pomocí acetonu (lze aplikovat pouze na ABS), konkrétně vy výparech koncentrovaných ve skleněné nádobě podlité vodou (aby byla vodotěsná).

Celý box se skládá celkem z 6 plastových dílů (všechny lze vidět v příloze C), navrhovaných s důrazem na modulárnost (tj. možnost opakovaného složení):

- *Vanička pro externí baterii* - samostatný díl, aby byla baterie zafixována a nepohybovala se (doplněna plastovou pěnou), pro tisk byl použit ABS materiál - lépe se tiskly vnitřní hrany na kterých je usazena vrchní část (úchyt Waspote modulu).
- *Hlavní úchyt* - díl s vytištěnými sloupky na kterých je nasunut Waspote modul s připevněným GPS a LoRaWAN modulem, také disponuje podpurnými sloupky, aby se celá deska nekývala; tento díl je nasunut na spodní vaničku s baterií.



Obrázek 4.5: Sestavený testovací senzor v boxu z 3D tiskárny

- *Držák GNSS antény* - zafixování v horní části Waspote modulu, jsou v něm nataveny kolíčky, aby mohl být zasunut do pinů modulu.
- *Vnější box* - krabička v které jsou umístěny ostatní díly. Disponuje otvory pro připojení 868 MHz antény k LoRaWAN rádiovému modulu, přístupem k on/off přepínači a USB portu. Také má částečně vykrojený otvor v prostorech LED diod, do kterého je vlepeno průhledné sklíčko.
- *Horní kryt* - uzavírá (nasunutím) celou krabičku ze shora a částečně fixuje anténu, aby nevypadla ze svého ukotvení.
- *Sklíčko* - průhledný díl vytištěn z PLA, slouží k vizuální kontrole funkcí senzoru (LED diody).

Díly byly modelovány, a následně konvertovány do formátu STL, v programu SolidWorks [53]. Kontrola vyexportovaného modelu probíhala za pomoci Netfabb [54] a samotné generování (slicing) řídicího kódu pro 3D tiskárnu (G-code) v programech Simplify3D [55] a Slic3r [56].

Zkompletovaný senzor lze vidět na obr. 4.5, podrobněji v příloze C.

## Návrh lokalizačního řešení

### 5.1 Výpočet polohy

Princip výpočtu založený na technikách TDoA a multilaterace byl představen v sekci o geolokaci v LoRa sítích 3.2. Tento přístup je využit i v této práci.

Poloha jednotlivých bran je známa díky přítomnosti GNSS modulu; konkrétně se jedná o geodetické (zeměpisné) souřadnice, které je zapotřebí převést na kartézské (UTM a ECEF pro 2D, resp. 3D). Matematický zápis problému (v 2D prostoru pro názornost) je následující

$$r_i^2 = (x_i - x)^2 + (y_i - y)^2 = K_i - 2x_i x - 2y_i y + x_i^2 + y_i^2, \quad (1)$$

kde

$$K_i = x_i^2 + y_i^2.$$

Pak pro pseudovzdálenosti, tedy rozdíl vzdáleností od referenční brány (index roven jedné) a  $i$  brány platí ( $c$  značí rychlost světla cca 300 000 km/s)

$$r_{i,1} = cd_{i,1} = r_i - r_1, \quad (2)$$

pro  $i \in \hat{n}$ , kde

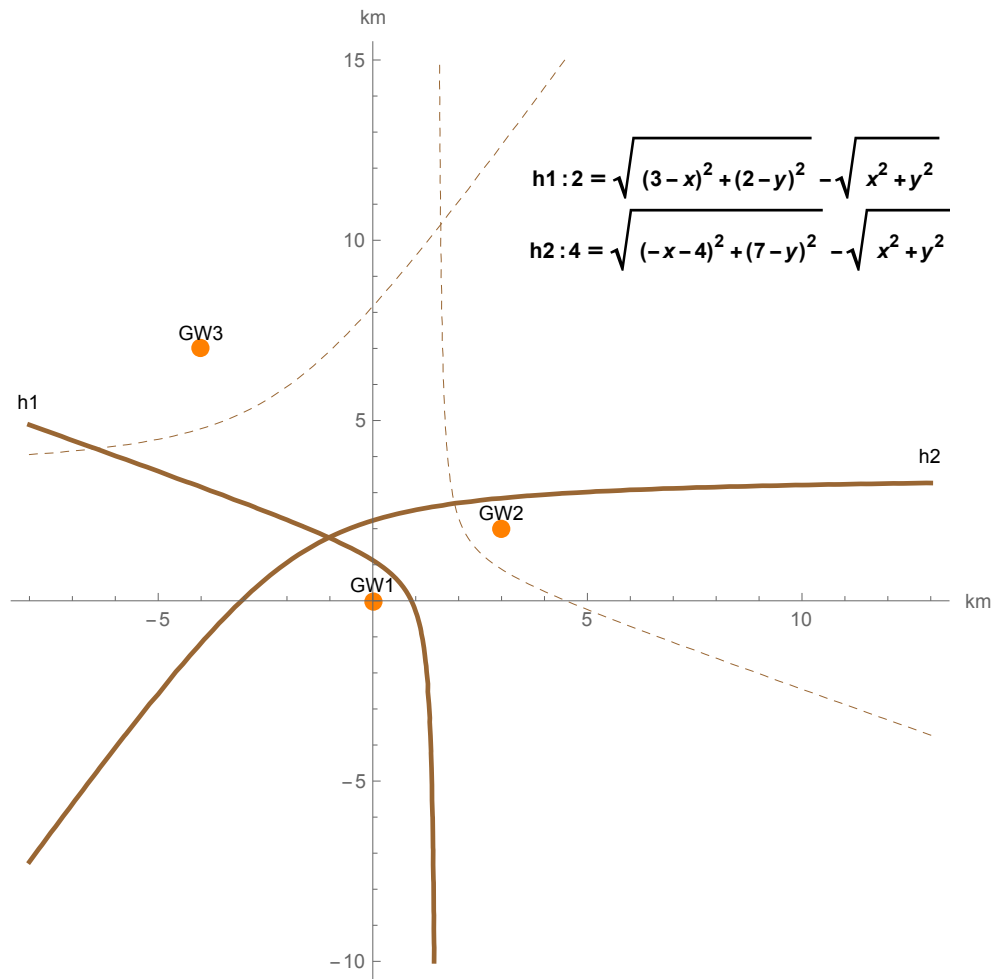
$n$	= počet bran (číslování od 1),
$r_i$	= vzdálenost senzoru od $i$ brány,
$d_{i,1}$	= TDoA spočten z referenčního času,
$(x_i, y_i)$	= souřadnice $i$ brány v kartézské soustavě,
$(x, y)$	= hledaná poloha senzoru.

Referenční brána, resp. čas je ten s nejnižší hodnotou. Tedy brána, která přijala zprávu jako první.

Na obr. 5.1 je vyobrazena vzorová situace s branami o souřadnicích

$$GW1 = (0, 0), GW2 = (3, 2), GW3 = (-4, 7)$$

## 5. NÁVRH LOKALIZAČNÍHO ŘEŠENÍ



Obrázek 5.1: LoRa lokalizace a nalezení řešení pomocí pseudovzdálenosti (geometrická interpretace hyperbola)

Referenční brána (GW1) je umístěna do počátku soustavy což zjednodušuje část rovnic  $h_1$  a  $h_2$ , jelikož souřadnice referenční brány s indexem 1 jsou rovny 0. Geometrická interpretace je patrná z vykreslení jednotlivých rozdílů vzdáleností. Jedná se o hyperboly (s jednou větví), jejichž průsečík představuje polohu senzoru (proto někdy nazývána jako hyperbolická lokalizace). Levá strana určuje o kolik je jedna z bran blíže či dále (v jednotkách vzdálenosti což odpovídá korespondující hodnotě TDoA roznásobené rychlostí světla viz vztah (2) od senzoru. Čárkované křivky vyznačují druhou větev každé hyperboly (odpovídají rovnicím s opačným znaménkem)

$$2 = -\sqrt{((3-x)^2 + (2-y)^2)} + \sqrt{(x^2 + y^2)} \quad (3)$$

$$4 = -\sqrt{((-4-x)^2 + (7-y)^2)} + \sqrt{(x^2 + y^2)} \quad (4)$$

Celá hyperbola (obě větve) je vykreslena při uzavření pravé strany rovnice do absolutní hodnoty.

Vyřešení nelineární soustavy rovnic (2) skýtá hledané řešení. V závislosti na počtu bran, které zprávu od senzoru přijmou mohou nastat následující situace:

1. *Nedostatek rovnic pro výpočet* - zpráva byla přijata méně než 3 branami, tj. je možné sestavit pouze jednu nebo žádnou rovnici (nelze vyřešit pro 2 neznámé).
2. *Právě 2 rovnice* - zprávu přijali právě 3 branami, jsou získány 2 TDoA oproti referenčnímu času a lze vypočítat polohu v 2D rovině.
3. *Právě 3 rovnice* - podobně jako v předchozím bodu pro 4 brány a výpočet 3 neznámých, tedy 3D pozice.
4. *Přeuročená soustava rovnic* - zpráva byla přijata více než 4 branami (v případě 2D alespoň 4), tj. soustava rovnic je tzv. přeuročená („overdetermined“). Algebraicky přesné řešení - které by vyhovovalo všem podmínkám/rovnícím - neexistuje. Hyperboly se v rovině, resp. prostoru neprotnou v jednom bodě (nepřesný čas, odrazy). Je nutné provádět algebraickou aproximaci, jenž se nejvíce blíží jednotlivým rovnicím (tedy bod, který je nejbližší všem křivkám hyperbol či plochám hyperboloidů). Další možností je využít vždy jen určitou podmnožinu bran (naměřených hodnot), v 2D prostoru to jsou 3 brány, a následně provést výpočet pro všechny možné kombinace. Výsledky se porovnávají, tj. spočítá se vzájemná korelace. Pokud jsou některé výrazně odlišné a zároveň je v nich využita stejná brána, provede se jejich vyřazení.

