

**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**

Fakulta elektrotechnická

Katedra telekomunikační techniky

**Analýza pokrytí zadaného území signálem  
pro technologii RTLS v pásmu 2,45 GHz**

**Květen 2014**

**Bakalant: Martin Cihlář**

**Vedoucí práce: Ing. Jaromír Hrad, Ph.D.**

## **Poděkování**

Rád bych poděkoval vedoucímu práce Ing. Jaromíru Hradovi, Ph.D. za ochotu zajistit mi všechny klíčové prvky při řešení práce, také za odborné vedení a řadu cenných připomínek. Rád bych také poděkoval Ing. Jaromíru Doležalovi, Ph.D. a Ing. Lukáši Vojtěchovi, Ph.D. za umožnění přístupu do prostor CAT. V neposlední řadě bych rád poděkoval i Ing. Marku Nerudovi za poskytnutí balíčku Ekahau a spektrálního analyzátoru.

## Čestné prohlášení

Prohlašuji, že jsem zadanou bakalářskou práci zpracoval sám s přispěním vedoucího práce a konzultanta a používal jsem pouze literaturu v práci uvedenou. Dále prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé bakalářské práce nebo její části se souhlasem katedry.

Datum: 23. května 2014

.....  
podpis bakalanta

České vysoké učení technické v Praze  
Fakulta elektrotechnická

katedra telekomunikační techniky

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Cihlář Martin**

Studijní program: Komunikace, multimédia a elektronika  
Obor: Síťové a informační technologie

Název tématu: **Analýza pokrytí zadaného území signálem pro technologii RTLS  
v pásmu 2,45 GHz**

Pokyny pro vypracování:

Seznamte se s dostupnou technologií RTLS. Pomocí spektrálního analyzátoru proveďte měření pokrytí zadané oblasti signálem v pásmu 2,45 GHz, výsledky vizualizujte a porovnejte s dostupnými výstupy modelu vytvořeného pomocí plánovacího software. Výsledky měření analyzujte a podle potřeby dále využijte pro návrh opatření a postupů vedoucích k optimalizaci realizované síťové infrastruktury WLAN tak, aby ji bylo možno využít pro účely prostorové lokalizace. Navrhněte lokalizační systém schopný určit polohu sledovaného objektu či uživatele v rámci budovy či její části. Vytvořte uživatelské rozhraní pro správce lokalizačního systému na platformě Android.

Seznam odborné literatury:

- [1] Malik, A.: RTLS for Dummies. Wiley Publishing, Inc. 2009. ISBN 978-0-470-39868-5.
- [2] Dokumentace Ekahau RTLS - <http://www.ekahau.com/> [online]

Vedoucí: Ing. Jaromír Hrad, Ph.D.

Platnost zadání: do konce letního semestru 2014/2015

prof. Ing. Boris Šimák, CSc.  
vedoucí katedry



prof. Ing. Pavel Ripka, CSc.  
děkan

V Praze dne 9. 12. 2013

## **Anotace:**

Tato práce se zabývá problematikou využití lokalizačních systémů v rámci budovy. Pro tyto účely byla využita stávající bezdrátová lokální síť, jejíž parametry bylo nejprve nutné analyzovat spektrálním analyzátozem a posléze i bezdrátovým USB adaptérem. Poté byly získané výsledky analyzovány, zda vyhovují požadavkům lokalizačních systémů. Paralelně s optimalizací sítě probíhal vývoj mobilní aplikace pro platformu Android OS. Aplikace je určena pro správce lokalizačního systému, kde i samotný návrh tohoto systému je také jedno z témat práce. V úplném závěru práce jsou provedeny provozní zkoušky a implementace optimalizací výpočetního algoritmu navrhovaného lokalizačního systému a s tím související aplikace správce.

## **Klíčová slova:**

analýza pokrytí, Android OS, bezdrátová lokální síť, lokalizace v reálném čase, lokalizační systém, mobilní aplikace, poziční identifikátor, algoritmus *Nejbližší Soused*

## **Annotation:**

This thesis deals with the issue of using location systems within the building. For these purposes the existing wireless local area network has been used. First, network parameters were necessary to analyze with spectral analyzer and then with the wireless USB adapter. Then the obtained results have been analyzed for their compliance with the requirements of location systems. The mobile application for platform Android OS was being developed in parallel with the optimization of the network. The application is intended for administrator of location system. Design of this system is also one of the themes of the thesis. At the end of the work, operational tests and implementation of computing algorithm optimization of proposed localization system and related administrator application are executed.

## **Keywords:**

coverage analysis, Android OS, wireless local area network, real-time localization, location system, mobile application, position identifier (tag), algorithm *Nearest Neighbor*

# Obsah

<b>Seznam použitých zkratek .....</b>	<b>viii</b>
<b>1. Úvod.....</b>	<b>1</b>
<b>2. Bezdrátové datové sítě.....</b>	<b>2</b>
2.1 Typy sítí dle rozlohy .....	2
2.2 Technologie Wi-Fi .....	3
2.3 Bezdrátové lokální sítě .....	5
<b>3. Lokalizační systémy.....</b>	<b>7</b>
3.1 Vymezení pojmu RTLS .....	7
3.2 Součásti systému .....	8
3.3 Techniky získávání dat.....	9
3.4 Algoritmy výpočtu pozice.....	11
3.5 Lokalizační systém Ekahau.....	14
<b>4. Analýza pokrytí bezdrátové sítě.....</b>	<b>16</b>
4.1 Simulace pokrytí s využitím Ekahau nástrojů.....	16
4.2 Analýza pokrytí s využitím měřicích přístrojů.....	18
4.2.1 Analýza spektrálním analyzátozem.....	18
4.2.2 Analýza externím Wi-Fi adaptérem.....	21
4.3 Implementace Ekahau systému .....	23
<b>5. Mobilní aplikace .....</b>	<b>24</b>
5.1 Návrh lokalizačního systému .....	24
5.2 Platforma Android.....	27
5.3 Použité vývojové nástroje .....	28
5.4 Návrh aplikace pro správce.....	29
5.5 Zpracování návrhu aplikace .....	30
5.5.1 Programové třídy aplikace .....	30
5.5.2 Aktivita RTLS.java.....	31
5.5.3 Aktivita Survey.java.....	32
5.5.4 Aktivita Localization.java.....	33
5.5.5 Aktivita Settings.java .....	35
5.6 Zhodnocení přesnosti lokalizačního systému.....	36
5.6.1 Optimalizace přesnosti lokalizace .....	36
5.6.2 Výsledná přesnost lokalizačního systému .....	38
<b>6. Závěr.....</b>	<b>40</b>

<b>Seznam použité literatury .....</b>	<b>42</b>
<b>Seznam příloh.....</b>	<b>44</b>
<b>A - Vizualizace pokrytí jednotlivých vysílačů z měření spektr. analyzátozem .....</b>	<b>45</b>
<b>B - Vizualizace pokrytí jednotlivých vysílačů z měření USB adaptérem .....</b>	<b>48</b>
<b>C - Vizualizace přesnosti stanovení pozice lokalizačním systémem.....</b>	<b>51</b>
<b>D - Obsah přiloženého CD .....</b>	<b>52</b>
<b>E - Návod pro obsluhu mobilní aplikace <i>RTLS Demo (RTLS Admin)</i> .....</b>	<b>53</b>

## Seznam použitých zkratek

AOA - *Angle of Arrival* (lokalizační metoda na základě směru dopadajícího signálu)  
AP - *Access Point* (přístupový bod - zařízení bezdrátové lokální sítě)  
DDMS - *Dalvik Debug Monitor Server* (ladící nástroj pro vývoj aplikací)  
DSSS - *Direct Sequence Spread Spectrum* (metoda rozprostření spektra signálu)  
ETSI - *European Telecommunications Standards Institute*  
FCC - *Federal Communications Commission*  
FHSS - *Frequency Hopping Spread Spectrum* (metoda rozprostření spektra signálu)  
FTP - *File Transfer Protocol* (protokol pro přenos souborů mezi prvky sítě)  
IEEE - *Institute of Electrical and Electronics Engineers*  
ISM - *Industrial, Scientific and Medical* (přenosová pásma pro bezlicenční využití)  
ITU - *International Telecommunication Union*  
KNN - *K Nearest Neighbors* (výpočetní algoritmus pro nalezení nejbližšího souseda)  
LAN - *Local Area Network* (lokální síť)  
MAC - *Media Access Control* (řízení přístupu k médiu na základě fyzických adres)  
MAN - *Metropolitan Area Network* (metropolitní síť)  
MIMO OFDM - *Multiple Input Multiple Output OFDM*  
MU-MIMO - *Multi User MIMO* (vstupně/ výstupní operace s podporou více uživatelů)  
NN - *Nearest Neighbor* (výpočetní algoritmus pro nalezení nejbližšího souseda)  
OFDM - *Orthogonal Frequency Division Multiplexing* (modulace signálu)  
PAN - *Personal Area Network* (osobní síť)  
RFID - *Radio Frequency Identification* (identifikace s využitím rádiových frekvencí)  
RSSI - *Received Signal Strength Indicator* (indikátor přijaté síly signálu)  
RTLS - *Real-Time Location System* (systém pro lokalizaci v reálném čase)  
SQL - *Structured Query Language* (dotazovací jazyk pro práci s databázemi)  
SSID - *Service Set Identifier* (jednoznačný identifikátor bezdrátové sítě)  
TDOA - *Time Distance of Arrival* (lokal. metoda na základě času příchodu signálu)  
TOA - *Time of Arrival* (lokalizační metoda na základě času příchodu signálu)  
WAN - *Wide Area Network* (geograficky rozlehlá síť)  
WiGig - *Wireless Gigabit Alliance*  
WKNN - *Weighted K Nearest Neighbors* (algoritmus pro nalezení nejbližšího souseda)  
WLAN - *Wireless Local Area Network* (bezdrátová lokální síť)  
WMAN - *Wireless Metropolitan Area Network* (bezdrátová metropolitní síť)  
WPAN - *Wireless Personal Area Network* (bezdrátová osobní síť)  
WWAN - *Wireless Wide Area Network* (bezdrátová rozlehlá síť)  
XML - *Extensible Markup Language* (zde jazyk pro formátování vzhledu aplikace)



# 1. Úvod

Za posledních pár let zažil obor lokalizačních systémů značný vývoj. Hlavní podíl bych přiřkl samotným výrobcům přenosných zařízení, kteří začali produkovat cenově dostupné chytré telefony či tablety. Výsledkem je přítomnost tohoto druhu osobních zařízení u větší části populace, a proto mohl, z pohledu široké veřejnosti, do popředí zájmu proniknout fenomén navigace či lokalizace v budovách.

Tato práce je zaměřena právě na obor lokalizačních systémů. Je však konkretizována na využití, dnes velmi v budovách rozšířených, bezdrátových lokálních sítí, jež jsou prvním stavebním prvkem pro tuto práci. Druhým prvkem je nejrozšířenější mobilní platforma, a to Android OS. Spojení těchto dvou prvků je z pohledu lokalizace velmi zajímavé, a proto se jím budu v této práci zabývat, ačkoli to bude má první zkušenost v rámci programování mobilní aplikace a ačkoli můj mobilní telefon nedisponuje platformou Android OS.

Prvním hlavním cílem práce je analýza bezdrátové lokální sítě pro konkrétní prostory univerzity z pohledu jejich vlastností, které ovlivňují správný chod lokalizačních systémů. Druhým hlavním cílem je vytvoření mobilní aplikace, jež bude v budově schopna lokalizovat mobilní zařízení, resp. osobu, která ho vlastní. Tato aplikace je základním elementem, v této práci navrhovaného, lokalizačního systému a je určena osobě, jež ho bude mít na starosti, tedy správci. Lokalizační systém však disponuje dalšími součástmi, kterým se však věnuje, ve své práci, kolega Jiří Pils. Z toho důvodu budu na některých místech v textu odkazovat právě na něj.

Základní struktura práce je koncipována tak, aby pokryla oba její cíle. Pominu-li úvodní první kapitole, tak ve druhé kapitole uvedu problematiku bezdrátových sítí se zaměřením na lokální bezdrátové sítě. Ve třetí kapitole naopak kompletně popíši funkci lokalizačních systémů. Toho je poté využito, jak pro analýzu parametrů stávající bezdrátové sítě, z pohledu vyhovění podmínkám umožňující lokalizaci v prostorách budov, v kapitole čtyři, tak pro návrh lokalizačního systému a vývoj mobilní aplikace v kapitole pět. Poslední kapitole tvoří závěrečné zhodnocení dosažených výsledků.

Pro názornější popis různých principů, lokalizačních metod, měření apod., se v práci objeví pojmy přijímač a vysílač. Vysílačem bude myšlen přístupový bod a přijímačem přenosné zařízení, pokud to nebude výslovně uvedeno jinak. Rád bych ale předeslal, že toto dělení Wi-Fi zařízení není principiálně správné, protože obě strany komunikačního spoje pracují ve funkci jak přijímače, tak i vysílače.

## 2. Bezdrátové datové sítě

### 2.1 Typy sítí dle rozlohy

Datová komunikace mezi zařízeními, jako je například stolní počítač, notebook nebo chytrý telefon, probíhá na základě navazování, udržování a ukončování spojení přes určitou datovou, neboli síťovou infrastrukturu. Pod pojmem infrastruktura je zde míněno propojení síťových prvků, které datovou komunikaci umožňují. Taková infrastruktura, co se týče komunikace na větší vzdálenost, může být velice složitá. To platí nejen o počtu prvků na trase, ale i o způsobu jejich vzájemné komunikace a jejich správě. Z toho důvodu je zde nutná dekompozice na dílčí části, které budou navzájem nezávislé, ale stále budou tvořit funkční celek. Nebudu se zde však zabývat mobilními sítěmi, technologií WiMAX apod., ale sítěmi, jejichž primárním smyslem je komunikace mezi počítači. V současnosti už to však není jen o počítačích, ale také o chytrých telefonech, tabletech apod. Tyto sítě se tedy dají rozdělit podle své geografické rozlehlosti, na tzv. osobní PAN, lokální LAN, metropolitní MAN a rozlehlé WAN sítě. Tento výčet zahrnuje pouze sítě, jejímž médiem je propojovací kabel. Pro problematiku této práce jsou však důležitější jejich modifikace v podobě bezdrátových řešení, jako jsou WPAN, WMAN, WWAN a zejména WLAN, na níž bude zaměřena větší pozornost. Následující řádky jsou věnovány stručnému popisu jednotlivých technologií.

PAN (WPAN): *Personal Area Network*, resp. *Wireless Personal Area Network*. Pod českým označením jako *osobní síť*, resp. *bezdrátová os. síť*. Tato síť může být tvořena například zařízeními naší osobní potřeby. Týká se to notebooků, mobilních telefonů, kamer, fotoaparátů apod. Tato komunikace tedy probíhá na vzdálenosti v řádech několika metrů a může být přiblížena na situaci, kde se například mobilní telefon propojí s notebookem. V případě propojení pomocí USB kabelu se jedná o síť PAN, v případě propojení pomocí technologie Bluetooth, tedy bezdrátovým přenosem, se jedná o síť WPAN [1]. Mezi další možnosti propojení se řadí například FireWire, Thunderbolt, IrDA či bezdrátové USB.

LAN (WLAN): *Local Area Network*, resp. *Wireless LAN*. Tedy *lokální síť* a její bezdrátová varianta. S tímto typem sítě se dnes setkáme téměř na každém rohu. Zejména v podobě bezdrátového přístupu k internetu, a to jak z internetových kaváren či firemních sítí, tak hlavně jako produkt bezdrátových sítí v domácnostech. Z toho vyplývá, že lokální sítě jsou vlastněny domácnostmi nebo různými organizacemi, jako jsou firmy, školy, úřady, nákupní centra apod. Jedná se tedy o sítě, které pokrývají spotřebu koncových zákazníků tím, že jsou připojené k nadřazeným sítím internetového poskytovatele, tzv. ISP (*Internet Service Provider*). Zároveň mohou LAN sítě tvořit síťovou infrastrukturu konkrétní organizace, díky nimž může dojít například k nasazení datových uložišť, poštovních serverů či firemní telefonie.

MAN (WMAN): *Metropolitan Area Network*, resp. *Wireless MAN*. Je síť, která je rozlohou větší než LAN, ale menší než WAN síť. V překladu se jedná o tzv. *metropolitní síť*, která se především využívá ve větších městech. Vzniká za účelem rozvoje městských částí, buď jako nástupce historických rozvodů telefonních linek [1], nebo budováním nových linek na bázi optického vlákna. Metropolitní síť může být složena propojením menších sítí, protože vlastníkem takové sítě může být lokální poskytovatel telefonie, či internetu v dané městské části. Na tuto síť mohou být připojené lokální sítě a naopak metropolitní síť je připojena na WAN síť. Pro doplnění

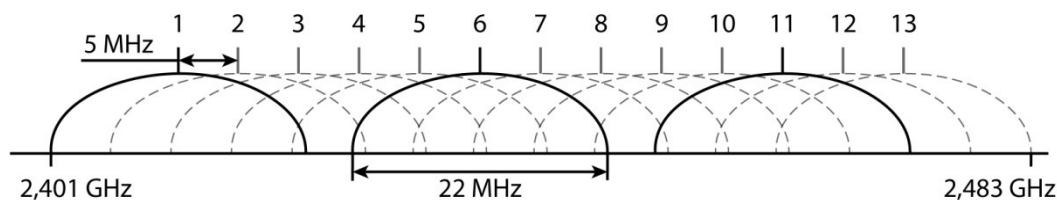
bezdrátového řešení můžeme uvést technologii FSO (*Free Space Optics*), která využívá přímé viditelnosti mezi vysílačem a přijímačem umístěnými na vyšších budovách a přenosu dat pomocí laserového paprsku.