Soustavu je možné řešit dvěma způsoby (obecně je nutné provést linearizaci, tj. převést nelineární rovnice na lineární a řešit standardní soustavu lineárních rovnic):

- *Neiterativně* - explicitní vyjádření neznámých vhodnou úpravou či aproximací jednotlivých rovnic. Výhodou je zpravidla nízká náročnost na výpočetní výkon a dopředu známý čas výpočtu. Nevýhodou je možná akumulace nepřesností v jednotlivých krocích algoritmu - ty se mohou akumulovat a ve výsledku představovat nezanedbatelnou chybu (jednotlivé chyby se stejně tak mohou částečně „vynulovat“ a v konečném důsledku vykompenzovat a podat přesné řešení). Pokud jsou rovnice sestaveny ze silně zkreslených/nepřesných dat, nemusí řešení existovat (v oboru reálných čísel), nebo naopak mohou být dvě (hyperboly se protnou ve dvou místech). Poté je zapotřebí určit, které z těchto řešení je to správné (podrobněji v kapitole o implementaci).

- *Iterativně* - nejdříve se provede prvotní odhad pozice, který je následně, v konečném počtu kroků, vylepšován. Kritériem k zastavení může být dosažení dostatečně přesného řešení (rozdíl aktuální a předchozí pozice je menší než předem daná konstanta - předpoklad, že nedojde k výraznému zlepšení) nebo např. přesažení maximálního počtu iteračních kroků (řešení nekonverguje, neakceptovatelná doba výpočtu). Výhodou je dosažení přesného řešení, možnost alespoň aproximovat hledanou pozici i při velkých nepřesnostech či odrazech. Nevýhodou je výpočetní a časová náročnost - při velkém počtu senzorů a četné komunikaci je zapotřebí dostatečně dimenzovaná a vysoce škálovatelná infrastruktura, aby nedocházelo k zahazování požadavků či nepřiměřenému zpoždění. Důležitým krokem je volba prvotního odhadu, který má vliv na další vývoj algoritmu, tj. konvergenci.

Pro účely této práce byla zvolena jak neiterativní, tak iterativní verze algoritmu. Důvodem je možnost porovnání a vzájemného doplnění obou přístupů (neiterativní pro odhad počáteční pozice a iterativní pro zpřesnění nebo alespoň aproximace řešení, pokud přesný průnik neexistuje). Konkrétně byl zvolen, přizpůsoben a implementován Chanův (který může dosahovat přesností iterativních přístupů [57]) a LS algoritmus založen na Taylorově řadě (linearizace) [58].

**Chanova metoda** Rovnice (2) je nejdříve nutné transformovat na soustavu lineárních rovnic. Z (2) je možné vyjádřit  $r_i^2 = (r_{i,1} + r_1)^2$ , tak že (1) může být přepsána jako

$$r_{i,1} + 2r_{i,1}r_1 + r_1^2 = K_i - 2x_i x - 2y_i y + x^2 + y^2. \quad (5)$$

Odečtením (1) pro  $i = 1$  od (5) je získán následující vztah

$$r_{i,1} + 2r_{i,1}r_1 = -2x_{i,1}x - 2y_{i,1}y + K_i - K_1, \quad (6)$$

kde  $x_{i,1}$  a  $y_{i,1}$  odpovídají  $x_i - x_1$ , resp.  $y_i - y_1$ . Vzniklá rovnice je již lineární (s neznámými  $x, y$  a  $r_1$ ). Chan navrhl následující postup (pro 3 brány, obdobný princip je v 3D aplikován na 4 brány). Ze vztahu (6) lze pomocí  $r_1$  vyjádřit  $x$  a  $y$  následujícím způsobem

$$\begin{bmatrix} x \\ y \end{bmatrix} = - \begin{bmatrix} x_{2,1} & y_{2,1} \\ x_{3,1} & y_{3,1} \end{bmatrix}^{-1} \times \left( \begin{bmatrix} r_{2,1} \\ r_{3,1} \end{bmatrix} r_1 + \frac{1}{2} \begin{bmatrix} r_{2,1}^2 - K_2 + K_1 \\ r_{3,1}^2 - K_3 + K_1 \end{bmatrix} \right) \quad (5)$$

Po substituci proměnných - neznámých souřadnic  $x$  a  $y$  - do (1) pro  $i = 1$  vznikne kvadratická rovnice s  $r_1$ . Následně je kladný kořen dosazen zpět do (5); tento krok eliminuje poslední neznámou a je možné vypočítat hledanou pozici.

V případě alespoň 4 a 5 bran v 2D, resp. 3D prostoru je systém přeurčen, a proto je použit odlišný přístup - metodou nejlepší shody (aproximace pomocí dvojnásobného použití váženého LS algoritmu viz [57]).

**Taylorův rozvoj a LS** Tento přístup využívá Taylorovu řadu a rozvoje s polynomy řádu menšího než dva (lineární aproximace, bod je nahrazen tečnou). Algoritmus začíná s počátečním odhadem pozice  $(x_0, y_0)$ , který je následně vylepšován iterativním procesem - v každém kroku proběhne nalezení lokálního LS řešení a následně je spočtena deviace

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = (G^T Q^{-1} G)^{-1} G^T G^{-1} h, \quad (6)$$

kteřá je následně přičtena k  $x_0$  a  $y_0$ .  $Q$  je kovarianční matice naměřených TDoA (tj. čtvercová velikosti  $n - 1$ ),  $G$  a  $h$  jsou definovány následovně

$$h = \begin{bmatrix} r_{2,1} - (r_2 - r_1) \\ r_{3,1} - (r_3 - r_1) \\ \vdots \\ r_{n,1} - (r_n - r_1) \end{bmatrix} \quad (8)$$

$$G = \begin{bmatrix} (x_1 - x)/r_1 - (x_2 - x)/r_2 & (y_1 - y)/r_1 - (y_2 - y)/r_2 \\ (x_1 - x)/r_1 - (x_3 - x)/r_3 & (y_1 - y)/r_1 - (y_3 - y)/r_3 \\ \vdots & \vdots \\ (x_1 - x)/r_1 - (x_n - x)/r_n & (y_1 - y)/r_1 - (y_n - y)/r_n \end{bmatrix}$$

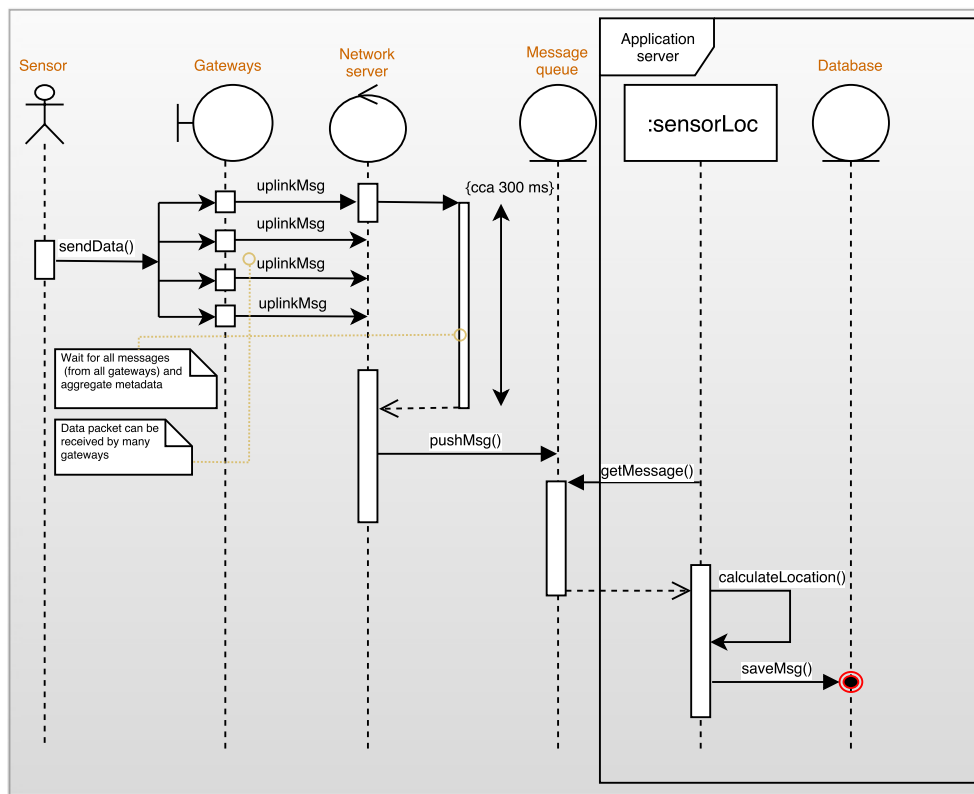
Neznámé  $(x, y$  a  $r_1, \dots, R_n)$  jsou spočteny za předpokladu  $x = x_0$  a  $y = y_0$ .

### 5.1.1 Průběh celého procesu lokalizace

Návrh průběhu lokalizačního procesu lze vidět na obr. 5.2. Zpráva vyslaná senzorem může být přijata 0 až  $x$  branami (předpoklad jsou alespoň 3 - pro účely lokalizace), které zprávu - po obohacení o metadata a příslušné kontrole - přepošlou na síťový server. Ten po přijetí první zprávy spustí časové okénko (předem dané, konstantní) a vyčká, zdali budou přijaty nějaké další zprávy. Následně všechny přijaté zprávy zkontroluje, že opravdu patří k sobě (na základě hashe aplikačních dat) a vytvoří novou agregovanou datovou strukturu, jež obsahuje všechna metadata<sup>18</sup> od jednotlivých bran. Výsledná struktura je uložena do fronty zpráv (pod identifikátorem určeným aplikací a konkrétním senzorem), kde je následně vyzvednuta lokalizační aplikací, běžící na aplikačním serveru. Aplikace zkontroluje a vyfiltruje (podle potřeby) přijaté informace na základě kterých vypočítá odhad pozice senzoru. Tento odhad je přidán k stávající zprávě (spolu s volitelnými údaji jako je vypočtená chyba / odhad chyby) a uložen do databáze. Původní data jsou ve zprávě ponechána, např. pro budoucí porovnání s jinými algoritmy nebo dokumentační/záznamové účely.

<sup>18</sup>Důležitá je informace o čase přijetí a pozice brány, případně identifikační údaj pro statické namapování na polohu či další statistické/monitorovací účely. RSSI a SNR jsou podstatné při hrubých odhadech polohy nebo estimaci kvality časové značky (odrazy atp.).

## 5. NÁVRH LOKALIZAČNÍHO ŘEŠENÍ



Obrázek 5.2: Sekvenční diagram celého procesu lokalizace a zúčastněné subjektů

### 5.1.2 Aplikační řešení sensorLoc

Lokalizační řešení - *sensorLoc* - je logicky rozděleno na dvě části

- *komunikační a řídicí část* - získává data od síťové serveru, iniciuje výpočet a ukládá výsledky do databáze,
- *výpočetní část* - implementuje samotné lokalizační algoritmy, konverzi mezi různými typy souřadnic, výpočet odchylek a chyb.

## 5.2 Úložiště

Jednotlivé zprávy přijaté od síťového serveru je možné uložit do DB okamžitě po přijetí, nebo až po provedení výpočtu a vytvoření výsledné struktury s informací o poloze. Na způsob využití DB lze nahlížet dvěma způsoby:

1. *Dočasné úložiště* - slouží pouze pro „odložení“ metadat přijatých od serveru. Ve chvíli, kdy je poloha zapotřebí jsou provedeny výpočty (nebo se může čekat na dodatečné či rozšiřující informace) a stávající struktura



je o polohu obohacena. Případně můžou výpočty probíhat periodicky. Vhodné pokud je výpočet prováděn pro značný počet senzorů - nebylo by možné provádět výpočet (může být náročný) ještě před samotným uložením, jelikož by docházelo k přetížení.

2. *Konečné úložiště* - hned po přijetí zprávy od síťového serveru je proveden výpočet a vytvořena nová struktura, která je doplněna o geolokaci a následně uložena do DB.

Pro účely této práce byla zvolena druhá varianta a v DB se rovnou uloží vypočtená poloha (ohodnocení a možnosti škálovatelnosti nejsou předmětem této práce). Formát této struktury byl zvolen následovně:

```
{
  "_id": "ObjectId()",
  "time": "ISODate()",
  "calculated": {
    "alt": "float",
    "error": "integer",
    "lat": "float",
    "lon": "float",
    "status": "string"
  },
  "gws": [
    {
      "alt": "float",
      "gtw_id": "string",
      "lat": "float",
      "lon": "float",
      "rssi": "integer",
      "snr": "float",
      "time": "float"
    },
    {
      "The rest of gws objects"
    }
  ],
  "reference": {
    "alt": "float",
    "hdop": "float",
    "lat": "float",
    "lon": "float"
  }
}
```

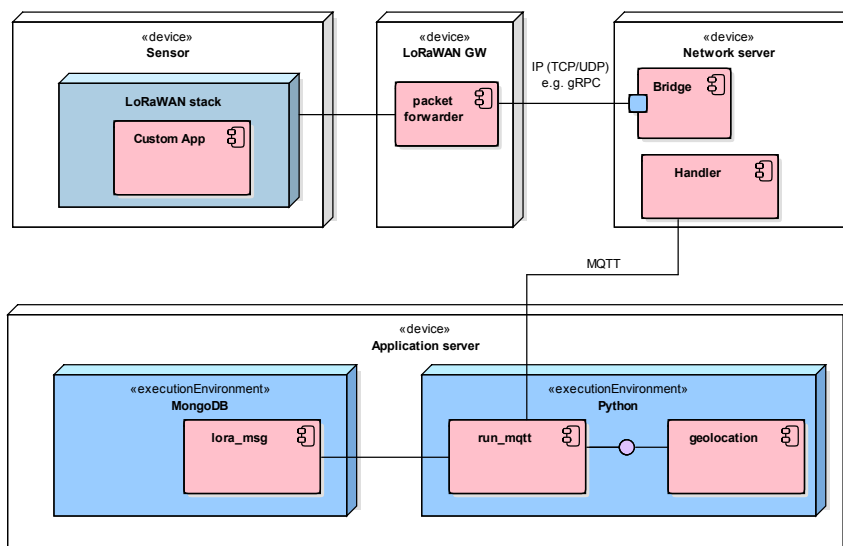
## 5. NÁVRH LOKALIZAČNÍHO ŘEŠENÍ

Tato data obsahují 3 základní objekty:

- *calculated* - vypočtená poloha senzoru, tj. zeměpisná šířka, délka a výška. Součástí je také informace o chybě (v metrech), která je spočtena na základě referenční informace o poloze. Absolutní hodnota o chybě je důležitá při vyhodnocení výsledků. *Status* představuje dodatečnou informaci o výpočtu, možné hodnoty:
  - „*2D*“ - byla vypočtena 2D poloha (šířka a délka), typicky pokud je paket přijat pouze 3 branami,
  - „*3D*“ - paket přijat více jak 4 branami a byl použit 3D algoritmus,
  - „*NA*“ - „not applicable“, tj. nedostatečný počet bran (nedostatek metadat, minimem jsou 3),
  - „*error*“ - výpočet selhal, např. špatná geometrie umístění bran, příliš velké odchylky.
- *reference* - referenční údaje o poloze, které byly z testovacího senzoru odeslány za pomoci GNSS modulu.
- *gws* - pole, které sdružuje objekty s metadaty z jednotlivých bran. Konkrétně jejich polohu, sílu signál RSSI, SNR, jednoznačný název brány a čas přijetí paketu.

### 5.3 Spolupráce všech komponent

Fyzické umístění komponent a jejich propojení lze vidět na obr. 5.3.



Obrázek 5.3: Diagram nasazení - rozložení komponent a propojení

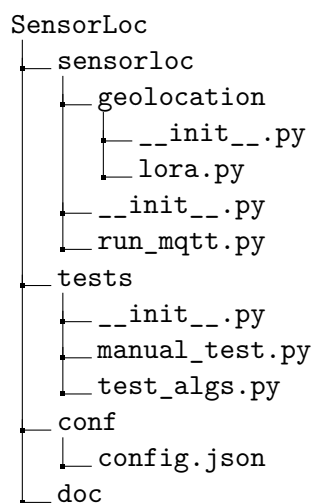
# Implementace

Pro implementaci byl zvolen programovací jazyk Python, a to z důvodu dostupnosti matematických knihoven, které usnadňují tensorové operace či symbolické proměnné pro řešení rovnic (přehlednější zápis a interpretace). Mezi externí knihovny patří převážně *numpy*, *scipy*, *sympy*, *utm*, *pymongo* nebo *paho-mqtt* užívaný pro sběr dat.

## 6.1 SensorLoc aplikace

### 6.1.1 Struktura

Struktura python balíčku je navržena tak, aby bylo snadné přidat další metody a lokalizační algoritmy, a proto jsou umístěny v samostatném pod-balíčku *geolocation*. Rozšíření o další protokol, jimž jsou data získávána, resp. přebírána od síťového serveru, je přímočaré. Stačí přidat nový modul do hlavního balíčku *sensorloc*. Základní uspořádání je následující:



```
|  
├─ sensorloc.py  
├─ setup.py  
├─ requirements.txt  
├─ MANIFEST.txt  
├─ LICENSE.txt  
└─ README
```

Start SensorLoc programu probíhá pomocí startovacího skriptu `sensorloc.py`, který importuje `main()` funkci `run_mqtt` modulu, případně po instalaci - za využití `setup.py` inicializačního skriptu - příkazem `sensorloc` (zaváděcí program je nainstalován do `PATH` proměnné).