WAN (WWAN): *Wide Area Network*, resp. *Wireless WAN*. Možno volně přeložit jako *rozlehlá síť*. Tento pojem opravdu odráží svůj význam, protože WAN sítě propojují nejen města a státy, ale i celé kontinenty. Je to tedy přenos datové komunikace na vzdálenosti od jednotek do desítek tisíc kilometrů a oprávněně se může považovat za páteřní síť pro veškerou komunikaci. Je proto důležité, aby byla dosažena co největší propustnost takové páteře. Z těchto důvodů se dnes využívají optická vlákna, která již dosahují řádově větších přenosových rychlostí než metalická vedení. Co se týče bezdrátových WAN, tak mezi ně patří technologie jako např. WiMAX, která však může být využita i pro WMAN. Jako další mohou být zmíněny mobilní technologie LTE či UMTS.

## 2.2 Technologie Wi-Fi

Tato zkratka [2] vychází z náhody, jež složila dohromady pojem *wireless* (bezdrátový) a zkratku *hi-fi* (*high fidelity*, čili vysoká přesnost). Tato náhoda však nic nemění na tom, že je to dnes asi nejrozšířenější pojem v oboru bezdrátových komunikací, který vyjadřuje technologii pro bezdrátovou komunikaci zařízení. Tato technologie je standardizovaná organizací *Institute of Electrical and Electronics Engineers*, známější spíše pod zkratkou IEEE, odkud také pochází standardy pro jednotlivé vývojové stupně Wi-Fi technologie, které se postupem času nasazují a zkvalitňují komunikaci zařízení. Komunikace definovaná těmito standardy je používána ve volných pásmech neboli pásmech nepodléhajících licenčním ujednáním. Taková pásma se označují jako pásma ISM (*Industrial, Scientific and Medical*) a slouží tedy pro volné použití v průmyslu, vědě a lékařství. O regulaci pásem ISM se stará organizace ITU (*International Telecommunication Union*), a to konkrétně její sektor pro správu radiokomunikací ITU-R. Jednotlivých ISM pásem je velké množství [3], avšak pro potřebu této práce budou stačit pásma dvě.

Prvním a tím nejrozšířenějším přenosovým pásmem je interval 2,4 až 2,5 GHz. Jeho rozšířenost je dána tím, že je podporován drtivou většinou komunikačních prvků. Tedy jak na straně vysílačů, čili přístupových bodů, tak na straně přijímačů, tedy notebooků, tabletů, mobilních telefonů či chytrých televizí. Díky vyššímu povolenému vyzářenému výkonu 100 mW [4], má také větší dosah než pásmo následující. To je v rozmezí 5,725 až 5,875 GHz a jeho výkon je omezen na 25 mW [4]. Tedy, i když je dosažena větší přenosová rychlost, tak naopak dosah je omezen. Naproti tomu má 5 GHz pásmo výhodu v tom, že není takovým způsobem rádiově zarušené oproti pásmu se středním kmitočtem 2,45 GHz.



Obr. 2.1: Uspořádání kanálů dle IEEE 802.11b [4, upraveno]

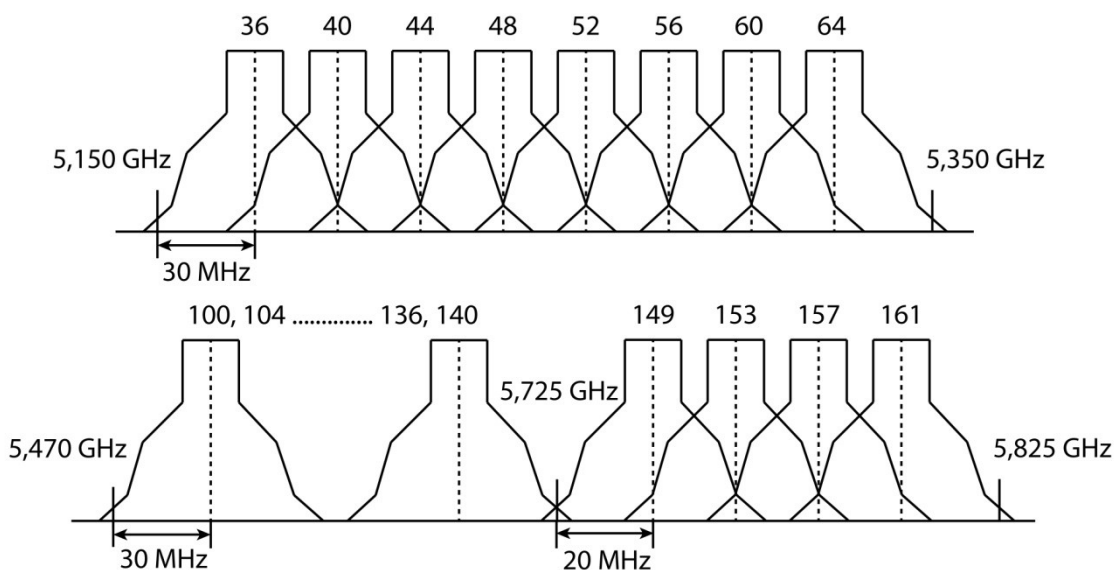
Tab. 2.1: Wi-Fi standardy [5, upraveno]

Wi-Fi standardy podle IEEE				
Standard	Rok vydání	Modulace	Pásmo [GHz]	Maximální rychlost [Mbit/ s]
IEEE 802.11	1997	DSSS, FHSS	2,4 <sup>1)</sup>	2
IEEE 802.11a	1999	OFDM	5 <sup>2)</sup>	54
IEEE 802.11b	1999	DSSS	2,4 <sup>1)</sup>	11
IEEE 802.11g	2003	OFDM	2,4 <sup>1)</sup>	54
IEEE 802.11n	2009	MIMO OFDM	2,4 <sup>1)</sup> ; 5 <sup>2)</sup>	600
IEEE 802.11ac	2013	MU-MIMO OFDM	5 <sup>2)</sup>	1000
IEEE 802.11ad	2014	OFDM	2,4 <sup>1)</sup> ; 5 <sup>2)</sup> ; 60 <sup>3)</sup>	7000

<sup>1)</sup> Tato hodnota se všeobecně prosadila, jako snadný identifikátor pro určení podporovaného pásma u spotřebního zboží. Pásmo jako takové má rozmezí frekvencí 2,401 - 2,483 GHz, kde na jeho počátku začíná první kanál a na jeho konci končí kanál třináctý (viz Obr. 2.1). Každý kanál má šířku 22 MHz a střední frekvence jednotlivých kanálů jsou od sebe vzdáleny 5 MHz. Kanál o šířce 22 MHz však spadá jen do standardu IEEE 802.11b, kdežto pro všechny ostatní se využívá šířka 20 MHz. Jelikož jsou tedy šířky kanálů větší, než jejich rozestupy středních frekvencí, tak dochází ke vzájemnému prolínání jednotlivých kanálů, a tedy k jejich interferenci. Podstatným důsledkem interference je snížení přenosové rychlosti Wi-Fi spoje, čili při návrhu bezdrátové sítě je snaha o to, využívat nepřekrývající se kanály. To znamená například kombinaci kanálů 1, 6, 11, nebo 2, 7, 12, nebo 3, 8, 13, což jsou tři možné a jediné kombinace s využitím maximální šířky bezlicenčního pásma ISM. Samozřejmě, můžeme využít i kombinaci různých dvou kanálů, které splňují podmínku ohledně vzájemného překrývání. Další možností, jak zvýšit spolehlivost, či propustnost Wi-Fi připojení je, že u standardu IEEE 802.11n můžeme využít větší šířky kanálu, a to ze stávajících 20 MHz na 40 MHz.

Co se týče využití kanálů, tak pro různé země platí i různé standardy ohledně počtu používaných kanálů. Mohou se tedy objevit takové případy, že kanálů bude jen 11 [6], to platí pro standardy organizace FCC (*Federal Communications Commission*), čili pro USA. Naopak v Japonsku používají maximální počet kanálů čili 14 [6]. My však zůstaneme u evropských zvyklostí a to dané organizací ETSI (*European Telecommunications Standards Institute*), tedy 13 kanálů.

<sup>2)</sup> Hodnota opět trochu zavádějící. Jak bylo zmíněno výše, tak ISM pásmo pro tyto frekvence je v rozmezí 5,725 - 5,875 GHz, kde máme omezený vyzářený výkon na 25 mW. Zajímavé je však to, že pro bezdrátový přenos v 5 GHz pásmu se používají i pásma, která jsou mimo vyhrazenou definici dle ISM. Prvním pásmem je interval frekvencí 5,150 - 5,350 GHz [7]. Ten se používá jen uvnitř budov a jeho omezení, co se týče výkonu, je 200 mW. Druhým pásmem je 5,470 - 5,725 GHz [7], které není omezeno místem jeho využívání, ale pouze vyzářeným výkonem 1 W. Jak už bylo zmíněno dříve, tak 5 GHz pásma mají menší dosah, což však mohou vynahradit svým počtem nepřekrývajících se kanálů. Těch je v celém intervalu, tedy 5,150 - 5,825 GHz, hned 23, a pokud tento interval zredukujeme na pokrytí pouze vnitřních prostor, tedy na pásmo 5,150 - 5,725 GHz, tak jich dostaneme 19. Konkrétnější pohled na tuto problematiku je na Obr. 2.2.



Obr. 2.2: Přenosová pásma pro 5 - 6 GHz [1, str. 71, upraveno]

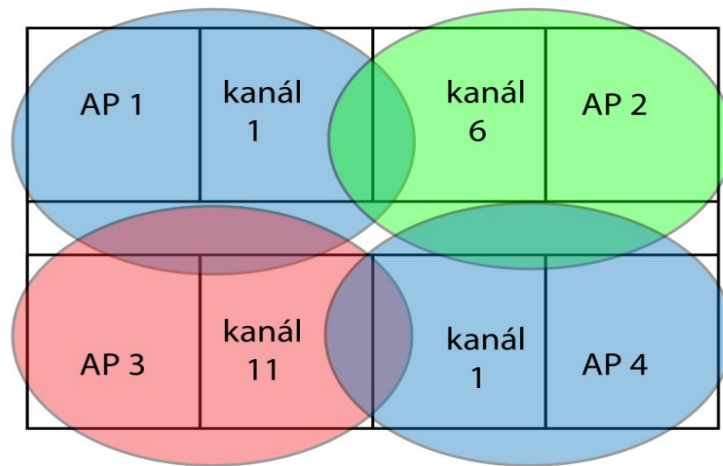
<sup>3)</sup> Nejnovější standard pro bezdrátový přenos rozšiřuje dosavadní pásma o další, a to pásmo okolo 60 GHz. Tento nový standard se občas nazývá podle jeho tvůrčí organizace WiGig, protože je schopen dosahovat přenosových rychlostí v řádu jednotek Gbit/s. Údajně až 7 Gbit/s, kde jeden kanál by měl být široký 2,16 GHz [8]. Co se využitého pásma na území Evropy týče, tak jeho hodnota by se měla pohybovat v rozmezí 57 - 66 GHz [9].

### 2.3 Bezdrátové lokální sítě

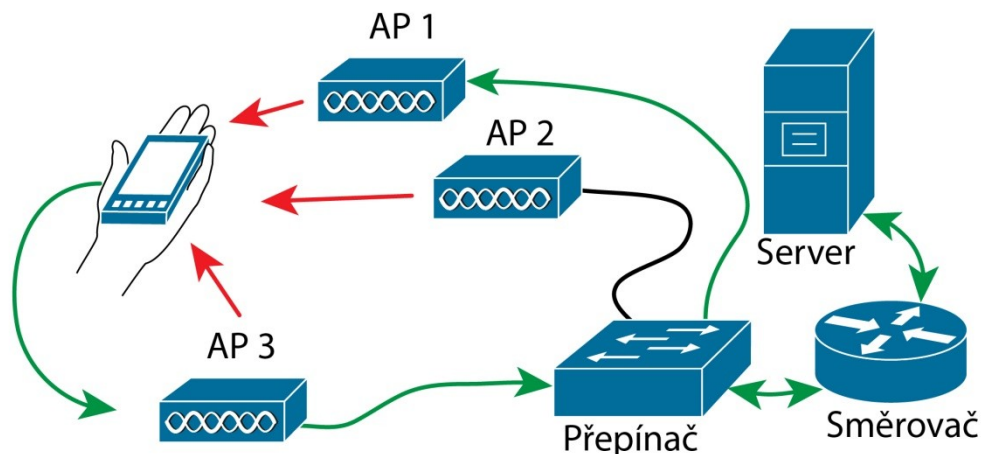
Co to vlastně bezdrátová lokální síť WLAN je, už bylo předesláno dříve. Nyní je čas, se na tento typ sítě podívat z bližšího hlediska. Základním stavebním prvkem této sítě je přístupový bod AP (*Access Point*), někdy jednoduše označován jako vysílač, který pokrývá danou oblast Wi-Fi signálem. V dnešní době je asi nejrozšířenější standard IEEE 802.11n, který umožňuje dosahovat přijatelných rychlostí na rozumnou vzdálenost pro pásmo 2,45 GHz, či 5 GHz. Zůstaneme však u pásma na nižších frekvencích, protože to je podporováno velkým množstvím zařízení. Abychom pokryli tímto signálem nějaké patro, je nutno provést plánování. Ve zkratce se jedná o to, abychom patro pokryli co nejmenším počtem AP a přitom dosáhli, co největšího a pokud možno spolehlivého, pokrytí. Samozřejmě, že se plánování odvíjí od požadavků na budoucí síť, proto například technologie RTLS (*Real-Time Location System*), vyžaduje více přístupových bodů, než vyžaduje například pokrytí kanceláře pro připojení notebooku k firemní síti. Modelový příklad je znázorněn na Obr. 2.3, který demonstruje způsob, jakým by se měl ubírat návrh pokrytí například celého patra kanceláří. Je důležité si uvědomit, že máme k dispozici šířku pásma, kde se však určité jednotlivé kanály překrývají, jak bylo popsáno dříve. Proto jednotlivé sousední zóny (kde zóna je pokrytá oblast signálem od jednoho AP) musejí být nastaveny na kanály, které jsou od sebe dostatečně vzdáleny tak, aby nedocházelo ke zbytečnému snížení kvality signálu.

Poté, co máme namodelované, rozmístěné a v reálných podmínkách analyzované pokrytí, tak můžeme dané vysílače připojit do stávající síťové infrastruktury, jež obsahuje další síťové prvky. RTLS systémy závisí zejména na síťových prvcích, jako je

například chytrý telefon, AP, přepínač (switch), směrovač (router) a server. Pro objasnění jejich vzájemných vztahů a principů slouží *Obr. 2.4*, jehož smysl popíše základním cílem lokalizačních systémů, a to stanovit na mapě místo, kde se právě osoba nachází. Proces začíná v telefonu, kde kupříkladu ztracený uživatel poklepe na tlačítko „Lokalizovat“. Prvním krokem je získávání informací o dostupných přístupových bodech a jejich silách signálu. Ve druhém kroku se získaná data posílají na analýzu přes libovolné AP. Jelikož jsou v dané aplikaci telefonu nakonfigurovány parametry koncového prvku, který bude provádět analýzu dat, tak jsou data v AP doplněna právě o tyto parametry (IP adresa, resp. MAC adresa, port, apod.). Ty poslouží přepínači a následně směrovači, k rozhodnutí, kterým směrem mají být data zaslána, aby dorazila na konkrétní databázový server. Třetím krokem je analýza získaných dat. Při uvažování lokalizační metody na bázi RSSI hodnot, získaná data se porovnávají s databází vzorků, která definuje jednotlivé body prostoru. Výsledkem porovnání je vypočtená pozice. Tato informace se přenáší zpět, přes jednotlivé prvky sítě, do chytrého telefonu, který už jen zobrazí výslednou polohu.



*Obr. 2.3: Příklad návrhu pokrytí pro síť WLAN*



*Obr. 2.4: Princip WLAN sítě při lokalizaci zařízení*

## 3. Lokalizační systémy

### 3.1 Vymezení pojmu RTLS

Zkratka RTLS už byla jednou v textu použita. Nyní se však budu snažit o prohloubení této problematiky, jež je jedním ze stavebních kamenů této práce. Pojem RTLS vyjadřuje anglické označení pro *Real-Time Location System*, což volně přeloženo znamená „Systém pro lokalizaci v reálném čase“. Mezi nejznámější takové systémy patří zejména GPS. Jeho princip je závislý na několika satelitech, kde každý bod na Zemi je vždy pokryt nejméně třemi satelity a pomocí GPS přijímače se zobrazí vlastní poloha. To je idea, která však funguje jen na přímou viditelnost přijímače a satelitů. Pokud bychom chtěli použít GPS přijímač uvnitř budovy, přesnost by se výrazně snížila působením překážek v cestě signálu mezi vysílačem a přijímačem. Překážkami rozumíme například střechu budovy, okna či vlastní interiér. Pro vyřešení tohoto problému se vyvinuly způsoby, jak využít stávajících technologií, díky nimž bude lokalizace v budovách skutečností. Pro tyto systémy se používá pojem *Indoor localization*, naopak pro lokalizaci venkovní (např. GPS) se používá pojem *Outdoor localization*. Příklady technologií jsou: Bluetooth, GPS, RFID, Wi-Fi, ZigBee, digitální TV, ultrazvuk či infračervené záření.