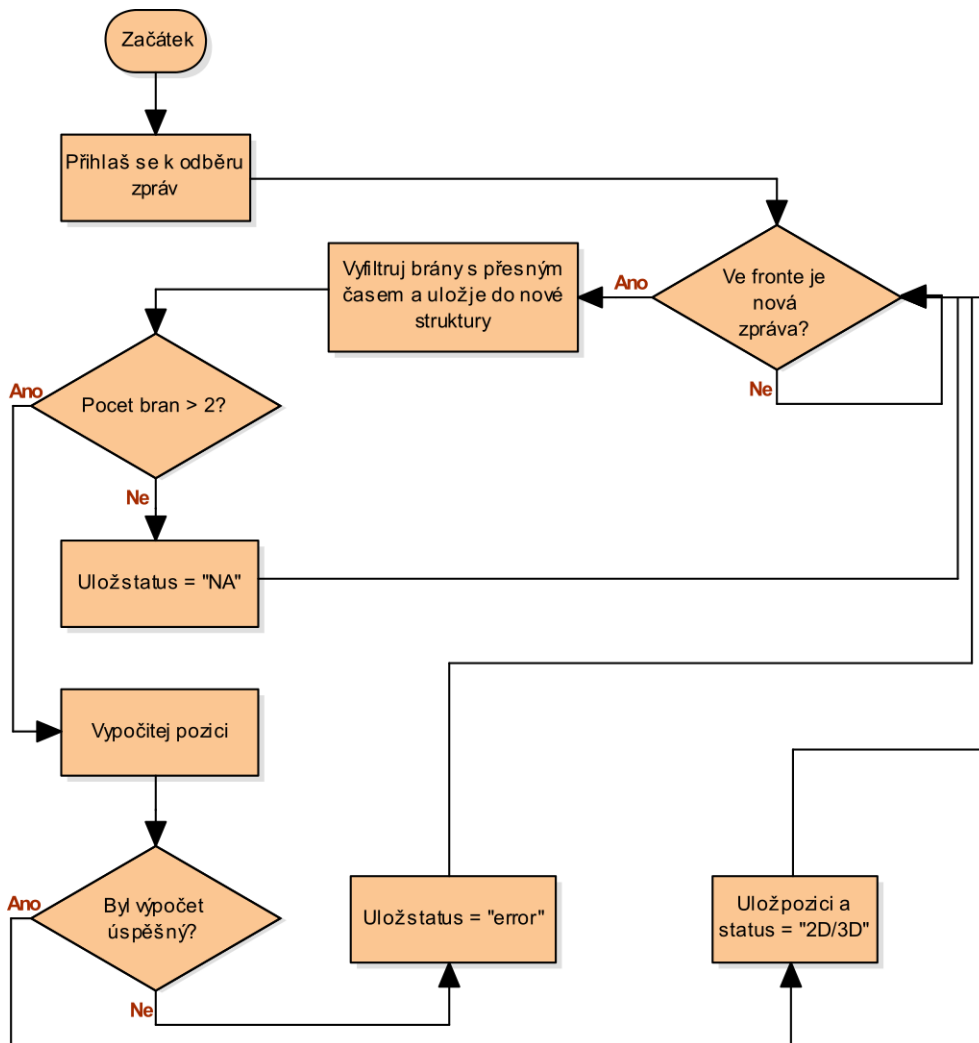
### 6.1.2 Komunikační modul

Implementován `run_mqtt` modulem. Přidání dalších/alternativních způsobů (pro získání datových zpráv od senzorů) je přímočaré, stačí přidat modul ve tvaru `run_xxx`, kde `xxx` může být např. `http` či `db` (`mysql`, `mongo`), pokud jsou data nejprve uložena (před samotným výpočtem).

Nejprve proběhne nastavení potřebných údajů pro MQTT spojení a následně jsou definovány speciální (tzv. callback) funkce `on_connect` a `on_message`, které jsou volány při přihlášení k danému serveru a tématu, resp. při přijetí nějaké zprávy. Kostra `on_message` funkce je následující

```
def on_message(client, userdata, msg):  
    msg.payload = msg.payload.decode("utf-8")  
    data = json.loads(str(msg.payload))  
    if msg.topic == "ttntest/devices/tst_tmp/up":  
        new_data = prepare_data_ttn(data)  
        new_json = append_sensor_loc(new_data)  
        new_json["time"] = dateutil.parser.parse(new_json["time"])  
        new_msg_s = json.dumps(new_json)  
        msgs.insert_one(new_json)
```

Data jsou nejprve dekodována a uložena do Python objektu (vnitřní reprezentace, tj. slovník). Jsou-li data od správného zařízení (v ukázce je ID rovno `tst_tmp`), provede se úprava JSON struktury a jednotlivých položek na očekávanou, předem definovanou, podobu - funkce `prepare_data_ttn` (při odběru z více zdrojů s mírně odlišnou strukturou je přirozeným postupem přidání rozcestníku a odpovídajících funkcí `prepare_data_xxx`). Dalším krokem už je samotné obohacení o poziční informaci. Tj. provede se filtrace bran s přesným časem a je zavolána lokalizační funkce viz následující podsekcce. Diagram průběhu a klíčová rozhodnutí viz obr. 6.1.



Obrázek 6.1: Průběh výpočtu z aplikačního pohledu

### 6.1.3 Výpočetní modul

Obsahuje samotné algoritmy, výpočetní funkce (konverze LLA na ECEF, odchylky v prostoru) a potřebné konstanty (WGS84 popis země atp.). Hlavní výpočetní funkcí, resp. rozcestníkem, je `get_location`, volaná s argumentem typu list. List struktura obsahuje Python tuply (uspořádané n-tice) v následujícím tvaru (lokality brány a čas přijetí paketu)

```
([latitude, longitude, altitude], timestamp),
```

tedy obsahuje tolik položek kolik je bran, které přijaly vyslaný paket. Podle počtu jsou volány jednotlivé algoritmy/funkce, jimiž jsou

- `get_loc_chan2d` a `get_loc_chan3d` - Chanův algoritmus pro nepřeurčenou soustavu rovnic, tj. 3, resp. 4 položky (přijímající brány),
- `get_loc_chan_overdet_2d` a `get_loc_chan_overdet_3d` - Chanův algoritmus pro přeúčenou soustavu rovnic, tj.  $> 3$ , resp.  $> 4$  položky,
- `get_loc_taylor_2d` a `get_loc_taylor_3d` - algoritmus využívající Taylorovu expanzi (linearizaci) a LS řešení soustavy rovnic; pro  $\geq 3$ , resp.  $\geq 4$  položky.

Obecným postupem - ve všech funkcích - je seřazení jednotlivých tuplů podle časové značky a zvolení první brány (obdržela zprávu první) jako referenční. dále jsou vypočteny TDoA vzhledem k první bráně a následně také pseudo-vzdálenosti (roznásobení rychlostí světla). Pracovní souřadnice jsou převedeny do UTM nebo ECEF a po provedení výpočtu (iterativní, neiterativní) jsou zpět převedeny na LLA souřadnice, jenž jsou předány návratovou hodnotu zpět volající funkci. Konkrétní příklad funkce implementující iterační lokalizační algoritmus:

```
def get_loc_taylor3d(gateways):
    COL = 3
    j = 0
    gws = []
    for loc, time in gateways:
        res = lla2ecef(*loc)
        gws.append([*res, time])
    sorted_gws = scipy.matrix(sorted(gws, key=lambda tstamp: tstamp[COL]))
    coord = sorted_gws[:, :COL]
    num = len(sorted_gws)
    # Start guess right at arithmetic mean of GW coordinates
    P = np.sum(np.asarray(coord), axis=0) / num
    while j < NITER:
        j += 1
        # Ranges (r1, ..., rM) to sensor position (guess)
        R =
        np.sqrt(((outer(ones(num), P) - coord) . A**2) . sum(axis=1)) . reshape(num, 1)
        R_rel = R[1:] - R[0]
        # pseudorange relative to reference gw
        Pr =
        ((np.array(sorted_gws[1:, COL]) - sorted_gws[0, COL]) * c) . reshape(num - 1, 1)
        h = scipy.matrix(Pr - R_rel)
        P_m = outer(ones(num - 1), P)
        G = ((np.full((num - 1, COL), coord[0]) - P_m) / R[0]) /
            - ((coord[1:] - P_m) / R[1:])
        # covariance matrix of TDoA's vector with respect to first/reference GW
```

```

Q = scipy.matrix((0.5 * scipy.identity(num - 1)) + 0.5)
G = scipy.matrix(G)
try:
    X = G.T*Q.I
    diff = (X*G).I * X * h
except:
    logger.error("Equations do not converge (bad geometry)")
    return LOCATION_ERROR
P += diff.A1
if not (abs(diff) > 1).any():
    break
return ecef2lla(P)

```

Běh algoritmu je omezen celkovým počtem iterací NITER (zkratka „number of iterations“) a dostatečně nízkou deviací (již nedochází k výraznému zlepšení, bylo dosaženo konvergence), která je přičítána k předchozí poloze. Počáteční odhad pozice je v ukázce počítán jako aritmetický průměr ze souřadnic jednotlivých bran, jenž paket přijaly. Dalším užívaným postupem je dosažení výsledku z Chanova neiterativního algoritmu (při omezení pouze na 3 brány).

## 6.2 Počáteční konfigurace a logy

Hlavní modul si při startu načte<sup>19</sup> konfigurační soubor, který slouží pro inicializaci jednotlivých parametrů MQTT připojení, umístění logu aj. Použit je - podobně jako u jednotlivých zprávy - JSON formát, který je dostatečně názorný (pro potřeby editace). Příklad testovací konfigurace:

```

{
  "username": "tntest",
  "password": "ttn-account-v2.q-PdG_Fm00UeJkameEekQUmVKM-mk5381Gb",
  "host": "eu.thethings.network",
  "port": 1883,
  "topic": "tntest/devices/+/up",
  "log_file": "./testlog.log"
}

```

Pro účely logování je použit standardní python balíček *logging*. Kořenový logger je definován v hlavním - komunikačním - modulu a ostatní moduly ho přebírají. Použity jsou dva handlers: jeden pro logování do souboru (který je definován vstupní konfigurací) a druhý do konzole (implicitně pouze důležitější zprávy; žádný debug).

<sup>19</sup>Prohledány jsou všechny standardní umístění - aktuální adresář, /etc/sensorloc, domácí adresář uživatele, proměnné prostředí a složky conf.

## 6. IMPLEMENTACE

---

Formát log zpráv je následující

```
FORMAT = "[%levelname)s:%(asctime)s:%(filename)s: \  
- %(funcName)s] %(message)s"
```



## Testování a měření

### 7.1 Mapování síly signálu

Mapování bylo provedeno pro představu, kde je ještě signál dostupný a v jakých místech je možné odeslat paket, tak aby byl přijat alespoň 3 branami (minimum pro určení pozice). Dále se hodí např. při porovnávání různých (typ a zisk) antén v reálném provozu. Nutností byla vizuální reprezentace a barevné odlišení dle úrovně hodnot RSSI - čistě textový záznam je téměř nepoužitelný, jelikož spojuje poziční informace ve formě zeměpisných souřadnic a číselné RSSI (velmi ztížená dedukce).

Proto byl využit projekt jednoho z členů TTN komunity *ttnmapper* [59], který umožňuje grafickou reprezentaci přijatých paketů a korespondujících bran. Projekt slouží primárně celé komunitě, aby měli všichni členové přehled o dostupném pokrytí, např. při plánování domácích projektů, tj. implicitně jsou informace z mapování nahrávány do globální mapy (přístupná hned na úvodní stránce projektu). Nicméně, pro potřeby testování či jiné specifické účely je možné data nahrávat do oddělených map, tzv. experimentů. Takový experiment byl použit i v případě této práce - informace jsou relevantnější, jelikož je použit pouze jeden senzor (stejná anténa - zisk, atp.) se staticky nastaveným DR.

Byly použity dvě metody integrace a nahrávání dat do *ttnmapper*, první zmíněna byla použita pouze zpočátku a následně nahrazena sestaveným senzorem, jenž disponuje GNSS modulem a snadněji se využívá.

1. *TTN Mapper aplikace* - možné nainstalovat na Android telefon s jakýmkoliv LoRaWAN senzorem/zařízením (nemusí disponovat zdrojem určení pozice), pouze je nutné ho zaregistrovat v aplikaci - název/ID aplikace a zařízení spolu s přístupovým klíčem. TTN Mapper se pomocí MQTT přihlásí k odběru těchto zpráv (publikování probíhá na TTN síťovém serveru) a pozici zjistí z daného Android telefonu (musí disponovat GNSS modulem), tj. existuje předpoklad, že je senzor v blízkosti

telefonu. Nevýhodou je nutnost používat telefon a udržovat předem danou přesnost pozice (HDOP), pokud hodnota stoupne jsou tyto pakety automaticky zahazovány. Tato metoda byla použita s RPi3 a LoraWAN modulem; odesílána byla statická data - jeden byte s volitelnou hodnotou (pro mapovací účely nepodstatné).

2. *Senzor s GNSS modulem* - v TTN síťovém serveru byla vytvořena nová HTTP integrace, která zprávy přeposílá pomocí HTTP POST na URL přístupový bod <https://integrations.ttnmapper.org/ttn/v2/>. Zároveň bylo nutné vytvořit dekodovací funkci, která dekoduje raw payload data do JSON struktury, povinné položky latitude, longitude, altitude a HDOP (viz 4.3.1). Volitelně je možné pomocí POST parametrů předat jméno experimentu (nebyla použita globální mapa).

Rozhraní mapy nově (na základě požadavků TTN komunity) umožňuje vyfiltrování konkrétních bran v experimentu (vhodné pokud majitele zajímá pouze jeho brána, nikoliv brány ostatních uživatelů, které mapu graficky znepřehledňují). Kompletní mapu síly signálů (RSSI) je možné vidět na obr. 7.1. Obrázek znázorňuje jednotlivé body, kde bylo měření provedeno, a tenké čáry, které jej propojují s branami, které zprávu od senzoru přijaly. Úroveň signálu je značena graficky - čím je barva teplejší, tím je RSSI vyšší. Síla čar je ovlivněna počtem provedených měření, tj. jejich signifikantností. Černě označené brány odpovídají synchronizovaným branám s GNSS modulem, jenž byly pro měření použity<sup>20</sup> (vyfiltrovány). Zbylé - modře označené - brány patří členům TTN komunity a pro účely této práce s nimi nebylo kalkulováno. Pro snazší pohyb a pokrytí rozsáhlejšího prostoru byl senzor připevněn na kolo viz fotka v příloze B.4.

## 7.2 Experimentální měření

Na základě vytvořené mapy pokrytí byly zvoleny dvě testovací místa. Test č. 1 v oblasti Strahova, konkrétně Růžový sad Petřín; test č. 2 v oblasti parku Parukářka. Testovací senzor byl pro účely měření přepnut do periodického režimu odesílání, s časovým intervalem cca 15 s. Pro výpočet byla použita kombinace Chanova algoritmu pro 3 brány a linearizace s LS pro více jak 3 brány (2D verze). Samotná aplikace SensorLoc byla spuštěna na virtuálním serveru poskytnutém sdružením CESNET.

Výsledky uložené v Mongo DB byly získány a následně upraveny pomocí Mongo shell (JavaScript prostředí) a Bash skriptů. Např. příkaz

---

<sup>20</sup>Čtyři brány provozovány sdružením CESNET: na střeše budovy sídla CESNET, z. s. p. o. v Dejvicích; Wiehlův dům Václavské náměstí; blok B na podolských kolejích a na budově ČTK v Žirovnické ulici. Jedna byla umístěna - dočasně - na bloku 10 strahovských kolejí, za asistence klubu Siliconhill (Studentská unie ČVUT).

```

mongo --quiet localization --eval 'db.messages.find(
  {$and: [{"calculated.status": "2D"}, {"time": {"$gt"
    : ISODate("2018-04-25T12:05:00.000Z")}}, {"time": {"$lt"
    : ISODate("2018-04-25T13:15:00.000Z")}}]}.sort( {"calculated.error": 1} )
  .forEach(function(x) {
    print(x.calculated.lat,",",x.calculated.lon,",small_purple" ); })'

```

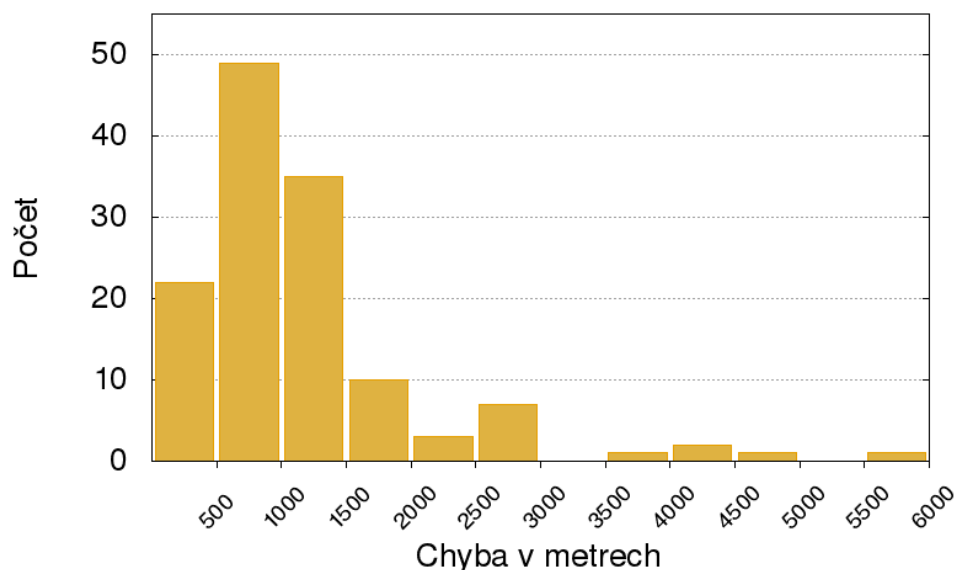
vyfiltruje úspěšně spočtené pozice, pro zprávy v určitém časovém rozsahu<sup>21</sup>, seřadí vzestupně podle odchylky a vytiskne jejich zeměpisnou šířku a délku, spolu s indikátorem pro typ použité značky v Google mapách, a to ve formátu CSV (Comma Separated Values). Takto získaná data byla použita jako vstup do *Google Fusion Tables* [60] viz následující odstavce a mapa s pozicemi. Data jako taková byla následně zbavena krajních hodnot v procentuálním zastoupení menším než 1-2 %.

Na obr. 7.2 a 7.3 lze vidět histogram jednotlivých chyb v testu provedeném na Strahově, resp. Parukářce. Šířka intervalu byla zvolena po 500 metrech.

<sup>21</sup> Časové porovnání je možné, jelikož se časová značka - v textové podobě *string* - před samotným uložením převede na *datetime.datetime* objekt.



Obrázek 7.1: Sestavení senzoru - zkompletovaná horní část



Obrázek 7.2: Histogram ze 131 odeslaných zpráv - test č. 1 Strahov

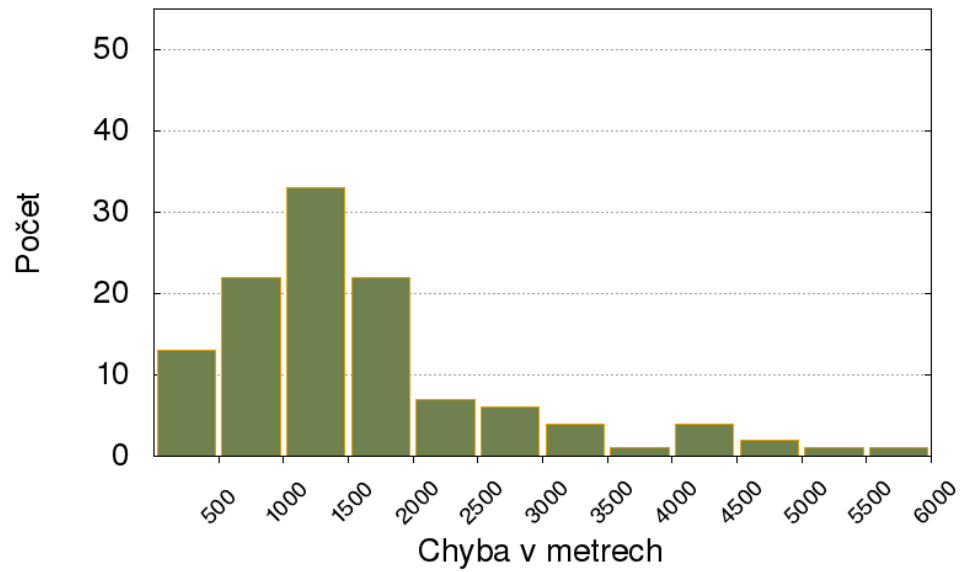
Tabulka 7.1: Statistika z provedených testů (chyby v metrech)