Hlavním měřítkem lokalizačního systému je jeho přesnost, která závisí na tzv. stupni lokalizace. Nejprve je však nutno definovat přesnost RTLS, ta uvádí míru toho, jak se liší odhad pozice určené RTLS technologií a pozice, kterou zaujímáme. Může se vyjadřovat procentuálně, a to jako podíl počtu správně odhadnutých pozic a celkového počtu měření. Naopak stupeň lokalizace udává úroveň preciznosti RTLS technologie. Více následující výčet [10].

- a) *lokalizace na základě zóny*: v situacích, kdy se identifikátor dostane do určité zóny (př. pacient převezen do části budovy A)
- b) *lokalizace na úrovni pokoje*: odhadnutá poloha identifikátoru je definována pouze tím, ve kterém pokoji se nachází (př. pokoj 112)
- c) *lokalizace na úrovni části pokoje*: odhadnutá poloha identifikátoru uvádí, ve které části pokoje se nachází (př. pacient u okna v pokoji 112)
- d) *lokalizace na úrovni průchodu*: odhadnutá poloha identifikátoru je dána jeho pohybem kolem přijímače (př. pacient vyšel z pokoje 112)
- e) *lokalizace vztažená k něčemu*: odhadnutá poloha identifikátoru je podle polohy jiného identifikátoru (př. pacient je daleko od své postele)
- f) *lokalizace precizní*: odhadnutá poloha identifikátoru je dána konkrétním údajem (př. pacient je na pozici  $x, y$ )

Z tohoto výčtu tedy vyplývá, že přesnost RTLS technologie je daná použitým stupněm lokalizace, protože například nemůžeme chtít na metry přesně určit, kde je pacient, když nemocnice používá lokalizaci na základě zón. Záměrně bylo uvedeno jedno využití, spíše jedno ze základních využití, a to nemocniční prostory. To jsou místa, kde je často velká penetrace lidí, kteří mohou být ohroženi na životech, a tedy místa závislá na neustálém dohledu. RTLS technologie se zde mohou zhostit řady úkolů. Mohou samozřejmě sledovat pohyb a stav pacientů, kteří jsou například na cestě z toalet, a kteří také mohou kdykoli ztratit vědomí, aniž by si toho někdo všiml. Na to může navazovat další využití, které využívá nějakého zobrazovacího zařízení, jež upozorní a oznámí polohu pacienta v tísni lékaři. A jako posledním článkem řetězu je lokalizace pacientova

voziky s transfuzí, jejíž kanylu si omylem na toaletě strhl. Tento příklad by byl možná reálně i nepravděpodobný, ale to nic nemění na tom, že k jeho vyřešení vedly tři kroky RTLS technologie a vyplývá z něj, že nejčastějším předmětem lokalizace jsou osoby a věci. Dalším místem pro nasazení mohou být různé mezisklady, kde je vysoká náročnost na rychlou organizaci předmětů apod.

## 3.2 Součásti systému

Základním cílem RTLS technologií je tedy lokalizovat osoby a předměty. Způsob lokalizace jako takový, se dá rozdělit na dva typy. Buď je využita lokalizace na vyžádání. To znamená, že určitá osoba chce být lokalizována na mapě ve svém příručním zařízení, nebo naopak někdo jiný chce lokalizovat danou osobu. To byl způsob na vyžádání. Druhým typem je lokalizace v reálném čase. Tam už není nutné zasílat požadavek, protože daná osoba je neustále pod dohledem. To samé platí samozřejmě i pro předměty.

V předešlé podkapitole je zmíněn jeden pojem. Tím je výraz *identifikátor*. V anglické literatuře se sice setkáme, pro tento druh zařízení, s označením *tag* (štítek či visačka), avšak v této práci budu používat *identifikátor*, protože může označovat různorodá zařízení, která však mají stejný účel. Nicméně, identifikátor je základní stavební jednotka všech lokalizačních systémů. Je to totiž malé bezdrátové zařízení, které je lokalizováno systémem. Proto, abychom mohli někoho nebo něco lokalizovat, musíme daný subjekt tímto zařízením opatřit. Identifikátory mohou nabývat různých rozměrů, či velikostí. Tedy jak větších rozměrů, pro dobrou montáž na lokalizaci předmětů, tak i menších, jako řemínky na zápěstí osob. Technologii využitou pro bezdrátové identifikátory můžeme rozdělit do tří skupin [10]. První jsou pasivní identifikátory. Ty jsou význačné tím, že nemají vestavěnou baterii, takže samy o sobě nemohou generovat radiové vysílání. Mezi tato zařízení patří zejména RFID identifikátory. Aby mohly být lokalizovány, tak musí být vystaveny rádiovému vysílání. Využívá se zde tzv. *backscatteringu*, kde se dopadající rádiové záření využije pro vyvolání odražené rádiové vlny, která obsahuje informaci o daném identifikátoru. Proto jsou RFID identifikátory daleko levnější a menší, než další dva typy. Naopak disponují krátkým dosahem, což pro velké prostory není ideální vlastnost. Druhým typem jsou napůl pasivní identifikátory, které se od pasivních liší pouze tím, že obsahují baterii. Tu však nevyužívají pro generování signálu, ale například pro zjišťování teploty okolí a udržování této hodnoty v paměti. Posledním, a také nejpoužívanějším, jsou aktivní identifikátory. Ty obsahují baterii, která je využívána pro generování signálu, takže může být dosaženo toho, že po určitých intervalech se bude identifikátor sám hlásit a tím udávat svoji polohu.

Až doposud byl identifikátor líčen jako nějaké proprietární zařízení. Nástupem a rozvojem chytrých telefonů, tabletů a konec konců i notebooků, se také rozvíjí možnosti lokalizace právě pro tato zařízení. Stačí tedy dovybavit zařízení lokalizačním klientem (v podobě aplikace) a vznikne víceúčelový identifikátor. Jako další výhodu pro toto řešení by mohlo být i to, že se nemusí nakupovat množství identifikátorů, které by plnily ten samý účel. Samozřejmě závisí na situaci, pro jaký účel chceme identifikátory nasadit. Určitě by nebylo například vhodné využívat mobilní telefony pacientů, zde se musí využít identifikátorů, speciálně navržených pro tento účel. Opačným případem by však mohla být lokalizace, která bude využita jako součást navigace v obchodním centru. Abychom se ale vrátili k základním prvkům RTLS technologie. Druhým takovým je dohledový software [10]. Jeho jediným úkolem je poskytovat vizuální



znázornění polohy dané osoby nebo zařízení. Tento software může být obsluhován buď z určitého dohledového centra externí osobou, anebo může být součástí chytrého telefonu, či tabletu v podobě mobilní aplikace, jež využívá lokalizovaná osoba. Posledním členem, třech základních prvků RTLS technologie, je tzv. *location engine* [10]. V překladu jako lokalizační stroj. Jeho důležitým úkolem je určit výslednou polohu z informací, které jsou získávány přímo z lokalizovaného identifikátoru. Lokalizační stroj však není doslovné označení pro něco hmotného, ale jedná se o software, který využitím předem definovaných postupů, určí výslednou polohu. Zpravidla se instaluje na servery dané organizace, kam je směřována všechna komunikace pro lokalizaci určená a odkud je také možno zasílat výsledky po celé síťové infrastruktuře. Tedy jak do dohledového centra, tak i do chytrého telefonu. Je to tedy prvek, který dotváří RTLS technologii složenou z prvků: identifikátor, dohledový software a lokalizační stroj. To vše je implementováno do stávající WLAN sítě organizace, která umožňuje součinnost těchto prvků.

### 3.3 Techniky získávání dat

V předchozí podkapitole byly vysvětleny prvky tvořící lokalizační systémy. Nyní se však zaměříme na prvek poslední, jímž je lokalizační stroj, protože stále nebyl uveden způsob, jakým se určí výsledná poloha identifikátoru. Celý proces je rozdělen na dvě dílčí části [10, str. 38]. První je využití vybrané techniky pro získání potřebných dat, která definují vzájemný vztah mezi identifikátorem a přístupovým bodem WLAN sítě. Mezi data patří zejména doba trvání, za kterou signál dorazí z vysílače do přijímače, tedy z identifikátoru do AP, resp. z AP do identifikátoru. Další veličinou je vzdálenost mezi těmito dvěma zařízeními, která lze určit právě z doby trvání šíření signálu. Poslední často využívanou, je síla signálu, udávaná v jednotce dBm, jež je v místě identifikátoru, tedy v určité vzdálenosti od AP. Jak je vidno, tak zmíněné veličiny jsou na sobě závislé. Jejich závislosti se dají popsat následujícími vztahy pro výpočet vzdálenosti  $D$  a pro ztráty šířením  $L_p$ .

Vzdálenost  $D$  při šíření elektromagnetické vlny prostorem za čas  $t$ , při rychlosti šíření vlny  $c$  ( $c = 3 \times 10^8$  m/s; rychlost šíření světla):

$$D = c \times t \quad [m] \quad (3.1)$$

Empirický model pro ztráty (útlum) šířením  $L_p$  vlny v závislosti na vzdálenosti  $d$  od vysílače [11, (8.2)]:

$$L_p(d) = L_0(d_0) + 10 \times n \times \log\left(\frac{d}{d_0}\right) \quad [dB] \quad (3.2)$$

kde ztráty volným prostorem  $L_{FSL}$  [11, (3.8), upraveno]:

$$L_{FSL} = L_0(d_0) = 10 \times \log\left(\frac{4\pi d}{\lambda}\right)^2 \quad [dB] \quad (3.3)$$

$d_0$  - referenční vzdálenost pro pikobuňku (prostor budovy) je 1 metr

$n$  - útlumový činitel, v budově na přímou viditelnost 1,6 - 1,8 a na nepřímou 4 - 6

$\lambda$  - vlnová délka elmag. vlny

Poté, co byly zjištěny jednotlivé veličiny (vzdálenost apod.), přichází na řadu druhá část procesu. Ta je zodpovědná za zpracování získaných hodnot pomocí vybraného matematického algoritmu. Díky němu získáme výsledné souřadnice určující polohu

identifikátoru. Více informací o jednotlivých typech algoritmů je obsaženo v následující podkapitole. Zde budou naopak vylíčeny principy nejčastěji využívaných technik [10, str. 38] pro získávání dat.

#### *TOA (Time of Arrival)*

Tato technika je založená na měření doby, neboli zpoždění signálu, za kterou signál urazí vzdálenost mezi vysílačem (identifikátor) a přijímačem (AP). Doba je počítána od určitého časového okamžiku  $t_0$ , který je stejný pro obě účastnické strany. To znamená, že je velice důležité, aby obě strany byly kvalitně synchronizovány po celou dobu běhu RTLS technologie. Tato technika je tedy velice náchylná na šíření signálu, což pro přímou viditelnost vysílače a přijímače není velká potíž. Potíž nastává tehdy, když se v dráze signálu objeví překážka, například procházející osoba. Stačí jen krátký okamžik a signál změní dráhu svého šíření, což při prodloužení časového intervalu o pouhých 100 nanosekund změní vypočtenou dráhu o 30 metrů. Nicméně, výsledná vzdálenost je poté v lokalizačním stroji vypočtena dle vztahu (3.1) a slouží jako vstupní hodnota pro algoritmus výpočtu pozice, nazývaného *trilaterace*.

#### *AOA (Angle of Arrival)*

Opět zajímavá technika, která však není závislá na časových odchylkách. Jejím principem je určování směru, ze kterého signál přichází. To znamená, že AP má definován určitý referenční směr a díky jeho směrově citlivé anténě poté dokáže určit úhel příchodu signálu. Ten je určen v lokalizačním stroji a slouží jako vstupní parametr pro algoritmus s názvem *triangulace*. Bohužel ani tato technika není bezchybná. Jelikož se jedná o lokalizaci uvnitř budov, je téměř jisté, že se signál šíří více cestami. Za předpokladu, že nenastala situace, kde identifikátor a AP splňují podmínku přímé viditelnosti, může docházet k tomu, že dopadající signál, směrem k anténě, je vychýlen jen o pár stupňů. Což ve výsledku může způsobit chybu v řádu jednotek až desítek metrů. Z části můžeme zlepšit přesnost nasazením kvalitnějších anténních polí či hned několika antén, avšak tato cesta je zase ekonomicky náročná.

#### *TDOA (Time Distance of Arrival)*

Obdobně, jako u TOA, je tato technika závislá na měření časových okamžiků. Je založená na tom, že identifikátor vyšle signál, který obdrží všechny AP, avšak v různých časových okamžicích. Každé AP je tedy definováno časem příchodu signálu. V lokalizačním stroji poté dojde ke zjištění rozdílů, mezi těmito okamžiky, a podle známých souřadnic pozic jednotlivých AP se určí výsledná poloha identifikátoru. Stejně jako u TOA, tak i zde se používá algoritmus *trilaterace*, který však nepracuje na bázi průniku kružnic, ale hyperbol. Co se týče vlivu prostředí, opět se setkáváme s vícecestným šířením, které ovlivňuje čas šíření signálu, a tedy určitou chybu. Na rozdíl od TOA tato technika není tolik náročná na synchronizaci. Je to způsobeno tím, že jsou synchronizovány pouze AP, z jejichž časů se stanovují jednotlivá zpoždění, proto není nutné využívat počáteční časový okamžik  $t_0$  i pro identifikátor.

#### *RSSI (Received Signal Strength Indicator)*

V porovnání s předešlými je tato technika založena na zcela jiném principu. Každý vysílač je definován svým maximálním vysílacím výkonem. Pro přístupové body využívající Wi-Fi standardy je na území Evropy maximální povolený výkon 100 mW [4]. Cílem je tedy změřit tuto hodnotu síly signálu od AP, jež by popisovala vzdálenost mezi identifikátorem a AP. Jelikož síla signálu díky útlumu se vzdáleností klesá, můžeme takto dostat vrstevnicový popis okolí AP, jež by udával místa se stejnou silou

signálu. Tzv. hodnoty RSSI, jež vyjadřují velikost síly signálu v daném místě, můžeme změřit spektrálním analyzátozem, anebo můžeme využít mobilní, či počítačové programy. Tato technika tedy nevyužívá hodnoty získané pomocí AP (jako u TOA, TDOA a AOA), ale hodnoty získané z mobilního zařízení. Druhým způsobem jak získat tyto hodnoty je tzv. predikce. Ta je dosti často také využita při lokalizaci, kdy napomáhá ke zkvalitnění informací, které popisují dané prostory, popřípadě pokud známe výkon antény a výkon v místě lokalizace, můžeme využít vztahů (3.2), (3.3) a (3.5), abychom určili vzdálenost identifikátoru od AP. Pro tuto predikci můžeme použít vztah, který popisuje výkonovou bilanci spoje, v našem případě na straně vysílače identifikátor a na straně přijímače AP. Výkonová bilance [12, (2.12)] je dána jako výkon vyzářený vysílačem  $P_V$ , který je snížený útlumem prostředí  $L$ , jež odpovídá vztahu (3.2). Pro lepší názornost zanedbáme zisky antén  $G_V, G_P$  (vztah (3.4)) na obou stranách spoje, tedy přijímače i vysílače, protože se pohybují v řádu jednotek dB.

$$P_p = P_V + G_V + G_P - L \quad [dBm, dBm, dB, dB, dB] \quad (3.4)$$

Vztah tedy můžeme upravit do tvaru (3.5), ze kterého můžeme určit přijatý výkon  $P_p$ , který odpovídá hodnotě RSSI, jež označuje sílu přijatého signálu, ale i aktuálně probíranou techniku pro získávání dat.

$$P_p = P_V - L \quad [dBm, dBm, dB, dB, dB] \quad (3.5)$$

Pro úplnost ještě doplníme vztah (3.6) [12, (2.8)], který udává převod mezi výkonem vysílače (přijímače), v našem případě 100 mW, na jednotku dBm. To je označení pro jednotku decibel dB vztaženého k výkonu 1 mW.

$$P_{dBm} = 10 \times \log\left(\frac{P_W}{10^{-3}}\right) \quad [dBm] \quad (3.6)$$

kde  $P_W$  je výkon v jednotce watt [W].

Bohužel i tato technika není dokonalá. Je to způsobeno faktem, že měříme velikost síly signálu pomocí bezdrátového adaptéru. Ten je však v každém přístroji kalibrován trochu jiným způsobem, a tak dochází k problému, kdy například na jednom místě změříme dvě různé hodnoty RSSI, jejichž rozdíl se pohybuje v řádech jednotek dBm. Dalším faktorem, už v této kapitole zmíněný, je vícecestné šíření signálu. Takže hodnoty RSSI se mohou například lišit i podle denní doby, tzn., že v pracovní době je patro zaplněno daleko více lidmi, které působí na signál jako přírodní překážky. Tyto překážky však nejsou statické, a o to k větším odchýlkám dochází.

### 3.4 Algoritmy výpočtu pozice

Ze získaných dat, z předešlé kapitoly, už jen zbývá určit konečnou pozici zařízení. K tomu slouží následující algoritmy [10, str. 40], jež jsou též součástí lokalizačního stroje.