	Test 1 (Strahov)	Test 2 (Parukářka)
Počet lokalizovaných zpráv	131	116
Počet chybových zpráv	8	4
Nejmenší chyba [m]	137	42
Největší chyba [m]	5755	5503
Aritmetický průměr chyb [m]	1160	1623
Medián chyb [m]	946	1394

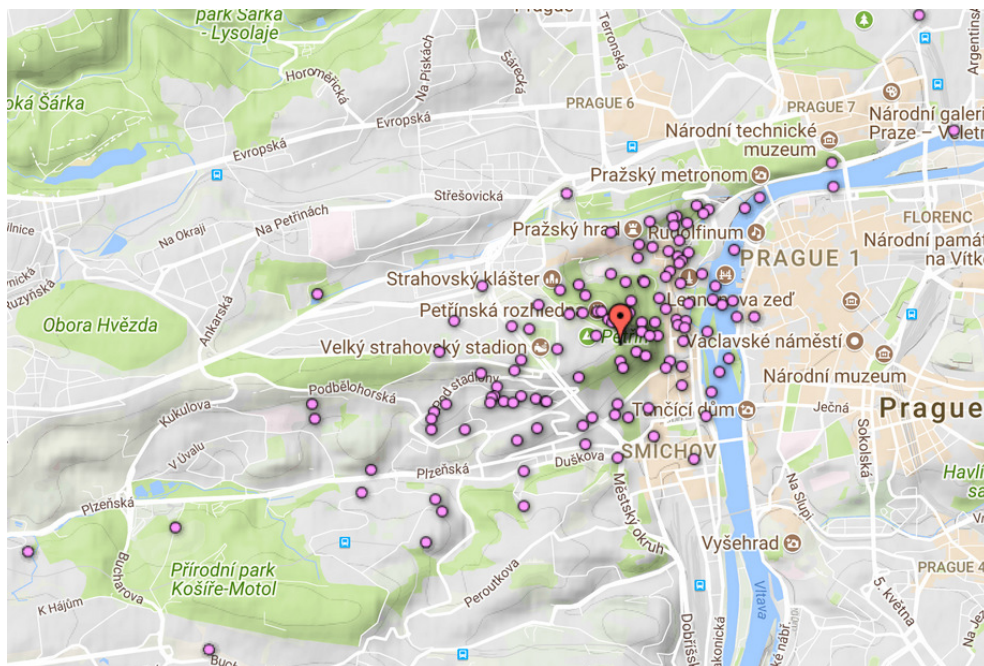
Jednotlivé - spočtené - pozice ve vztahu k referenční (přesné) hodnotě, tj. jejich distribuce lze vidět na obrázku 7.4 a heat mapu (s 15 radiusem a totožným přiblížením) lze vidět na obr. 7.5.

Lépe dopadl test na Strahově, kde byla většina (medián) odchylek menší než 1 km a nejvíce naměřených chyb bylo v rozsahu 500-1000 m. V tabulce 7.1 lze vidět další statistiky jako je nejlepší/nejhorší výsledek a aritmetický průměr či medián. Počet lokalizovaných zpráv určuje odeslané pakety, jejichž pozice byla úspěšně spočtena (ať už „2D“ nebo „3D“ status), chybové zprávy označují ty, u nichž výpočetní algoritmus selhal („error“ status) - nevhodná geometrie, příliš vysoké odchylky v časech (vychýlená synchronizace, vlnové odrazy), tj. neexistuje průnik hyperbol/hyperboloidů nebo algoritmus nekoneverguje, a výpočetní matice se eventuálně stane singulární.

## 7.2. Experimentální měření



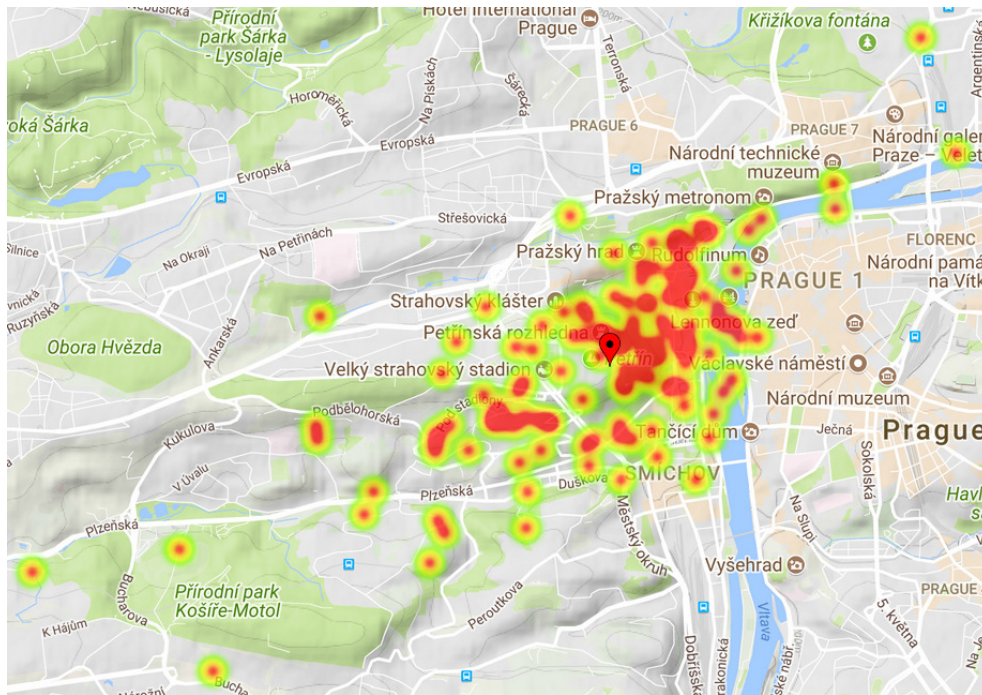
Obrázek 7.3: Histogram ze 116 odeslaných zpráv - test č. 2 Parukářka



Obrázek 7.4: Rozptyl spočtených hodnot pozice ku referenční (test č. 1)



## 7. TESTOVÁNÍ A MĚŘENÍ



Obrázek 7.5: Rozptyl spočtených hodnot pozice - heat mapa (test č. 1)

### 7.3 Zhodnocení

Naměřené výsledky demonstrují možnosti lokalizace v LoRaWAN síti za využití první generace LoRa bran - prakticky dosažitelnou přesnost a stabilitu, jenž je dána závislostí na fyzikálních principech rádiového přenosu. V porovnání s přesnějšími metodami (ve větších městech) jako GNSS je přímé srovnání nerovné, alespoň z pohledu absolutní přesnosti (situace by se znatelně zlepšila využitím druhé generace bran). Jedná se ovšem o lokalizaci, která nevyžaduje další prostředky, ať už energetické či investiční. Výpočet polohy je prováděn na základě běžného provozu senzoru (de facto vedlejší produkt primární funkce). Metoda může být dobře využita pro hrubý odhad pozice či různé varianty a aplikace geofencingu. Při chybovém mediánu menším než 1 km, lze při více odeslaných paketech operovat s vysokou pravděpodobností, že je spočtená poloha v toleranci této nepřesnosti.

Z důvodu nižšího počtu bran byly zvoleny - pro test - vhodné lokality. Není tedy předneseno objektivní stanovisko v rámci celého města. Situace na venkově a v méně obydlených oblastech bude z pohledu testované metody příznivější a požadavky na počet bran budou nižší (nebude docházet k odrazům, častěji bude nastávat přímá viditelnost, pásmo nebude zarušené).

**Možnosti zpřesnění**

Zapojením vyššího počtu bran, které by mezi sebou byly synchronizována a zvýšit přesnost přiřazení časové známky (momentálně kolísá řádově v jednotkách  $\mu\text{s}$ <sup>22</sup>). Aplikováním statistických metod a AI algoritmů. Momentálně jsou největším zdrojem nepřesností přiřazené časy a odrazy, tyto jevy se sčítají a je obtížné jejich odlišení, či vyvození nějakých důsledků - tedy zpřesnění. Eliminace nepřesných značek otevírá prostor pro aplikování zmíněných metod. Další možností je využití více antén, např. vertikálně a horizontálně polarizovaných.

---

<sup>22</sup>Testováno umístěním dvou bran v těsné blízkosti.





---

## Závěr

V průběhu práce jsem si prošel celým procesem lokalizace od návrhu senzoru a praktických měření po studium algoritmů či principů užívaných v radiové navigaci a lokalizaci. Naučil jsem se pracovat s 3D tiskárnou a jednotlivými materiály, provádět úpravy - pro tiskárnu - specifických parametrů (výška vrstvy, z osy), nebo třeba modelovat jednotlivé díly a provádět následnou kontrolu správného vygenerování. Reálné využití matematických principů a sledování jejich výsledků - změna pozice - bylo taktéž přínosné.

Vytyčené cíle práce jsem splnil: testovací senzor navrhl, sestavil a otestoval; potřebnou infrastrukturu připravil a nakonfiguroval; vhodné výpočetní algoritmy nastudoval a následně implementoval do uceleného řešení, a v poslední části práce všechny tyto související články celého procesu otestoval experimentálním měřením. Navržený způsob je v praxi použitelný a vhodný především pro specifické aplikace (např. geofencing, či mapování na předem známé objekty/polohy). Předpokladem je dostatečné pokrytí a vyhovující typ LoRaWAN bran - ty se ukázaly být nejvíce limitujícím faktorem, jelikož testované nepřizpůsobily dostatečně přesné časové značky. Novější brány, navržené pro potřeby geolokace, by tyto výsledky výrazně zlepšily. LoRa geolokalizaci tedy lze zařadit mezi dostupné lokalizační metody, vždy je ale nutné nejprve provést analýzu požadavků, a porozumět limitům/možnostem, jenž tato technologie nabízí.

Obsahové zaměření této práce zároveň nabízí řadu možných témat pro další či navazující diplomové nebo dizertační práce. A to jak z oblasti HW, tak SW. Velmi užitečným se jeví návrh integrovaného obvodu s rádiovým modulem, který by umožňoval přesnější přiřazování časových známek přijatých paketů, případně využití více antén s odlišnou polarizací. Dále např. aplikování statistických metod a algoritmů umělé inteligence na metadata získaná z jednotlivých bran. Případně aplikování vhodné kombinace pokročilých iteračních algoritmů pro řešení soustav hyperbolických rovnic. Nebo provedení návrhu, simulace a měření při požadavku na „real time“ zpětnou vazbu o pozici - vhodné škálování, úprava architektury, návrh DB nebo způsob obsluhy.



---

## Literatura

- [1] Sornin, N.: Interview Nicolas Sornin - Inventor of LoRa [online]. In: Youtube, Únor 2018, [cit. 2018-02-18]. Dostupné z: <https://www.youtube.com/watch?v=S2A792v-ZGY>
- [2] Semtech: LoRa Technology [online]. [cit. 2018-02-23]. Dostupné z: <https://www.semtech.com/technology/lora/what-is-lora>
- [3] LoRa Alliance [online]. [cit. 2016-04-05]. Dostupné z: <https://www.lora-alliance.org>
- [4] CRA: Služby CRA IoT [online]. [cit. 2018-03-23]. Dostupné z: <https://www.cra.cz/sluzby-iot>
- [5] Symphony Link [online]. [cit. 2016-04-05]. Dostupné z: <https://www.link-labs.com/symphony>
- [6] LoRa Alliance: *LoRaWAN specification [online]*. [cit. 2016-04-05]. Dostupné z: <https://www.lora-alliance.org>
- [7] Všeobecné oprávnění č. VO-R/10/11.2016-13 k využívání rádiových kmitočtů a k provozování zařízení krátkého dosahu [online]. [cit. 2018-03-16]. Dostupné z: <https://www.ctu.cz/sites/default/files/obsah/ctu/vseobecne-opravneni-c.vo-r/10/11.2016-13/obrazky/vo-r10-112016-13.pdf>
- [8] E-mailová korespondence s Petr Foltýn [online]. Březen 2016, [foltyn.petr@solidustech.cz](mailto:foltyn.petr@solidustech.cz).
- [9] IQRF technology [online]. [cit. 2018-02-18]. Dostupné z: <https://www.iqrf.org/>
- [10] Schatz, G.: SigFox Vs. LoRa: A Comparison Between Technologies and Business Models [online]. Leden 2016, [cit. 2018-02-18]. Dostupné z: <https://www.link-labs.com/blog/sigfox-vs-lora>

- [11] Sigfox: Sigfox Technology Overview [online]. [cit. 2018-02-18]. Dostupné z: <https://www.sigfox.com/en/sigfox-iot-technology-overview>
- [12] T-Mobile Launches Nation's First Plan for Narrowband IoT [online]. [cit. 2018-03-17]. Dostupné z: <https://newsroom.t-mobile.com/news-and-blogs/narrowband-iot.htm>
- [13] Service, N. O.: What is the geoid? [online]. [cit. 2018-02-27]. Dostupné z: <https://oceanservice.noaa.gov/facts/geoid.html>
- [14] Fraczek, W.: Mean Sea Level, GPS, and the Geoid [online]. September 2003, [cit. 2018-02-27]. Dostupné z: <http://www.esri.com/news/arcuser/0703/geoid1of3.html>
- [15] Mendelova univerzita v Brně: *Kartografická zobrazení [online]*. [cit. 2018-03-01]. Dostupné z: [https://is.mendelu.cz/eknihovna/opory/zobraz\\_cast.pl?cast=59996](https://is.mendelu.cz/eknihovna/opory/zobraz_cast.pl?cast=59996)
- [16] Britannica, E.: The Mercator projection [online]. [cit. 2018-02-27]. Dostupné z: <https://kaiserscience.wordpress.com/earth-science/maps/mercator-maps-use-and-criticism/>
- [17] Ball State University Libraries: *Maps and Cartography: Map Projections [online]*. [cit. 2018-03-01]. Dostupné z: <http://www.bsu.edu/libraries/collections/gcmc/tutorials/pdfs/mapscartographymapprojections.pdf>
- [18] Aitchison, A.: The Google Maps / Bing Maps Spherical Mercator Projection [online]. [cit. 2018-02-27]. Dostupné z: <https://alastaira.wordpress.com/2011/01/23/the-google-maps-bing-maps-spherical-mercator-projection/>
- [19] GMT: Universal Transverse Mercator zone layout [online]. [cit. 2018-02-27]. Dostupné z: [https://gmt-docs.readthedocs.io/zh\\_CN/5.2.1/\\_images/GMT\\_utm\\_zones.png](https://gmt-docs.readthedocs.io/zh_CN/5.2.1/_images/GMT_utm_zones.png)
- [20] Lei, Z.: China's Beidou satellite system upgraded for global reach [online]. [cit. 2018-03-03]. Dostupné z: <https://www.telegraph.co.uk/news/world/china-watch/technology/new-china-satellite-system/>
- [21] Stevens, J.: Global Positioning Systems [online]. [cit. 2018-03-03]. Dostupné z: <https://www.e-education.psu.edu/geog160/node/1923>
- [22] Elliott Kaplan, C. H.: *Understanding GPS: Principles and Applications*. Boston: Artech-House, druhé vydání, 2006, 703 s., iISBN 1-58053-894-0.
- [23] GPS Tutorial [online]. [cit. 2018-03-03]. Dostupné z: [http://www.trimble.com/gps\\_tutorial/](http://www.trimble.com/gps_tutorial/)

- 
- [24] Melton, R. G.: Details of the GPS position calculation [online]. [cit. 2018-03-03]. Dostupné z: [https://www.courses.psu.edu/aersp/aersp055\\_r81/satellites/gps\\_details.html](https://www.courses.psu.edu/aersp/aersp055_r81/satellites/gps_details.html)
- [25] J.S. Abel, J. C.: Existence and uniqueness of GPS solutions [online]. [cit. 2018-03-03]. Dostupné z: <http://ieeexplore.ieee.org/document/104271/>
- [26] GPS Solutions: Closed Forms, Critical and Special Configurations of P4P [online]. [cit. 2018-03-03]. Dostupné z: <https://link.springer.com/article/10.1007%2FPL00012897>
- [27] Kalman, D.: An Underdetermined Linear System for GPS [online]. [cit. 2018-03-03]. Dostupné z: [https://www.maa.org/sites/default/files/pdf/upload\\_library/22/Polya/Kalman.pdf](https://www.maa.org/sites/default/files/pdf/upload_library/22/Polya/Kalman.pdf)
- [28] Committee, L. A. S.: GEOLOCATION WHITEPAPER [online]. January 2018, [cit. 2018-03-04]. Dostupné z: [https://docs.wixstatic.com/ugd/eccc1a\\_d43b3b29dfff4ec2b00f349ced4225c4.pdf](https://docs.wixstatic.com/ugd/eccc1a_d43b3b29dfff4ec2b00f349ced4225c4.pdf)
- [29] Hyperbola and Parabola [online]. [cit. 2018-03-04]. Dostupné z: <https://www.math24.net/hyperbola-and-parabola/>
- [30] Ray, B.: LoRa Localization [online]. [cit. 2018-03-04]. Dostupné z: <https://www.link-labs.com/blog/lora-localization>
- [31] Semtech's LoRa® Geolocation Solution for Low Power Wide Area Networks is Now Available [online]. [cit. 2018-03-04]. Dostupné z: <https://www.semtech.com/company/press/Semtechs-LoRa-Geolocation-Solution-for-Low-Power-Wide-Area-Networks-is-Now-Available>
- [32] Wirnet iBTS [online]. [cit. 2018-03-04]. Dostupné z: <https://www.kerlink.com/product/wirnet-ibts/>
- [33] Cisco Wireless Gateway for LoRaWAN [online]. [cit. 2018-03-04]. Dostupné z: <https://www.cisco.com/c/en/us/products/routers/wireless-gateway-lorawan/index.html>
- [34] Semtech Announces Industry's First Disposable LoRa-Enabled Nano-tag for Internet of Things (IoT) Applications [online]. [cit. 2018-03-04]. Dostupné z: <https://www.semtech.com/company/press/semtech-announces-industrys-first-disposable-lora-enabled-nano-tag-for-internet-of-things-iot-applications>
- [35] Lansdowne, R.: Geolocation - Unlocking New Value For IoT Solutions. In *Geo IoT World 2017*, Brussel, Belgium: Semtech, 2017.