#### *Trilaterace*

Někdy nesprávně zaměňována za triangulaci. Základem algoritmu je výpočet pozice, který vznikne průnikem třech kružnic. Pokud tedy máme tři přístupové body a jeden identifikátor, tak pomocí techniky TOA, TDOA případně i RSSI, stanovíme jednotlivé vzdálenosti mezi ním a třemi AP. Je důležité, aby byly minimálně tři, jinak by nebylo možné tento algoritmus využít. Tři vzdálenosti vztažené k daným AP, definují kružnici,

kteřou musíme danému AP opsat, protože nevíme přesně, v jakém směru identifikátor je. Touto metodikou opišeme všem třem AP kružnici a tím vznikne průsečík hledané pozice. Názorný příklad je uveden na Obr. 3.1. Další podmínkou dosažení výsledku je ta, abychom znali přesnou polohu jednotlivých AP, díky které můžeme použít vztah (3.7) pro výpočet polohy.

$$r_i^2 = (x - x_i)^2 + (y - y_i)^2, i = 1, 2, 3 \dots N \quad (3.7)$$

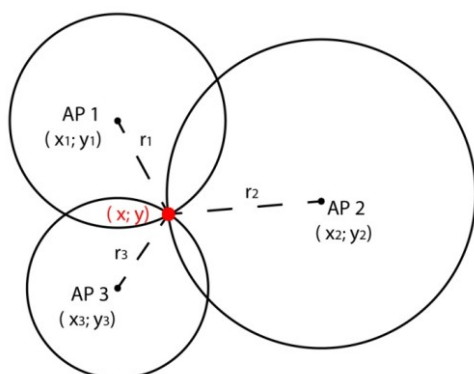
kde  $r$  je vzdálenost od AP a  $N$  je jejich počet AP.

### Triangulace

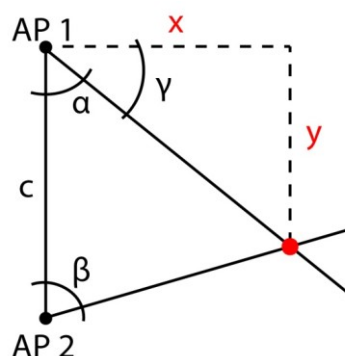
Jejím principem je výpočet polohy na základě získaných úhlů, jež zajistila technika AOA. Pro výpočet nám však budou stačit pouze dva úhly, tedy dvě AP, které jsou dány odklonem od referenčního směru přístupových bodů. K výpočtu použijeme vztah (3.8) [13], díky kterému získáme velikost kolmice na spojnici dvou AP. Tím získáme jednu souřadnici, například  $x$ . Druhá souřadnice se vypočítá dle vztahu (3.9), kde výsledná pozice je vztažena k AP 1. Více na Obr. 3.2.

$$x = \frac{c \times \tan(\alpha) \times \tan(\beta)}{\tan(\alpha) + \tan(\beta)} \quad (3.8)$$

$$y = \tan(|90^\circ - \alpha|) \times x \quad (3.9)$$



Obr. 3.1: Model trilaterace



Obr. 3.2: Model triangulace

### Nejbližší soused

V tomto případě bude popsán algoritmus, který určuje co nejpřesnější polohu díky databázi dat. Nebude zde myšlena metoda se stejnojmenným označením, která na základě RSSI hodnot od určitého počtu AP, stanoví pozici v okolí určitého AP, na základě nejvyššího RSSI. Co se týče zmíněného pojmu databáze dat. Ten souvisí s lokalizační metodou pod anglickým označením jako *Location fingerprinting*, která bude popsána později v této práci a která také bude využita pro navrhovaný systém. Pro popsání tohoto algoritmu [14] se prozatím spokojíme s tvrzením, že máme k dispozici vektor  $P$ , který obsahuje takový počet  $N$  RSSI hodnot, jež je v daném okamžiku k dispozici přístupových bodů. Dále vektor obsahuje dvě souřadnice  $x, y$ , které popisují pozici, ze které bylo měření vektoru provedeno. Takový vektor může být popsán následujícím způsobem:

$$P = (\vartheta_1, \dots, \vartheta_N, x, y) \quad (3.10)$$

Prvním krokem algoritmu je seřazení hodnot ve vektoru  $P$  sestupně, kromě  $x, y$ , které budou použity až v závěrečné fázi algoritmu, a tudíž k nim nebudeme přistupovat. Nyní

vytvoříme aktuálně naměřený vektor  $R$ , který obsahuje  $M$  RSSI hodnot z dostupných vysílačů, kterých musí být stejné množství, jako u vektoru  $P$ , čili  $M = N$ .

$$R = (\varphi_1, \dots, \varphi_N, x, y) \quad (3.11)$$

Do vektoru  $R$  však nezapisujeme souřadnice  $x, y$ . Pokud bude vektor  $R$  kratší (z pohledu RSSI hodnot) než  $P$ , dosadíme na zbylá místa hodnotu  $-100$  dBm a seřadíme ho sestupně. Pokud bude naopak delší, nejprve provedeme sestupné řazení, a až poté ho zkrátíme na stejnou délku, jako má vektor  $P$ . Nyní jsou k dispozici dva stejně velké a stejně seřazené vektory  $P, R$ . Pro ně vypočítáme tzv. Spearmanovu vzdálenost (vztah (3.12)), což je druhá mocnina Euklidovy vzdálenosti mezi dvěma vektory [14, (1)].

$$D = \sqrt{\sum_{k=1}^n (\vartheta_k - \varphi_k)^2} \quad (3.12)$$

kde  $n$  je počet prvků ve vektorech,  
 $k$  je pozice hodnoty ve vektoru.

Nyní dostaneme číslo, které udává rozdílnost těchto dvou vektorů, tedy čím menší číslo, tím jsou vektory podobnější. Může nastat situace, kdy vyjde nula, v tom případě jsou vektory stejné. Pokud bychom takto porovnávali více vektorů, tedy vektorů už dříve zjištěných s vektorem aktuálním, vzniklo by nám o to více výsledků. Cílem tohoto algoritmu je to, že výsledky seřadíme vzestupně a první číslo, zároveň nejmenší, je tzv. *nejbližší soused* pro aktuálně naměřený vektor. Posledním krokem je vypsání souřadnic z vektoru, který se stal nejbližším sousedem, což jsou hledané souřadnice identifikátoru. Může však nastat situace, kdy budeme mít více nejbližších sousedů. Kvůli těmto případům je lepší použít dodatečný filtr pro předzpracování vektorů, než budou použity pro výpočet své odlišnosti.

Zde bych mohl předeslat, že tento algoritmus bude využit v mobilní aplikaci, jejímž návrhem se budu zabývat v kapitole pět. Metoda nejbližšího souseda NN (*Nearest Neighbor*) však není jediná svého druhu. Na jejím základě, a tedy jejím vylepšením, jsou vytvořeny algoritmy KNN (*K Nearest Neighbors*) a WKNN (*Weighted K Nearest Neighbors*). Metoda KNN spočívá na principu výpočtu výsledné pozice z určitého počtu souřadnic z databáze. Jednoduše řečeno, pokud je  $k = 3$ , tak ze tří nejmenších rozdílových vektorů, tedy tří hodnot  $x$  a  $y$ , vypočítáme průměr (3.13), který značí výslednou polohu [14, (2)].

$$(x, y) = \frac{1}{K} \times \sum_{i=1}^K (x_i, y_i) \quad (3.13)$$

kde  $K$  je počet vektorů, ze kterých se určují souřadnice,

$(x, y)$  jsou výsledné souřadnice a  $(x_i, y_i)$  souřadnice  $k$ -tého nejbližšího souseda ve vektoru  $P_i$ .

Metoda WKNN je odlišná z toho pohledu, že se počítá váha, s níž se poté jednotlivé souřadnice váhují. Váha je v podstatě převrácená hodnota druhé mocniny vypočítané vzdálenosti (3.12) dvou vektorů. Výsledné souřadnice se poté určí dle vztahů (3.14), (3.15).

$$(x, y) = \sum_{i=1}^K (x_i, y_i) \times w_i \quad (3.14)$$

$$w_i = \frac{1}{D_i^2} / \sum_{j=1}^K \frac{1}{D_j^2} \quad (3.15)$$

kde  $w_i$  je váha a  $D$  je Spearmanova vzdálenost.

### 3.5 Lokalizační systém Ekahau

V této podkapitole bude představen balíček [15] lokalizačního systému od firmy Ekahau. Balíček obsahuje softwarové i hardwarové prostředky, díky nimž může být na konkrétním případě objasněna funkce lokalizačních systémů. V tomto případě se jedná o, již dříve zmíněnou, technologii RTLS. Hlavní uplatnění tohoto řešení je mířené zejména do nemocničních prostor. Zde může být využito pro lokalizaci a správu nemocničních zařízení, což napomáhá k jejich hledání nebo zabezpečení proti krádeži. Dalším využitím je řízení personálu z hlediska efektivní spolupráce lékařů a sester. Samozřejmě je tato technologie také využita pro monitorování pohybu pacientů. U náramkových identifikátorů, jež jsou vybaveny pootovostním tlačítkem, je dokonce možné přivolat rychlou pomoc. Lokalizační technologie však nemusí nutně zastávat jen funkci monitorování pozice osob či předmětů. Namísto identifikátorů se mohou využít různé bezdrátové senzory, kde zmiňovaný systém Ekahau podporuje například senzory teploty i vlhkosti.

Základní struktura Ekahau RTLS technologie je rozdělena do čtyř součástí, které pokrývají oblasti od návrhu bezdrátové sítě až po její využívání lokalizačními procesy. Softwarové vybavení pro návrh a optimalizaci WLAN sítě se nazývá *Ekahau Site Survey*. Díky němu můžeme navrhnout bezdrátovou síť, která bude splňovat požadavky Ekahau systému, avšak můžeme ho využít i pro návrh normální sítě. Základem pro navrhování, je mít dobré podklady daného prostoru a hlavně mít daný prostor prostudován. Jedná se zde o to, že po načtení mapy prostoru se do programu zadávají konkrétní informace, které tento prostor popisují. Proto potřebujeme vědět například určitý reálný rozměr, aby bylo možné v mapě zadat její měřítko. Dalším krokem je zakreslení zdí, dveří či oken (dle vložené mapy), kde pro jednotlivé elementy můžeme specifikovat jejich tloušťku, ale hlavně jejich materiál a z toho odvozený útlum. Útlum je důležitý, protože návrh sítě je postavený na simulaci pokrytí daného prostoru, kde se využívá matematických modelů, jako například pro výkonovou bilanci (3.4). Po specifikaci měřítka a zdíva, přichází na řadu rozmístění přístupových bodů. Ty se mohou umisťovat kdekoli na mapu, čili opět je lepší, mít přehled o možnostech jejich umístění, díky osobní návštěvě prostorů. Umístěné vysílače můžeme samozřejmě blíže specifikovat, ať už standardem IEEE 802.11a/b/g/n/ac, konkrétním kanálem, anténou nebo vyzářeným výkonem. To vše má dopad na výslednou simulaci, ze které hned vidíme síly signálů v jednotlivých prostorech, hodnoty odstupe signálu od šumu, hodnoty maximálních přenosových rychlostí, dosah apod. Dle těchto informací můžeme ještě optimalizovat pokrytí před tím, než se bude fyzicky síť instalovat. Druhým základním využitím programu je tvorba tzv. *Site survey*, čili průzkum místa. Ten je možno využít pro dodatečnou optimalizaci sítě, protože díky němu získáme kompletní reálný přehled o síti tím, že projdeme celý prostor a pouhým klikáním do mapy zaznamenáme potřebná data. Ta se střádají nejen při samotném klikání, ale také v době, mezi jednotlivými kliky. Posléze jsou tato data interpolována a krom možnosti optimalizace sítě, tvoří také kvalitní (záleží na způsobu pečlivosti tvorby) základ pro následnou lokalizaci.

Je to z toho důvodu, že systém Ekahau je založen, dle mého mínění, na principu tzv. *fingerprinting* (sběr otisků), který je ještě doplněn o částečnou predikci. Vyplývá to ze způsobu, jakým se sbírají data, určená pro využití lokalizace. Tedy prvním krokem se provede již zmíněný průzkum prostoru, který sbírá konkrétní data pro konkrétní pozice na mapě. Poté je nutné definovat otevřené prostory, čili místnosti a chodby, a cesty, která tato místa propojují. Tím dochází k dopočítání dalšího množství pravděpodobných

hodnot, například u definovaných cest, kde se mohou dopočítávat hodnoty do určité šíře cesty. Zde bych ještě doplnil skutečnost, že pro provedení průzkumu není třeba definovat jednotlivé zdi, to slouží jen pro účely simulace. Definování cest a volných prostorů také napomáhá k tomu, že se může počítat pravděpodobnost, která vyjadřuje to, jestli je identifikátor na daném místě, nebo není. Kombinací predikce a sběru otisků se může výsledná přesnost lokalizace výrazně zvýšit.

Zpět však k systému Ekahau. Poté co je vytvořena zmíněným průzkumem mapa, načte se do lokalizačního stroje (podkapitola 3.2) a tam se aktivuje. Lokalizační stroj (tzv. *RTLS Controller*) je v tomto případě systémová služba běžící na pozadí, kterou můžeme konfigurovat buď konfiguračním programem, anebo k ní můžeme přistupovat přes grafické webové rozhraní. Lokalizační stroj může být spuštěn buď na serverovém zařízení, anebo přímo v dohledovém počítači. Spíše je však doporučena varianta první, už jen kvůli tomu, že servery jsou v provozu neustále a jejich napájení je zálohované. Z pohledu dozoru je však důležitější další služba, a to *Ekahau Vision*. Může i nemusí být spuštěna na stejném stroji jako lokalizační stroj, a ke které se přistupuje také přes webové rozhraní. Jejím hlavním úkolem je vizualizace jednotlivých identifikátorů na mapě, která se aktivovala na lokalizačním stroji. To znamená, že vidíme pohyb v téměř reálném čase jednotlivých osob či zařízení, která můžeme seskupovat, vytvářet k nim záznamy, dokonce zasílat textové zprávy na dané identifikátory apod.

Pro úplný popis Ekahau RTLS ještě zbývají doplnit informace o používaných identifikátorech. Pominu-li senzory pro měření teploty nebo vlhkosti, tak jsou k dispozici tři základní druhy. Všechny pracují na IEEE 802.11b/g/n. Prvním je poziční identifikátor *T301W* pro osoby, *Obr. 3.3*. Může být dovybaven náramkem pro jednoduchou aplikaci na zápěstí pacienta. Pracuje na principu zasílání dat o dostupných silách signálů buď v určitém intervalu, při pohybu, anebo po stisknutí tlačítka na jeho vnější straně. To může také vykonávat funkci žádosti o asistenci nemocničního personálu. Umožňuje lokalizaci na úrovni pokoje či místa v pokoji. Druhým identifikátorem je *A4 Asset*, *Obr. 3.4*, pro lokalizaci zařízení. Je o něco větší, kvůli vyměnitelným bateriím, jejichž životnost se však udává až pět let. Vybaven je také dvěma programovatelnými tlačítky, ale hlavně infračerveným senzorem, který je využíván ve spojení s infračerveným vysílačem pro precizní lokalizaci. Na jeho zadní straně je umístěn další senzor, který se aktivuje při oddálení od podložky, tedy slouží jako poplachový spínač, kdyby ho někdo chtěl sundat ze zařízení. Posledním identifikátorem je *B4 Badge*, *Obr. 3.5*, který navíc poskytuje přijímání textových zpráv. Je tedy vhodný pro ochranu, zaměstnance apod. Existuje však ještě jedna možnost identifikátoru, a to díky softwarové aplikaci *Positioning Client* (poziční klient), který si mohou uživatelé *Windows*, *Windows Phone*, nainstalovat a poté budou automaticky přiřazeni na seznam identifikátorů a do dohledového programu.



Obr. 3.3: T301W [15]



Obr. 3.4: A4 Asset [15]



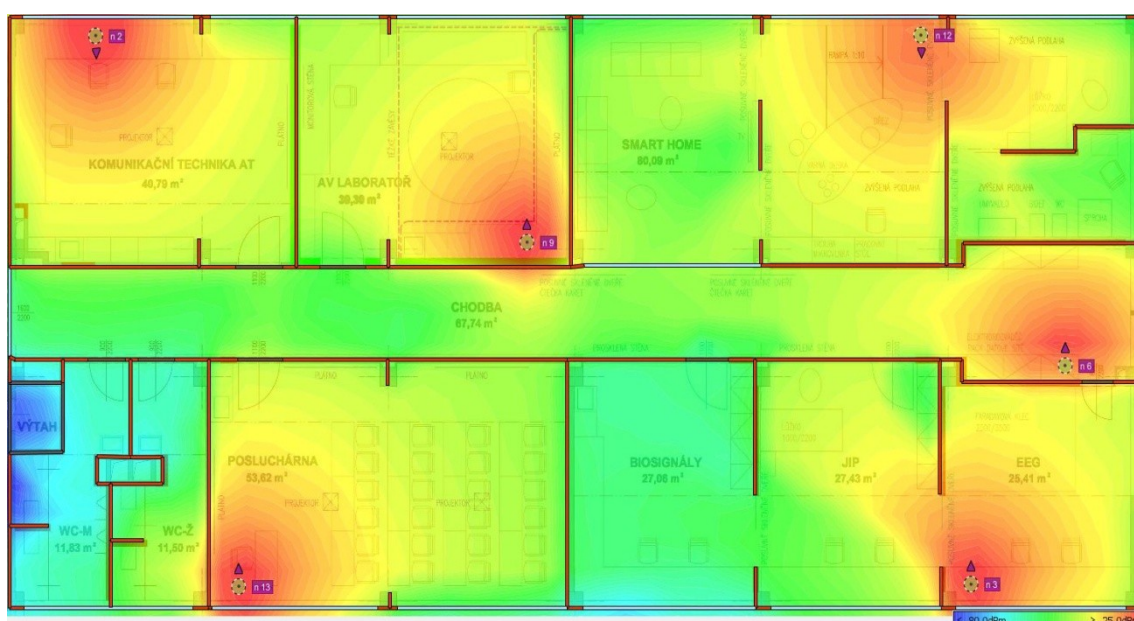
Obr. 3.5: B4 Badge [15]



## 4. Analýza pokrytí bezdrátové sítě

### 4.1 Simulace pokrytí s využitím Ekahau nástrojů

Pro účely zkoušek lokalizačního systému Ekahau RTLS, ale hlavně zkoušek návrhu vlastního systému, bylo v budově univerzity vyčleněno jedno patro. Jedná se o prostředí, které kombinuje různorodé stavební prvky, jako například dveře z masivu, různé dělicí příčky, cihlové nosné zdi, skleněné zatahovací dveře, toalety a třeba také výtahovou šachtu. Pestré je také samotné vybavení prostor v podobě laboratorních a výukových předmětů. Mým úkolem je nyní toto prostředí analyzovat z pohledu lokalizačních systémů. Základní rozvržení přístupových bodů na patře vychází z fáze plánování od mého kolegy. Schéma rozložení přístupových bodů jsem převzal a vytvořil simulaci v programu *Ekahau Site Survey* (viz *Obr. 4.1*).



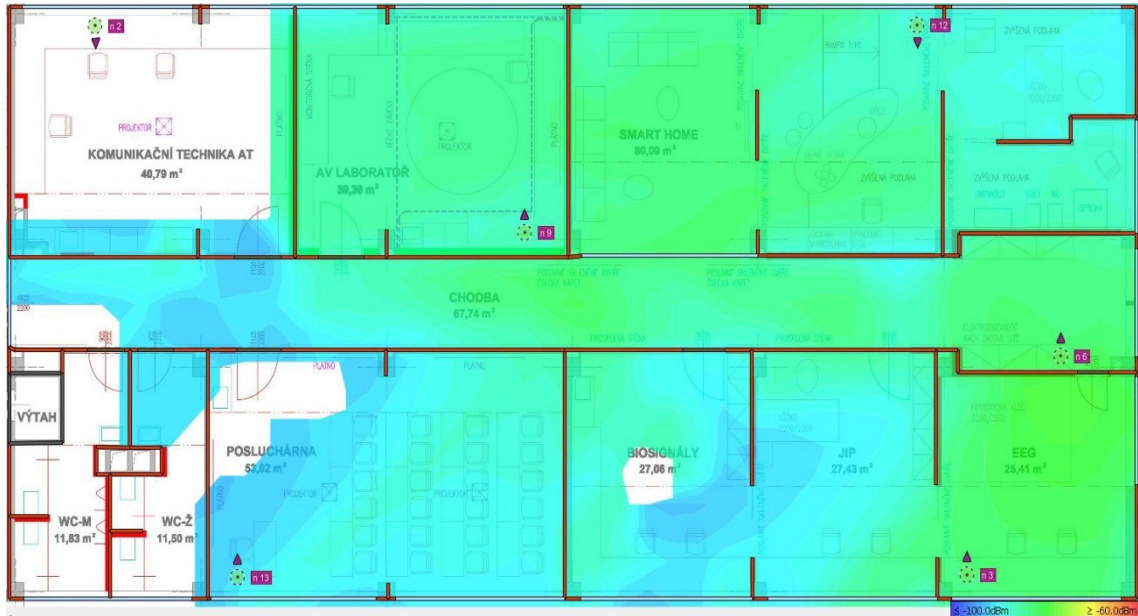
*Obr. 4.1: Simulace rozložení sil signálů*

Z obrázku tedy můžeme vyčíst, že pro pokrytí bylo využito šest přístupových bodů, kde každý vysílá na jiném kanále. Konkrétně to jsou kanály 2, 3, 6, 9, 12 a 13, které vysílají na standardu IEEE 802.11n. Simulace na obrázku výše konkrétně zobrazuje rozvržení síly signálu od jednotlivých přístupových bodů, kde je stupnice nastavena v rozmezí od modré do červené barvy, resp. od  $-80$  dBm do  $-20$  dBm. V tomto ohledu je tedy nejhorší pokrytí na toaletách a ve výtahu, což je zobrazeno modrou barvou. V místě nejsytější modré je výtahová šachta, kde však požadavek na lokalizaci není nutný. Už z důvodu takového, že dostatečné pokrytí signálem ve výtahu není možné, protože abychom zajistili v tomto místě signál z minimálně tří vysílačů, bylo by nutné jeho jádro obklopit vysílači v co neblížší vzdálenosti. Nejbližší možná místa, pro umístění přístupových bodů, jsou už však obsazena aktuálními pozicemi vysílačů, protože na toaletách či chodbě není dostupná žádná síťová a ethernetová zásuvka. Pro úplnost, zelené oblasti dosahují hodnot kolem  $-56$  dBm.

Co se týče koexistence vysílaných kanálů a jejich vzájemné ovlivňování. Na *Obr. 4.2* je znázorněna simulace vzájemných interferencí mezi kanály. Barevná stupnice je zde stejná jako výše, avšak čím sytější modrá (do  $-100$  dBm), tím přívětivější

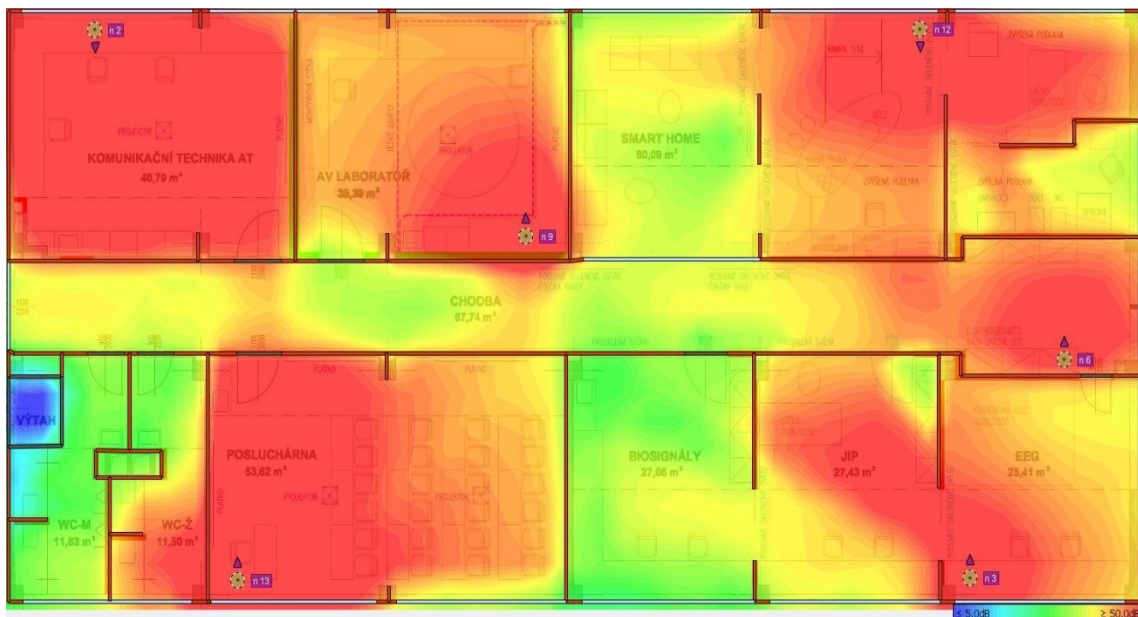


hodnota. Zaměříme se spíše na oblasti zelené (okolo  $-75$  dBm). Zde mezi sebou kolidují kanály 2 a 3, 3 a 6, 6 a 9, 9 a 12, 12 a 13. Jejich vzájemné překrytí se pohybuje kolem 5 MHz (viz *Obr. 2.1*) z celé šířky kanálu, jež činí 20 MHz pro standard IEEE 802.11n. Tato situace samozřejmě není ideální, musíme však vzít v potaz, že okolo daného prostoru existují i další WLAN sítě, například univerzitní. Z tohoto hlediska je i přizpůsoben výběr a umístění kanálů, aby nedocházelo k interferencím právě s těmito sítěmi. Důležité totiž je, aby laboratorní síť zásadním způsobem neovlivňovala okolí.



*Obr. 4.2: Simulace interference kanálů*

Dalším parametrem WLAN sítě je tzv. SNR (*Signal to Noise Ratio*), tedy odstup signálu od šumu. Ten je definovaný jako podíl hladiny signálu a šumu. Čím je tedy odstup mezi hladinami větší, tím dochází k menší chybovosti při přenosu dat. Na *Obr. 4.3* je vyobrazena simulace parametru SNR. Oblasti s červeným zvýrazněním dosahují



*Obr. 4.3: Simulace parametru SNR*

odstupu 50 dB a výše. Problematická oblast toalet dosahuje hodnoty kolem 25 dB, což můžeme pokládat za přijatelnou hodnotu.

Co však do jisté míry není tolik závislé na okolních sítích, je pokrytí každého bodu prostoru minimálně třemi přístupovými body z dané sítě (*Obr. 4.4*). Dle předešlých rozborů se dá předvídat, že špatná situace bude opět v oblasti toalet. Simulace zde místy zobrazuje dokonce jedno dostupné AP. V lepším případě dvě. Pro účely lokalizace to je však nepříjemné. Na tento problém se tedy musím zaměřit ve fázi analýzy spektrálním analyzátozem. Až zde bude jasno, jestli bude nutné a možné zasáhnout do stávající infrastruktury. Každopádně ostatní místa jsou pokryta minimálně třemi a více vysílači. Nevhodnocené (bílé) okraje mapy jsou způsobené simulačním programem, avšak dá se předpokládat, že tato místa požadavky splňují.



*Obr. 4.4: Simulace počtu dosažitelných vysílačů na různých místech*

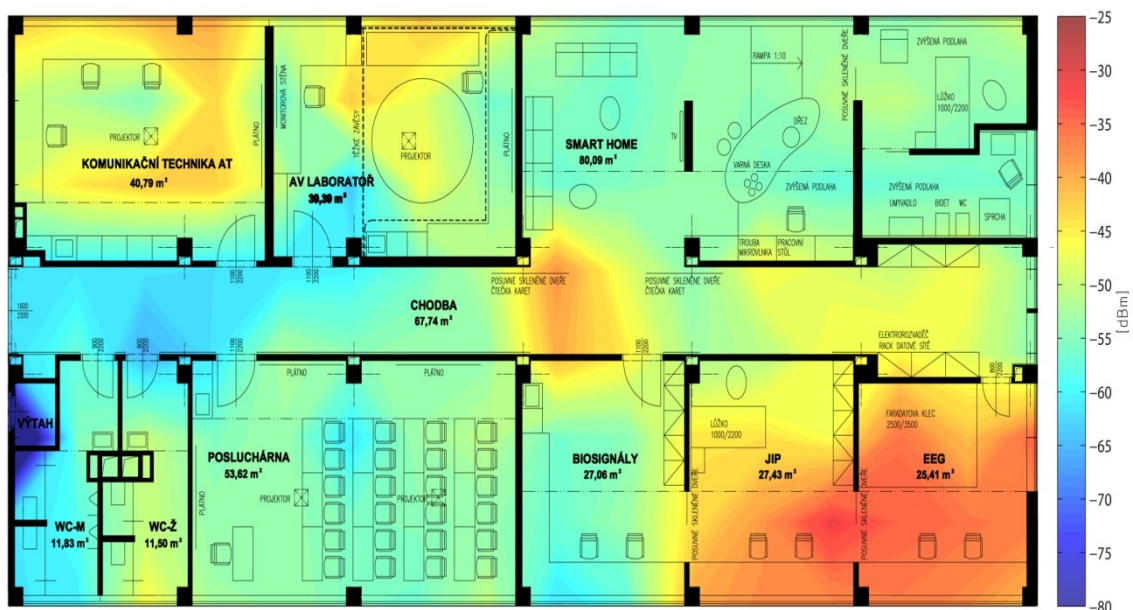
## 4.2 Analýza pokrytí s využitím měřících přístrojů

Pro účely následujících dvou metod měření pokrytí jsem si daný prostor popsal pomocí maticové mřížky. Jednotlivé rozestupy bodů v obou osách jsou dva metry. Ve výsledku tím vznikne 128 bodů, jejichž výsledná vizualizace je zpracována programem Matlab.

### 4.2.1 Analýza spektrálním analyzátozem

Pro měření jsem použil spektrální analyzátor Spectran HF - 6060 a jeho doplňkový software MSC, jež umožňuje konfiguraci analyzátoru a vizualizaci hodnot. Měření probíhalo s využitím všesměrové antény. V předdefinovaném nastavení pro měření celého pásma, tedy 2,4 - 2,5 GHz, naměřené hodnoty, jednotlivých úrovní signálu pro konkrétní kanály, dosahovaly neobvyklých výsledků. Konkrétně se jednalo o veliký interval kolísání signálu, a to i v případě, kdy byl přístroj v klidu a hned vedle vysílače. Po konzultaci problému jsem zjistil, že kolísání signálu svědčí o správném chodu vysílače, neboť přístupové body mohou disponovat regulací vyzářeného výkonu nebo regulací šířek pásem přenosových kanálů. Proto i při měření pouze šířky jednoho celého kanálu (viz *Obr. 2.1*) hodnota signálu nadále dosahovala velkých fluktuací. Z tohoto pohledu bylo nutné upravit výchozí nastavení přístroje. Konkrétně se jednalo

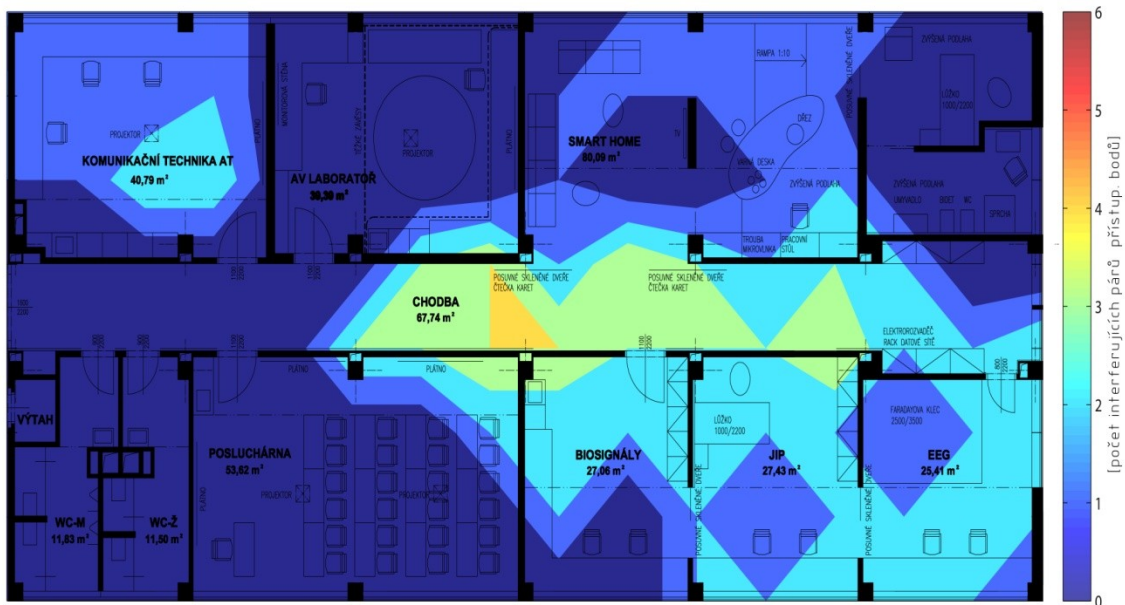
o vytvoření měřicích profilů, které vždy definují měřené pásmo v konkrétním kanálu (1, 2, 6, 9, 12 a 13) na 4 MHz, jehož střední frekvence je vždy střední frekvence kanálu. Dále bylo nutné upravit dobu měření této šířky pásma a to konkrétně na tři vteřiny. Mohlo být využito i pět vteřin, avšak rozdíl naměřených hodnot nebyl tak markantní, že by tím bylo znehodnoceno měření. Poslední důležitou položkou bylo nastavení filtru pásma na 100 kHz. Pro každý bod byly naměřeny čtyři hodnoty, které se posléze průměrovaly, resp. po jejich převodu na miliwatty. Kompletní měření jednoho bodu tedy zabralo pět minut. Nemohl jsem však měřit jeden bod pro všech šest současně zapnutých přístupových bodů, protože v případě, kdy se například měřil bod u AP s kanálem 3, nebylo možné měřit sílu signálu kanálu 2, jehož vysílač byl na druhé straně podlaží. Je to z důvodu jejich částečně překrývaných pásem, takže při měření kanálu 2 u vysílače s kanálem 3, naměříme hodnotu signálu vyšší, která však patří kanálu 3. Je důležité také dodat, že celý prostor je ovlivněn okolními sítěmi, hlavně těmi univerzitními (eduroam, FEL). Ty kompletně pokrývají celé pásmo, kde největší koncentrace vysílačů je na kanálu 1, poté 12 a 13. V měřeném prostoru je tedy možno objevit několik desítek vysílačů, když tedy započítáme i veškeré okolní sítě. Proto jsou výsledky tohoto měření z části ovlivněné právě existencí okolních sítí, kde však nebylo možné tento faktor nijak ovlivnit, protože vysílače v průběhu dne mohou měnit své kanály. Z tohoto pohledu je tedy vliv nekonstantní.