- [36] Objenious: Ubysol: Solution for the traceability of materials [online]. In: Youtube, Prosinec 2017, [cit. 2018-03-04]. Dostupné z: <https://youtu.be/TcFXi1ABfDk>
- [37] WEYN, M.: MULTIMODAL LOW-POWER COMMUNICATION AND LOCALIZATION. In *Geo IoT World 2017*, Brussel, Belgium: imec, 2017.
- [38] KPN launches world's first nationwide LoRa Internet of Things Network, powered by Actility's ThingPark [online]. [cit. 2018-03-04]. Dostupné z: <https://www.actility.com/news/kpn-launches-nationwide-lora-internet-of-things-network/>
- [39] News [online]. [cit. 2018-03-04]. Dostupné z: <http://allsetra.nl/nieuws/>
- [40] Skyhook [online]. [cit. 2018-03-03]. Dostupné z: <https://www.skyhookwireless.com/>
- [41] Tracko [online]. [cit. 2018-05-04]. Dostupné z: <https://www.onyxbeacon.com/tracko/>
- [42] Sewio [online]. [cit. 2018-05-04]. Dostupné z: <https://www.sewio.net/>
- [43] Pozyx [online]. [cit. 2018-05-04]. Dostupné z: <https://www.pozyx.io/>
- [44] Lugt, G.: Light, Location, Magic: lift indoor location to the next level with light. In *Geo IoT World 2017*, Brussel, Belgium: PHILIPS Lighting, 2017.
- [45] Hourri, C.: Precise positioning using Computer Vision. In *Geo IoT World 2017*, Brussel, Belgium: ACCUWARE, 2017.
- [46] Jones, K.: Precise positioning using Computer Vision. In *Geo IoT World 2017*, Brussel, Belgium: Skyhook, 2017.
- [47] resin.io [online]. [cit. 2018-03-08]. Dostupné z: <https://resin.io/>
- [48] Meijers, J.: ttn-resin-gateway-rpi. In: GitHub, 2018, [cit. 2018-03-08], commit = 43f611a3d22ff4c093af2d80b4471499f4dce245. Dostupné z: <https://github.com/jpmeijers/ttn-resin-gateway-rpi>
- [49] LoRa network: lora\_gateway. In: GitHub, 2018, [cit. 2018-03-08]. Dostupné z: [https://github.com/Lora-net/lora\\_gateway](https://github.com/Lora-net/lora_gateway)
- [50] The Things Network [online]. [cit. 2018-03-09]. Dostupné z: <https://www.thethingsnetwork.org/>
- [51] libelium dev: Integrated pull-up resistor [online]. In: Diskuzní fórum Libelium (technická podpora), [cit. 2018-03-08]. Dostupné z: <https://www.libelium.com/forum/viewtopic.php?f=26&t=25521>

- 
- [52] 3D Tiskárna Poseidon DUO [online]. [cit. 2018-03-08]. Dostupné z: <http://reprapobchod.cz/>
- [53] Dassault Systèmes SE: *SolidWorks [software]*. [cit. 2018-04-10]. Dostupné z: <https://www.solidworks.com/>
- [54] Autodesk, Inc.: Netfabb [software]. [cit. 2018-04-10]. Dostupné z: <https://www.autodesk.com/products/netfabb/overview>
- [55] Simplify3D, Inc.: Simplify 3D [software]. [cit. 2018-04-10]. Dostupné z: <https://www.simplify3d.com/>
- [56] Slic3r [software]. [cit. 2018-04-10]. Dostupné z: <http://slic3r.org/>
- [57] Chan, Y.; Ho, K.: A Simple and Efficient Estimator for Hyperbolic Location. *IEEE Transactions on Signal Processing*, ročník 42, č. 8, Srpen 1994: s. 1905–1915, ISSN 1941-0476, publikováno IEEE.
- [58] Foy, W. H.: Position-Location Solutions by Taylor-Series Estimation. *IEEE Trans. Aerospace and Electronic Systems*, ročník AES-12, č. 2, Březen 1976: s. 187–194, ISSN 0018-9251, publikováno IEEE.
- [59] Meijers, J.: ttnmapper. [cit. 2018-03-31]. Dostupné z: <https://ttnmapper.org>
- [60] Google Fusion Tables [online]. [cit. 2018-04-28]. Dostupné z: <https://sites.google.com/site/fusiontablestalks/home>





## Seznam použitých zkratk

- ABP** Activation by Personalization
- ADR** Adaptive Data Rate
- AES** Advanced Encryption Standard
- AI** Artificial Intelligence
- API** Application Programming Interface
- ASCI** Advertising Standards Council of India
- BLE** Bluetooth Low Energy
- BSSID** Basic Service Set Identifier
- BTS** Base Transmit Station
- CRC** Cyclic Redundancy Check
- CSV** Comma Separated Values
- ČTU** Český telekomunikační úřad
- DB** Database
- EU** European Union
- EUI** Extended Unique Identifier
- FPGA** Field Programmable Gate Array
- FSK** Frequency-shift Keying
- GNSS** Global Navigation Satellite System
- GPS** Global Positioning System

## A. SEZNAM POUŽITÝCH ZKRATEK

---

**GSM** Global System for Mobile Communications

**GW** Gateway

**HAL** Hardware Abstraction Layer

**HDOP** Horizontal Dilution of Precision

**HTTP** Hypertext Transfer Protocol

**IoE** Internet of Everything

**IoT** Internet of Things

**IP** Internet Protocol

**ISM** Industrial, Scientific and Medical

**LBT** Listen Before Talk

**LCS** Location Services

**LLA** Latitude, Longitude and Altitude

**LLH** Latitude, Longitude and Height

**LPWAN** Low-power Wide-Area Network

**LSB** Least Significant Bit

**MAC** Media Access Control

**MQTT** Message Queuing Telemetry Transport

**MSL** Mean Sea Level

**NFC** Near-field Communication

**NMEA** National Marine Electronics Association

**NTP** Network Time Protocol

**OTAA** Over-the-Air Activation

**PoE** Power over Ethernet

**PPS** Pulse-per-Second

**QoS** Quality of Service

**RSSI** Radio Signal Strength Indicator

**SF** Spreading Factor

---

**SNR** Signal-to-Noise Ration

**SPI** Serial Peripheral Interface

**SSH** Secure Shell

**SSID** Service Set Identifier

**TDoA** Time Difference of Arrival

**TOA** Time of Arrival

**ToF** Time of Flight

**TTN** The Things Network

**UART** Activation by Personalization

**WAN** Wide-Area Network



## Fotky z instalace bran a testování



(a) První verze LoRaWAN brány

(b) Druhá (nová) verze LoRaWAN brány

Obrázek B.1: Instalované LoRaWAN brány

## B. FOTKY Z INSTALACE BRAN A TESTOVÁNÍ

---



Obrázek B.2: Vnější pohled na LoRaWAN bránu (verze 1)



Obrázek B.3: Vnější pohled na LoRaWAN bránu (verze 2)



Obrázek B.4: Testovací kolo - mapování síly signálu





## Testovací senzor konstrukce



Obrázek C.1: Sestavení senzoru - přehled jednotlivých dílů

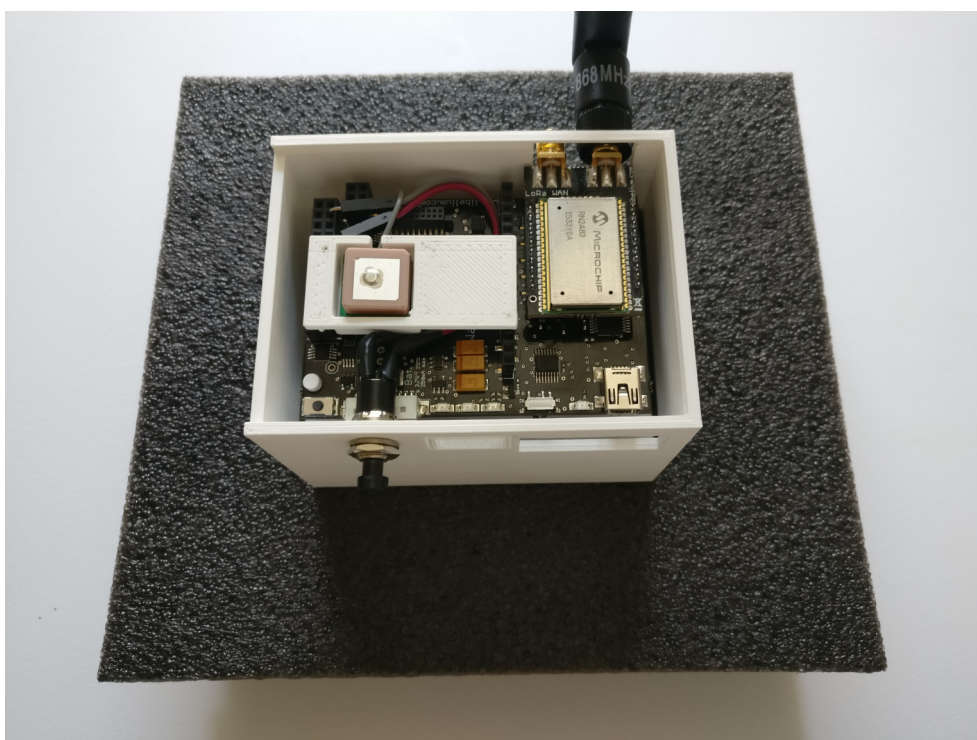
## C. TESTOVACÍ SENZOR KONSTRUKCE



(a) Vanička externí baterie

(b) Hlavní úchyt modulu

Obrázek C.2: Sestavení senzoru - spodní část



Obrázek C.3: Sestavení senzoru - zkompletovaná horní část

---

## Obsah přiloženého CD

readme.txt.....	stručný popis obsahu CD
src	
_ impl.....	zdrojové kódy implementace
_ 3D print.....	STL soubory tisknutých dílů
_ thesis.....	zdrojová forma práce ve formátu $\LaTeX$
text.....	text práce
_ DP_Pospíšilík_Jiří_2018.pdf.....	text práce ve formátu PDF
_ DP_Pospíšilík_Jiří_2018.ps.....	text práce ve formátu PS