Obr. 4.5: Naměřené hodnoty sil signálů

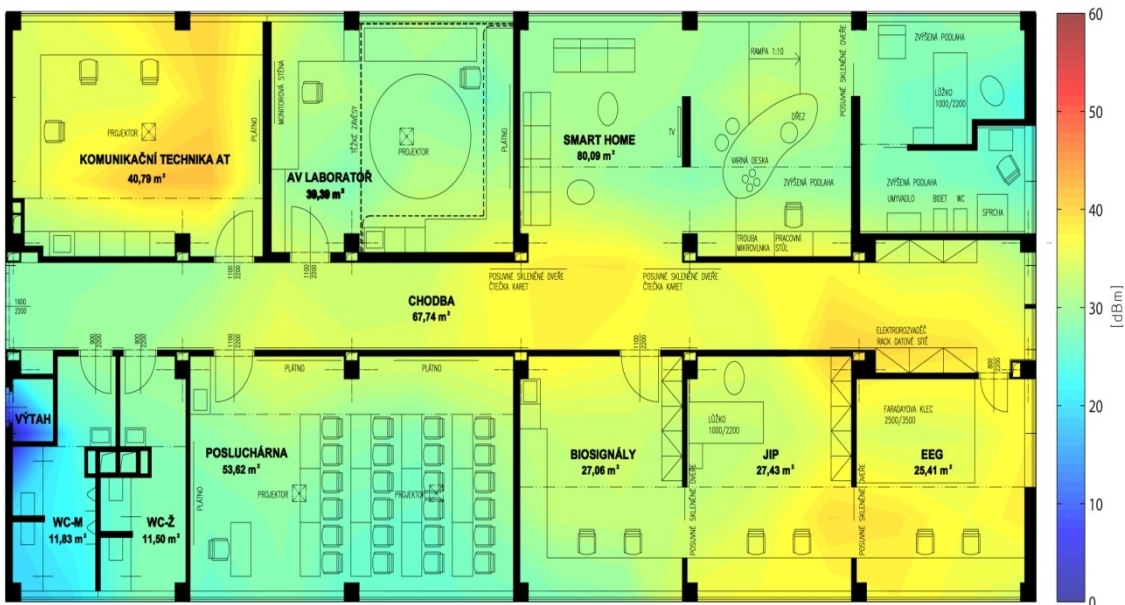
Obr. 4.5 znázorňuje rozložení síly signálu. Každý bod mřížky je zde reprezentován vždy jako maximální hodnota ze souboru šesti hodnot, kde každé AP se podílí jednou hodnotou pro dané místo. Vizualizace tedy ověřila skutečnost, že je pokrytí dostatečné. Tmavé místo v prostoru výtahu jsem neměřil. Místa, kde vznikají největší interference, z pohledu vzájemného ovlivňování překrývajících se kanálů, jsou vyobrazena na Obr. 4.6. Zde čísla na stupnici označují počet dvojic kanálů, které v daném místě navzájem interferují. Jednotlivé dvojice kanálů jsou: 2 a 3, 3 a 6, 6 a 9, 9 a 12, 12 a 13. V určitých motivech rozložení kopíruje simulaci z Obr. 4.2. Největší interference tedy vznikají na chodbě, což je logické, protože svými proporcemi připomíná vlnovod, do kterého signál prostoupí a poté se v něm signál dále šíří.





Obr. 4.6: Počet dvojic vysílačů, jejichž kanály na daném místě interferují

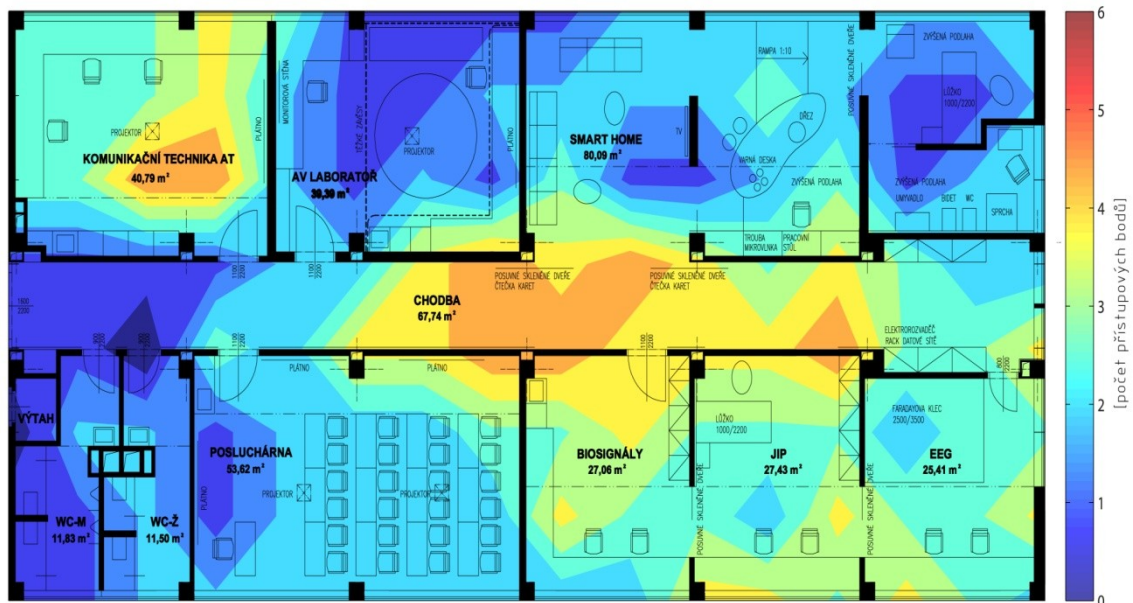
Následná vizualizace (Obr. 4.7) znázorňuje rozložení parametru SNR, který popisuje odstup hladiny signálu od hladiny šumu na pozadí, tedy jakým způsobem je ovlivněna chybovost přenosu. V logaritmickém měřítku je SNR rozdíl signálu a šumu, kde jsem



Obr. 4.7: Rozložení hodnoty užitečného signálu k šumu

referenční hladinu šumu zvolil  $-100$  dBm. Vycházím z úvahy, že je signál nedosažitelný při  $-100$  dBm, což je také hodnota spektrálního analyzátoru v případě, že nic nedetekuje. Z tohoto pohledu, tedy při hladině signálu  $-100$  dBm a hladině šumu  $-100$  dBm, je výsledek  $SNR = 0$ . Vizualizace nebere v úvahu interference mezi kanály, proto je v místech vyšší interference také vyšší SNR. V opačném případě by bylo nutné použít parametr SIR (*Signal-to-Interference Radio*) nebo SINR (*Signal-to-Interference-plus-Noise Ratio*), jež berou v úvahu i interference. Nejdůležitější pro lokalizační systémy je však počet dostupných vysílačů pro každý bod prostoru. Tento problém je

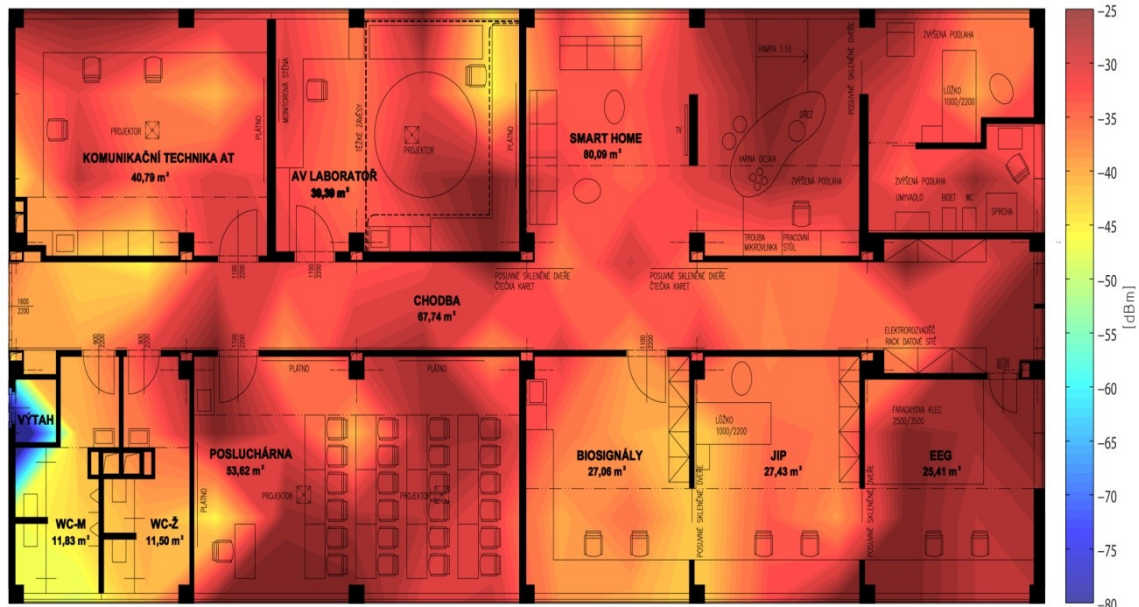
zobrazen na *Obr. 4.8*, který bere v úvahu body, kde signál od konkrétního vysílače dosahuje hodnoty nejméně  $-65$  dBm. Výsledkem tedy je, že většina míst podmínku minimálně třech dosažitelných vysílačů nespĺňuje, samozřejmě to je pro hodnoty síly signálu větší jak  $-65$  dBm. Tento problém však není tak jednoznačný, protože v měření následující metodou, je tento výsledek zcela odlišný.



*Obr. 4.8: Počet dostupných vysílačů pro dané místo*

#### 4.2.2 Analýza externím Wi-Fi adaptérem

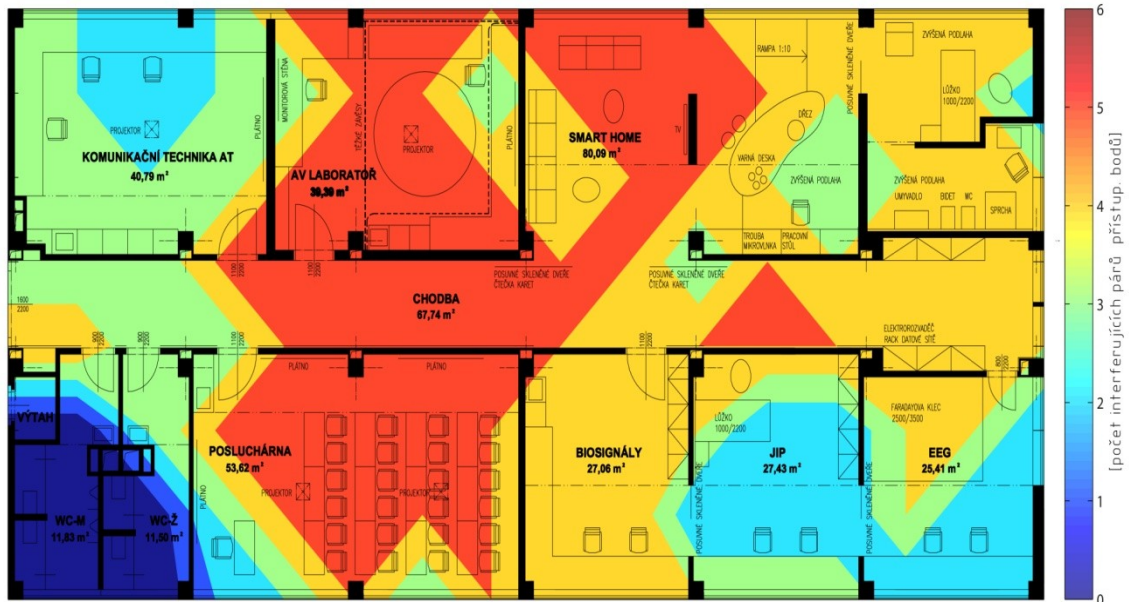
Druhá analýza byla provedena USB Wi-Fi adaptérem Realtek RTL8191S, který byl připojen k notebooku, a který byl umístěn nad horní hranou víka notebooku, aby byly



*Obr. 4.9: Naměřené hodnoty sil signálů USB adaptérem*

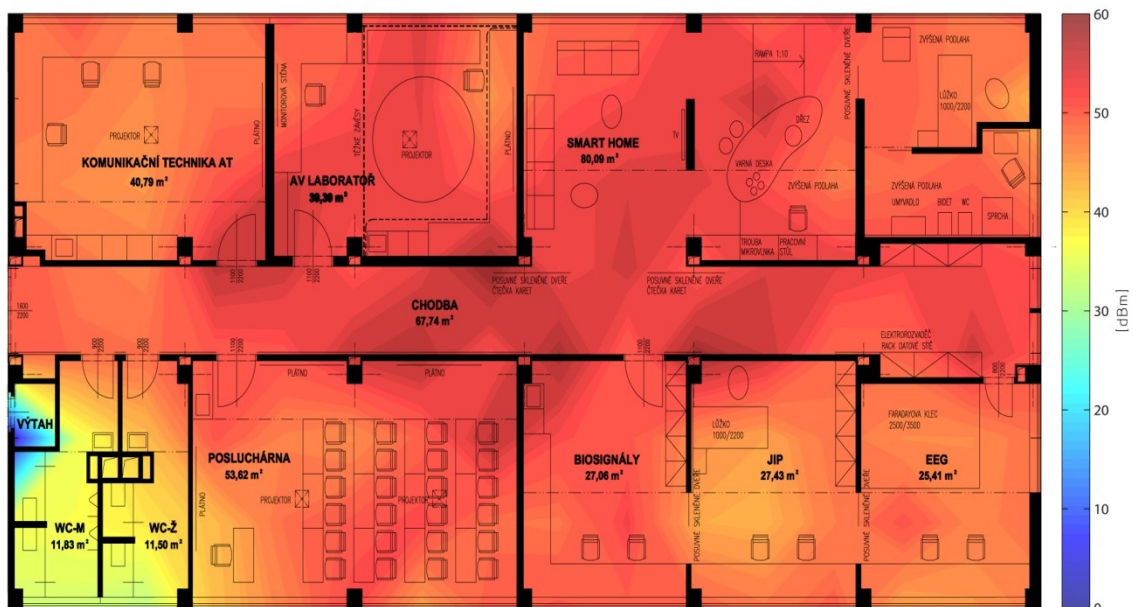


zčásti omezeny interference způsobené samotným víkem a lidským tělem. Pro měření jsem použil program inSSIDer 3. Tuto metodu jsem zvolil pro to, abych upřesnil výsledky z měření spektrálním analyzátozem. Konkrétně kvůli chybě metody měření, protože spektrální analyzátoz měří jakoukoli hladinu v měřeném pásmu. Oproti tomu počítačový program konkretizuje sílu signálu pro daný kanál naší síť. Vedlejším důvodem bylo porovnání těchto dvou metod. Z jednotlivých vizualizací je patrné, že hodnota úrovně síly signálu naměřená analyzátozem je o několik decibelů nižší než

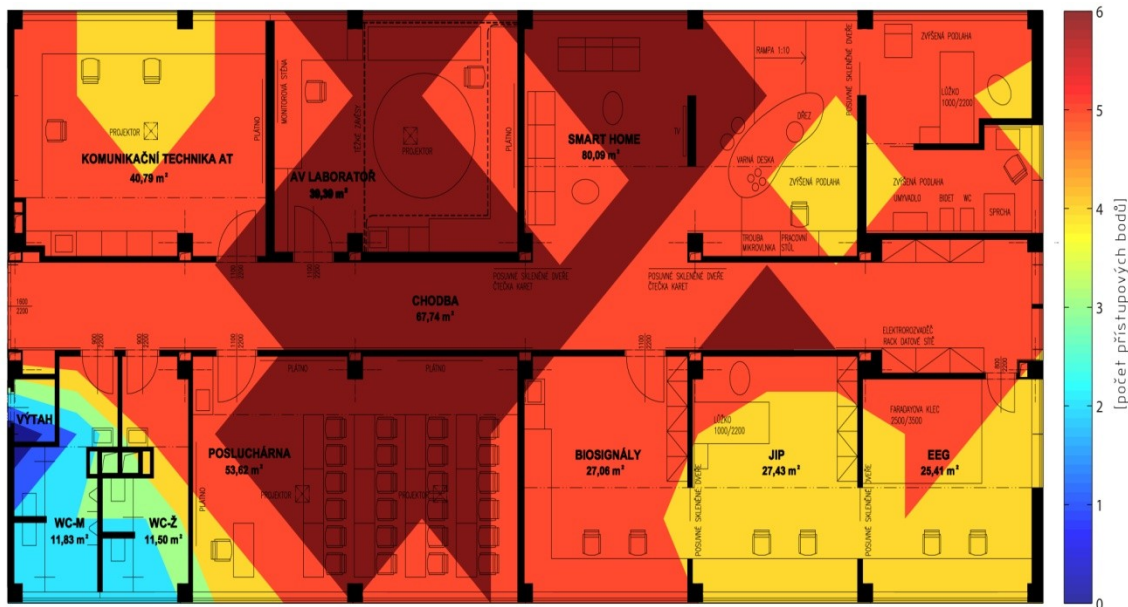


Obr. 4.10: Počet dvojic vysílačů, jejichž kanály na daném místě interferují

hodnota naměřená notebookem (Obr. 4.5 a 4.9). Tento rozdíl je způsobem metodikou měření analyzátozem. Při něm jsem nastavoval průměrování naměřených hodnot signálu z celé šířky měřeného pásma, tedy 4 MHz. Když jsem měřil pouze maximální hodnotu, která byla po celou dobu skenování měřeného pásma dosažena, tak občas docházelo k neodpovídajícím výsledkům, pravděpodobně v důsledku zarušeného prostoru.



Obr. 4.11: Rozložení hodnoty užitečného signálu k šumu



Obr. 4.12: Počet dostupných vysílačů pro dané místo

Na vizualizacích interference a parametru SNR (Obr. 4.10, resp. Obr. 4.11) můžeme opět pozorovat, jednotlivá rozložení hodnot a jejich podobnost s metodou měření v minulé podkapitole. Chtěl bych se však zaměřit na Obr. 4.12, který dosahuje zcela odlišných výsledků v porovnání s předchozí metodou. Mezní hodnota signálu je zde také brána jako  $-65$  dBm. Zde je, i díky metodice měření, splněna podmínka minimálně tří dostupných vysílačů téměř po celém prostoru pro signál vyšší jak  $-65$  dBm.

Co se tedy týče závěru z těchto dvou analýz. Pro účely funkce lokalizačního systému je toto prostředí náročné na jeho správnou funkci, zejména díky interferencím z okolních sítí. Prostor toalet sice pokrývají minimálně tři vysílače, avšak s omezenou silou signálu. Díky těmto faktorům může být lokalizace zejména na toaletách a chodbě výrazně ovlivněna, samozřejmě záleží na využití technologii. Z pohledu lokalizačního systému, který bude v této práci později navrhován, bude asi největší problém s interferencemi. Základem systému bude totiž práce s hodnotami síly signálu. Podrobnější analýzy pokrytí od jednotlivých vysílačů obsahuje Příloha A a B.

### 4.3 Implementace Ekahau systému

Prvním krokem, implementace Ekahau RTLS, bylo vytvoření tzv. *Site survey* (proměření místa), který je označován jako trénovací či offline fáze. Poté bylo nutné na jednom notebooku nakonfigurovat lokalizační stroj a dohledový software. Co se týče přesnosti lokalizace, tak ta se odvíjí od kvality nasbíraných dat v offline fázi.

Vlastní proces lokalizace probíhal za pomoci programového klienta v notebooku. Kvůli časovému omezení a zaměření se spíše na správnou funkci našeho lokalizačního systému, nebyly využity identifikátory zmiňované v kapitole 3.5. Přesnost lokalizace se pohybovala v rámci jednoho až třech metrů, kde bylo samozřejmě nutné setrvání na daném místě, aby se výchylka signálu ustálila a bylo dosaženo lepších výsledků.

## 5. Mobilní aplikace

### 5.1 Návrh lokalizačního systému

V kapitole o lokalizačních systémech jsem provedl rozbor toho, jaké jsou možnosti pro to, aby byl lokalizační systém vůbec schopen lokalizovat. To znamená principy využití stávající síťové infrastruktury pro získávání dat a jejich zpracování. V kapitole o analýze pokrytí bezdrátové sítě jsem naopak pochopil, že pro správnou funkci takového systému, je pro něj nutné vytvořit správné podmínky, což bylo ověřeno i na systému Ekahau RTLS. V této kapitole, mobilní aplikace, bych rád využil dosavadní poznatky o problematice lokalizačních systémů nasazovaných v prostorách budov k tomu, abych na toto téma navrhl lokalizační systém a vytvořil mobilní aplikaci pro správce systému. Mobilní aplikace bude jednou ze součástí systému, nebude tedy primárně určena jako samostatná jednotka pro lokalizaci, i když k tomu bude mít implementované všechny potřebné náležitosti.

Abych se ale vrátil k návrhu lokalizačního systému. Jeho hlavním účelem bude lokalizace osob v prostorách budovy. To znamená, že by se jeho přesnost měla pohybovat minimálně na principu lokalizace na úrovni pokoje, kde bude samozřejmě snaha o to, dostat se na lokalizaci precizní, jejíž odchylka pro systémy na základě Wi-Fi technologie se pohybuje okolo jednoho až pěti metrů. Za předpokladu, že parametry bezdrátového pokrytí jsou už v daném prostoru optimalizovány pro nasazení lokalizačního systému, tak se nyní musím zaměřit na požadavky, které jsou kladeny na lokalizaci v budově. Systém by měl tedy splňovat následující vlastnosti:

- a) implementace systému by měla minimálně ovlivnit stávající síť, samozřejmě je nutnou podmínkou, aby každé místo prostoru bylo pokryto minimálně třemi přístupovými body, což je společné pro většinu lokalizačních systémů
- b) pro implementaci systému, by se nemusely dokupovat speciální identifikátory, v podobě těch, které jsou využívány systémem Ekahau (viz Kap. 3), ale mohlo by se využít mobilních zařízení, samozřejmě záleží na účelu lokalizace
- c) používání systému by mělo být jednoduché
- d) lokalizace by měla dosahovat co nejpřesnějších výsledků
- e) je požadováno rychlé vyhodnocení pozice
- f) systém by měl využívat operační systémy, které jsou aktuálně nejvíce rozšířené
- g) přesnost systému by měla být co nejméně ovlivnitelná faktory prostředí

Dále by měl lokalizační systém obsahovat hlavní tři komponenty, které umožňují získat dat, jejich zpracování a následnou vizualizaci. Tedy tyto:

- h) identifikátor
- i) lokalizační stroj
- j) dohledový software

Nejprve je tedy nutno zvolit platformu, kterou bude lokalizační systém podporovat. Nabízí se Windows, Mac OS X, Linux, z těch mobilních Windows Phone, iOS a Android OS. Aby se nemusely pořizovat speciální hardwarové identifikátory a jelikož navrhovaný systém nebude sloužit pro speciální účely (nemocnice apod.), je možno



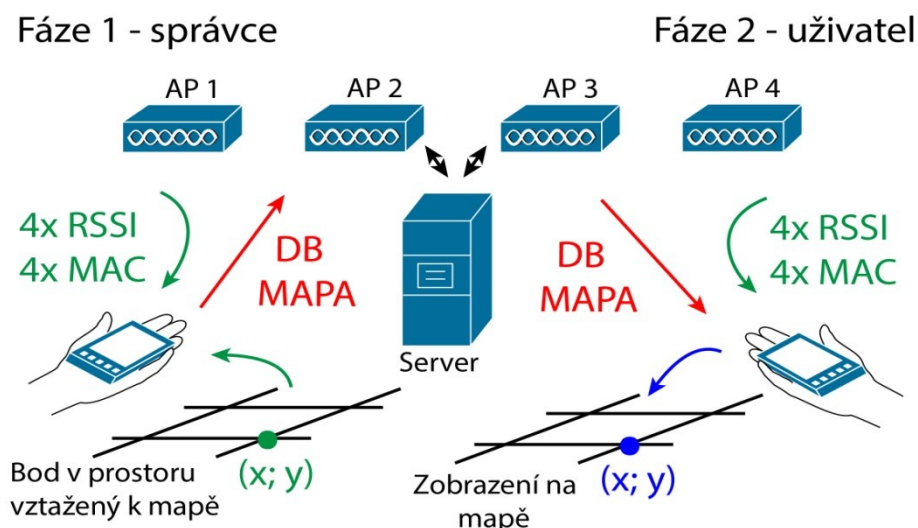
využít mobilních zařízení v podobě chytrých telefonů či tabletů. Rozšířenost těchto zařízení stoupá, takže využití notebooků můžu odsunout do pozadí, protože dnes už přeci jen téměř každý vlastní chytrý telefon, jehož váha, kompaktnost, delší výdrž na baterie a neustálá přítomnost jsou důležité faktory. Z toho důvodu tedy dojde k nasazení přenosného zařízení uživatele, které bude vybaveno lokalizačním klientem a jako celek bude plnit funkci identifikátoru. Abych se tedy vrátil k výběru platformy, aktuálně nejrozšířenějším systémem pro tato zařízení je Android OS. Proto, na rozdíl od konkurence, může nabídnout širší spektrum potenciálních zájemců. Nyní je nutné stanovit základní strukturu celého lokalizačního systému. Jako identifikátor jsem zvolil přenosné zařízení na platformě Android. Co se týče dohledového programu, tak ten by mohl být buď využíván na stolních počítačích, notebookích, ale jako perspektivnější se mi jeví využít opět přenosné zařízení.

Zde ale musím ujasnit jednu věc. Existují dva různé pohledy na lokalizaci. Jeden pohled je takový, že osoba vlastní například chytrý telefon, na něm si spustí lokalizační aplikaci, kde je zobrazena mapa prostoru a na vyžádání zobrazí polohu uživateli. V této situaci je identifikátor chytrý telefon, lokalizační stroj je buď na vzdáleném serveru, nebo přímo také v telefonu, a dohledový software je vlastně také součástí aplikace v telefonu, protože vizualizace výsledku probíhá právě v ní. Druhý pohled je takový, že osoba si do telefonu nainstaluje pouze lokalizačního klienta, který po spuštění běží na pozadí a zasílá data do lokalizačního stroje, který je také buď na vzdáleném serveru, nebo přímo v telefonu, ale dohledová aplikace je fyzicky přístupná na jiném přenosném zařízení. Mým cílem je tedy dosáhnout varianty první, tedy „chci vidět svoji polohu na mapě“. Aby tedy systém lokalizoval, pokud možno za co nejkratší dobu, rozhodl jsem se využít toho, že lokalizační stroj bude součástí aplikace. Nyní ještě zbývá poslední důležitá věc, a to, jakou techniku a algoritmus pro tento systém využít.

Lokalizační metodu, kterou bych rád implementoval do návrhu lokalizačního systému, jsem jednou zmínil u RSSI, techniky pro získávání dat. Jedná se o *Localization fingerprinting* [16], doslovně přeloženo jako lokalizace otisků prstů. Otisky však budou zcela jiného původu. Základem metody jsou dvě fáze, první je tzv. *offline*, někdy nazývaná též jako *trénovací*, druhá je tzv. *online*, což nazveme *vlastní lokalizace*. Základem je samozřejmě mít pečlivě nastavené pokrytí prostoru Wi-Fi signálem (opět, aby minimálně tři vysílače pokrývaly každé konkrétní místo), kde chceme provádět lokalizaci. Poté se provede trénovací fáze, ta spočívá v tom, že si daný prostor rozdělíme na souřadnicovou síť. V jednotlivých průnicích osy  $x$  a osy  $y$  nyní provedeme měření sil signálů od všech dostupných vysílačů. Počet RSSI hodnot je však lepší mít pro každý bod stejný a nejlépe, aby se počet hodnot pohyboval mezi čtyřmi až pěti. Málo hodnot způsobuje nepřesné stanovení pozice, naopak více hodnot než pět, lokalizaci už nezlepší, naopak v některých případech může i zhoršit. Tímto způsobem se tedy projde celý prostor a vznikne databáze, kde jeden záznam obsahuje pozice  $x$  a  $y$ , kde měření hodnot probíhalo. Dále obsahuje čtyři hodnoty RSSI hodnot, avšak těch čtyřech nejvyšších, které byly v daném bodě dostupné. Ty se do databáze ukládají seřazené sestupně. Dále můžeme získávat například i MAC adresy, které mohou být využity pro zpřesnění lokalizace, a které se musí do databáze ukládat v pořadí, v jakém se ukládají RSSI hodnoty k nim vztažené. Po vytvoření databáze hodnot se může spustit vlastní lokalizace. Ta spočívá v tom, že na žádost osoby, která klikne na tlačítko *Lokalizovat*, dojde ke skenování dostupných vysílačů a vybrání čtyř největších hodnot RSSI, tímto způsobem vznikne vektor  $R$ . Ten je poté algoritmem nejbližšího souseda (podkap. 3.2), porovnán postupně se všemi vektory  $P$  z databáze, která obsahuje záznamy z trénovací fáze.

Tímto způsobem dojde k nalezení bodu z databáze, který nejlépe odpovídá aktuální pozici. Jelikož může nastat situace, kdy dostaneme několik bodů, jejichž vektory hodnot jsou nejpodobnější, a to i v případě, že body mohou být různě v prostoru, je tomu nutné aspoň částečně předejít. Proto jsem tento algoritmus pro navrhovaný lokalizační systém doplnil o filtr MAC adres. Princip algoritmu je tedy následující. Nejprve se vypočítá odlišnost (vzdálenost) vektoru aktuálně naměřeného a vektoru z databáze. Pokud mají oba vektory stejný počet MAC adres, tak se danému výsledku přiřadí priorita. Ta se odvíjí podle počtu, tedy čím více, tím větší priorita (max. 4). Tím bude výsledek obsahovat přesnější vektor, protože má více hodnot. Tímto způsobem se projde celá databáze a zapamatují se hodnoty  $x, y$  pro čtyři vektory nejméně odlišné, avšak s největší prioritou. Dodatečným filtrem se tedy omezí případné nedokonalosti lokalizace na minimum. I když tedy budou na opačných koncích daného prostoru existovat například dvě místa se stejnými hodnotami RSSI, pomocí MAC adres se dojde k závěru, že právě jedno místo a jeho nejbližší AP (největší RSSI hodnota) má odlišnou fyzickou adresu.

Abych tedy shrnul dosavadní úvahy, které se týkají základních funkcí aplikace. Mobilní aplikace, pracující na platformě Android, bude plnit funkci identifikátoru, dohledového programu a lokalizačního stroje (databáze může i nemusí být součástí aplikace). Tuto aplikaci bych zdefinoval jako aplikaci *uživatelskou*. Je to z toho důvodu, že se ještě musí nějakým způsobem vytvořit databáze dat, která bude vytvářena při fázi trénování. Aby se zbytečným způsobem situace nekomplikovala, tak je možné využít stávající uživatelské aplikace, která bude doplněna o funkci vytváření databáze. Tuto nově vzniklou aplikaci bych zdefinoval jako aplikaci *správce* (jež je cílem práce). Ve výsledku by tedy měly vzniknout dvě aplikace, pro uživatele a správce. Je totiž zbytečné a celkem i nežádoucí, aby normální uživatel, který přijde do lokalizovaných prostor, měl funkci pro tvorbu databáze. Jak z hlediska bezpečnostního, aby například nedošlo k tomu, že by přepisoval dosavadní databázi, tak i z hlediska pohodlí, aby naopak nemusel celou databázi hodnot nejdříve uživatel vytvořit a až poté by ji mohl používat.



Obr. 5.1: Princip metody fingerprinting

Z tohoto pohledu by v první fázi (trénovací) například správce dané sítě prošel prostor, který bude sloužit pro lokalizaci, a vytvořil pomocí aplikace správce databázi hodnot

pro následnou lokalizaci. Tato databáze by se poté umístila na sdílené místo, odkud by se mohla distribuovat do uživatelských aplikací, například na firemní server. Normální uživatel by si tedy jen stáhl aplikaci pro něj určenou. Poté by se automaticky stáhla aktuální verze databáze (popřípadě mapa prostoru), a bezprostředně by mohl využívat lokalizaci ve svém chytrém telefonu či tabletu v podobě vizualizace výsledku na mapě. Stáhnutím databáze do uživatelského telefonu by se také zajistila patřičná rychlost lokalizace, protože síťová infrastruktura by byla využita jen na úrovni přístupových bodů. Výsledné schéma systému popisuje *Obr. 5.1*.

## 5.2 Platforma Android

Pro vývoj mobilní aplikace jsem se rozhodl využít platformu Android. K tomuto rozhodnutí mě vedlo hned několik faktorů. Z části to vyplývá z návrhu lokalizačního systému. Hlavním pozitivem je jeho velká popularita, která pramení z několika míst. Jedná se totiž o systém, který je z velké části otevřený, tzn., že vývojáři či výrobci mohou volně přistupovat k jeho zdrojovým kódům. Některé jeho technologie však podléhají licenci, protože jsou patentované. To nic nemění na tom, že si ho mohou výrobci optimalizovat dle svých potřeb a jeho využití není tolik ekonomicky náročné, jako u jeho konkurence. I díky tomu se stal nejrozšířenějším systémem nasazovaný v přenosných zařízeních (chytré telefony, tablety). S tím však souvisí i jeho negativum, při vývoji aplikace musíme počítat s tím, že může být spuštěna právě na různých typech přenosných zařízení. Ta disponují různorodými hardwarovými konfiguracemi, jak z pohledu procesoru, paměti, tak i rozlišením samotného displeje. Z pohledu softwaru je také nutné zajistit podporu různých verzí Android platformy (viz *Tab. 5.1*), jak už z pohledu kompilace pro cílovou verzi, tak už samotným využitím podporovaných programových prostředků.

*Tab. 5.1: Verze systému Android [17, upraveno]*

Verze	Označení	API
2.2	Froyo	8
2.3.3 - 2.3.7	Gingerbread	10
3.2	Honeycomb	13
4.0.3 - 4.0.4	Ice Cream Sandwich	15
4.1.x	Jelly Bean	16
4.2.x		17
4.3		18
4.4	KitKat	19

Co mě také však přimělo, využít platformu Android, je pohled čistě programátorský. Základem je totiž programovací jazyk Java, se kterým jsem se už v průběhu studia seznámil, a proto stačilo připomenout pár základů. Abych se ale vrátil zpět k samotnému systému, ten je rozdělen na pět základních vrstev. Základem je linuxové jádro, které má na starosti hardwarové ovladače či správu napájení. Nad ním jsou knihovny pro práci například se SQL databázemi, grafikou apod. Další vrstvou je virtuální stroj Dalvik, jež je optimalizovaná náhrada na přenosná zařízení za JVM (*Java Virtual Machine*). Čtvrtou vrstvou je aplikační rámec, který obsahuje manažery pro zobrazení, zdroje, připojení, telefonování, pro práci s aktivitami apod. Jako poslední a nejzajímavější je aplikační vrstva, jež obsluhuje chod samotných aplikací.

Ve zkratce bych ještě uvedl základní princip aplikací Android. Hlavním stavebním prvkem jsou tzv. *aktivita*. Jsou to třídy odvozené od nadtržidy `Activity`. Můžeme si je představit jako aktuální obrazovku aplikace, která čeká na uživateli pokyny. Nepleťme si to však s grafickým rozložením tzv. *layoutů*, které obsahují prvky odvozené od třídy `View`. Těmito prvky mohou být tlačítka, zadávací pole, umístění obrázků, textová pole, ale také vlastnosti, jako jejich vzájemné umístění, vzhled, velikost apod. Ty jsou definovány v XML souborech a tvoří již zmíněný *layout*, který je tedy jakousi vizuální podobou aktivity. Základem každé aktivity je metoda `onCreate()`, která je volána při každém jejím spuštění. Zpravidla na jejím začátku se volá metoda `setContentView()`, jež načítá *layout*, který zobrazí definované rozložení prvků (viz *Obr. 5.2*). V příkladu jsem použil úryvek kódu z aplikace, kde dochází ke spuštění aktivity `Localization` a načtení rozložení definované v XML souboru `layout_localization`.

```
public class Localization extends Activity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.layout_localization);  
    }  
}
```

*Obr. 5.2: Aktivita Localization*

### 5.3 Použité vývojové nástroje

Pro samotný vývoj aplikací platformy Android se používá svazek nástrojů, tzv. ADT (*Android Developer Tools*) Bundle, který je volně ke stažení. Obsahuje dvě základní komponenty, a to Android SDK (*Software Development Kit*), jež obsahuje knihovny programového rozhraní a vývojové prostředí Eclipse IDE (*Integrated Development Environment*), které je oficiálně podporováno společností Google, pod kterou Android spadá. Součástí IDE je tedy implementovaný doplněk ADT, který do prostředí integruje vývojové nástroje platformy Android. Mezi tyto nástroje patří různé perspektivy. Například Java, sloužící pro psaní vlastního kódu, DDMS, což slouží k optimalizaci, ladění a simulaci chování zařízení. Další využívanou perspektivou může být i debugger, pro analýzu chování kódu.

Mezi další nástroje ADT doplněk patří například editor *layoutů*. Ten slouží k vytváření grafického rozložení prvků aplikace. Pokud vývojář nedisponuje zařízením se systémem Android, což byl bohužel i můj případ, může využít emulátor zařízení. Podmínkou však je vytvoření konkrétního emulátoru, který odpovídá kýženému zařízení. Můžeme proto nastavovat různé velikosti obrazovky, hustoty obrazových bodů, hardwarovou konfiguraci, včetně SD karet, kamer, hardwarových tlačítek apod. Je také možno libovolně nastavovat podporovanou verzi Android systému a samozřejmě také typ zařízení, tedy chytrý telefon nebo tablet. Vesměs jsou tato zařízení už předdefinována tak, aby odpovídala skutečným vzorům. Takovýchto emulátorů můžeme mít definováno hned několik a funkcionalitu aplikace můžeme tedy vyzkoušet na hned několik odlišných typech zařízení. Počet paralelně spuštěných emulátorů je poté už jen omezen velikostí systémové paměti počítače.

## 5.4 Návrh aplikace pro správce

Základní komponentou lokalizačního systému je mobilní aplikace pro správce. Jejím hlavním účelem je tvorba databáze vzorků při fázi trénování, o které jsem se zmínil v předchozí podkapitole. Hlavním požadavkem na aplikaci je tedy práce s databází a mapou. Pro práci s databází jsem přemýšlel o využití MySQL databáze na vzdáleném serveru. Toto řešení by mělo své výhody, například databáze dat by se nemusela udržovat přímo v telefonu, ale byla by přístupná kdekoli z firemní sítě. Pokud by například došlo k její aktualizaci, změny by se ihned projevíly. Samozřejmě by její velikost mohla být také řádově větší, než u databáze přímo v telefonu. Na druhou stranu, dnešní kapacity úložných prostor přenosných zařízení, typu chytrý telefon a tablet, dosahují takových hodnot, že tento rozdíl je téměř smazán. Důvod, proč jsem ale od tohoto řešení nakonec ustoupil, byl fakt, že platforma Android obsahuje svoji modifikaci databáze, a to s názvem SQLite. Ta je přímo určená pro mobilní zařízení, čili práce s ní je do určité míry svižná, což je u lokalizace vcelku zásadní požadavek. Jelikož se bude aktuální databáze stahovat přímo do zařízení, tak o to vyšší bude rychlost výpočtu při vlastní lokalizaci.

Využití databáze bude tedy následující. Ve fázi trénování se budou data zapisovat do nově vzniklé databáze (souboru) na kartě zařízení. Po ukončení této fáze, se soubor s daty umístí na server, odkud bude k dispozici pro danou uživatelskou aplikaci, která si ho stáhne společně s mapou prostoru. Pokud tedy dojde k výpadku serveru, po stažení databáze, lokalizace bude fungovat i nadále. Samozřejmě menší nevýhodou je aktuálnost databáze. Jednak při fázi trénování se nemění přímo daný soubor, který si mezitím stahují uživatelé, takže nemají aktuální data. Navíc je nutné, aby uživatelé měli stále tu nejaktuálnější verzi databáze ve své aplikaci. Z toho důvodu je nutné po každém spuštění uživatelské aplikace vyvolávat dialog, který by měl na starosti stažení aktuální verze balíčku. Balíčkem je myšlen soubor obsahující mapu, soubor s databází a popřípadě další konfigurační soubory.

Dalším důležitým bodem aplikace správce je práce s mapou. Jelikož je nutné, aby byla naměřená data RSSI a MAC vztažena k danému místu v prostoru, je zapotřebí, aby se při každém odběru vzorků zaznamenaly konkrétní souřadnice nejen do mapy, ale i do databáze. V principu to tedy bude fungovat tím způsobem, že se načte mapa daného prostoru, pro který chceme vytvořit databázi. Mapa se do aplikace může vložit buď přímo do určené složky na kartě, anebo ji bude možno stáhnout ze serveru. Poté se na displeji zobrazí značkovač, v podobě nějakého křížku, který bude statický a bude umístěn přesně uprostřed displeje. Mapu však bude možno po displeji posouvat a také přibližovat či oddalovat. Tím dosáhneme toho, že místo, na kterém stojíme, nastavíme přesně pod daný křížek a po klepnutí na dané tlačítko, pro zadání nového bodu, se provede skenování dat z přístupových bodů, do mapy se zakreslí bod a do databáze se zapíše záznam se všemi hodnotami. Hodnotami jsou míněny dané souřadnice  $x$ ,  $y$  měřené polohy, čtyři nejvyšší hodnoty přijatého signálu a k nim čtyři hodnoty MAC adres. Z vytvořené databáze, mapy a konfiguračních souborů se tedy poté může vytvořit konečný balíček. To by byl tedy základní účel aplikace správce.

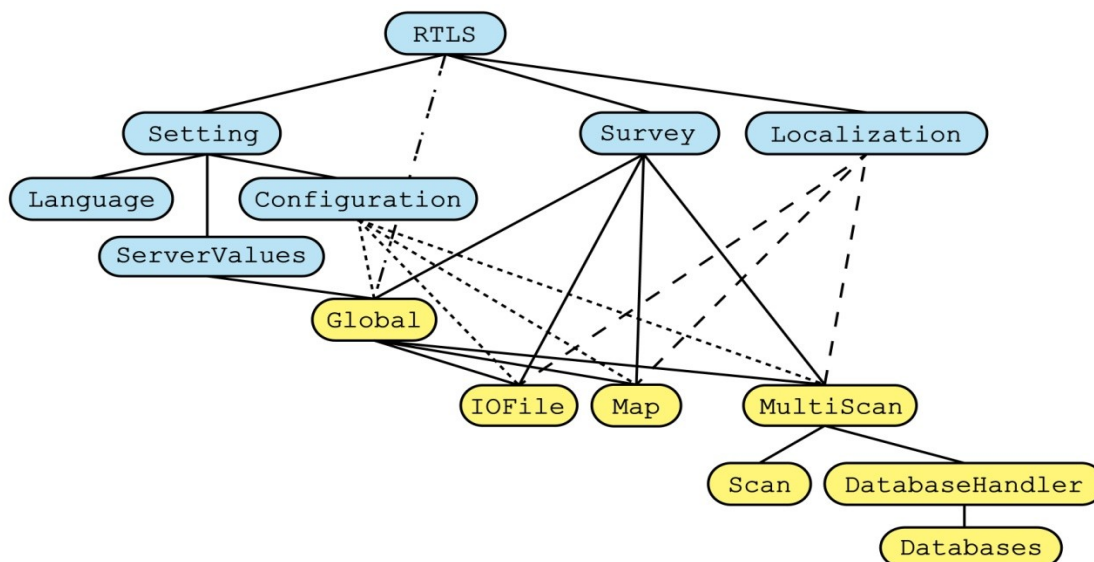
## 5.5 Zpracování návrhu aplikace

### 5.5.1 Programové třídy aplikace

V předchozí podkapitole jsem popsal základní pohled na funkčnost aplikace správce. Nyní však popíši její základní strukturu z pohledu využitých tříd, jež tvoří logiku aplikace. Je nutno podotknout, že zmíněné třídy jsou částí aplikace, druhá část tvoří komunikaci se serverem, které se věnuje kolega, a proto je zde nebudu popisovat. Třídy bych rozdělil na dva typy, a to třídy tvořící aktivity a třídy pomocné. Co je to aktivita jsem popisoval v podkapitole 5.2. První aktivita (třída), která je spuštěna po zapnutí aplikace, má název `RTLS.java`. Ta tvoří hlavní aplikační menu, které slouží jako spouštěč pro tzv. *podaktivitu*. Podaktivita je to samé jako aktivita, akorát je spuštěna z jiné aktivity. Ve výsledku to znamená, že spuštěná podaktivita běží na popředí hlavního uživatelského prostředí a předchozí aktivita je pozastavena. Abych se ale vrátil zpět. Z hlavního menu tedy můžeme spustit podaktivitu `Survey.java`. Ta je základním prvkem pro vytváření databáze hodnot a funkcí s tím spojených. V menu se skrývá pod označením *Proměření prostoru*. Další je podaktivita s názvem `Localization.java`. Ta plní stejnou funkci jako uživatelská aplikace, a to lokalizace na základě aktuální databáze. Tuto funkci jsem zde umístil z toho důvodu, aby správce, který vytvoří databázi hodnot, mohl ihned vyzkoušet její funkčnost. Nemusí tedy zapínat další aplikaci a tam zbytečně stahovat novou databázi a mapu. V menu je označena jako položka *Spustit lokalizaci*. Další položka v menu je *Nastavení*. Příslušná aktivita je `Settings.java`. V něm je možno, v některých případech nutno, nastavit základní parametry aplikace. Těmi může být konfigurace parametrů serveru, používaný jazyk aplikace, nastavení měřítka mapy apod. Konkrétně se jedná o aktivitu `Language.java`, jež nastavuje jazyk. V nastavení přístupná pod pojmem *Jazyk*. Poté o aktivitu `Configuration.java`, jejímž základním cílem je zadání měřítka mapy, což by mělo v budoucnosti sloužit pro účely navigace. V menu je označena jako *Konfigurace*. Poslední aktivitou, označenou tlačítkem *Nastavení sítě*, je `ServerValues.java`, kde uživatel nastaví parametry serveru. Pokud se vrátíme do hlavní nabídky aplikace, tak zde je ve vývoji ještě další položka, která bude mít funkci vzdáleného monitorování uživatelů, tedy jakýsi dohled.

To jsou tedy základní aktivity, se kterými se uživatel přímo seznámí. Co mu však zůstane skryto, jsou třídy pomocné, které už nejsou odvozené od nadtřídy `Activity`, a tak se nejedná o aktivity jako takové. Každopádně, první takovou je třída `Map.java`, která umožňuje práci s mapou, tedy posun, přiblížení, oddálení, ale také pomáhá s určením souřadnic na mapě. Druhou velice důležitou třídou je `Scan.java`. Ta obstarává skenování signálu v okolí zařízení. Díky ní se získávají data jako RSSI a MAC. Další pomocnou třídou je `IOFile.java`, která vyvolává dialogy, načítá mapu a pracuje se soubory. `Global.java` je třída, jež definuje statické proměnné. Mezi důležité třídy patří i `MultiScan.java`. Jejím hlavním přínosem je vícenásobné volání skenování dat RSSI a MAC, jež však probíhá na pozadí aplikace. V tomto případě je tedy výpočetní výkon jednotlivých vláken přenosného zařízení rovnoměrně rozložen a nedochází ke zpoždění zpracování uživatelské aktivity. Jednoduše řečeno, aplikace běží plynule. Poté následují dvě třídy, které umožňují práci s databázemi, čili jejich vytváření, zápis a čtení hodnot apod. Jsou jimi `Database.java` a `DatabaseHandler.java`. Bližšímu popisu těchto dvou tříd se však věnovat nebudu, protože jejich autorem je kolega, který se této problematice bude věnovat

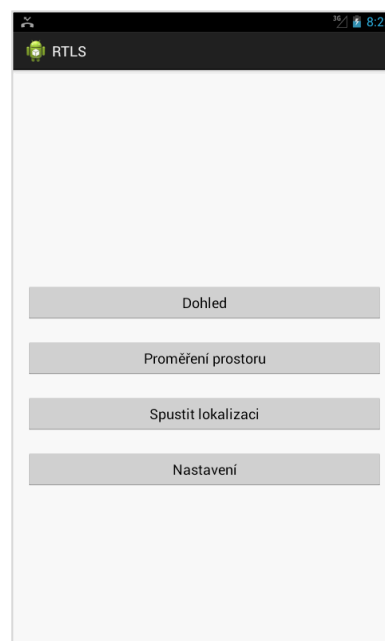
podrobněji. Nicméně, jednotlivé vztahy programových tříd jsou znázorněny na Obr. 5.3, kde aktivity jsou podbarvené modře a pomocné třídy žlutě. Ještě nutno dodat, že pro správný chod aplikace je potřeba verze Android 3.0 (Honeycomb) a vyšší. Je to z důvodu funkčnosti některých využitých funkcí.



Obr. 5.3: Vztahy programových tříd

## 5.5.2 Aktivita RTLS.java

Jak už bylo zmíněno dříve, tato aktivita je základním pilířem celé aplikace, protože se spouští jako první. Její vzhled je definován XML souborem, který obsahuje tlačítka menu pro spouštění dalších aktivit. Z tohoto pohledu jednoduchá aktivita, která však plní jednu důležitou funkci. Jelikož mohou soubory využívané touto aplikací nabývat větších rozměrů, samozřejmě také záleží na jejich počtu, bylo nutné definovat úložiště. Z počátku jsem definoval adresářovou strukturu v interním úložišti, avšak díky zkušenostem kolegy využívajícího OS Android, jsem tuto volbu zrušil. Jedním z důvodů je, že uživatel tohoto systému, nemá právo přistupovat do interní paměti. V tom případě by správce nemohl nahrávat nové mapy, či spravovat databáze, aniž by nemusel využít cesty přes síťovou infrastrukturu. Z tohoto pohledu by to mělo být pro něj jednodušší. Co je však důležitější, spousta aplikací využívá právě interní paměť a pokud není dostatečně velká, tak může docházet ke zbytečným komplikacím. Proto se po spuštění aplikace vytváří adresářová struktura přímo na SD kartě zařízení. Samozřejmě, i když drtivá většina zařízení obsahuje externí paměť, tak může dojít k její absenci. Proto se kontroluje přítomnost SD karty, a pokud není dostupná, tak se musí využít paměti interní, což však není doporučováno. Kořenovým adresářem je tedy /rtls, který obsahuje složky /maps, /surveys, /scale, /localization, /databases. Myslím, že názvy vystihují jejich funkci, tedy



Obr. 5.4: Aktivita RTLS

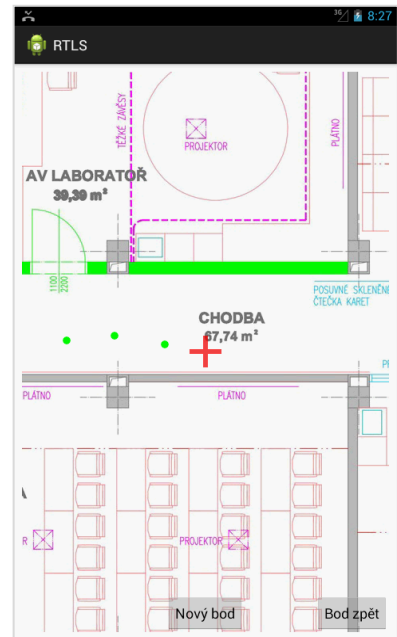


popořadě: ukládání map, souborů trénovací fáze, souborů obsahujících měřítka map, souborů definujících konkrétní mapu vztahenou ke konkrétní databázi pro účely lokalizace a poslední slouží pro ukládání souborů databází. Na *Obr. 5.3* je znázorněna struktura aplikace z pohledu vztahů tříd, můžeme tedy vidět, že zmiňovaná aktivita slouží jako základní rozcestník.

### 5.5.3 Aktivita Survey.java

Pod tímto názvem se skrývá základní funkce aplikace, která slouží pro tvorbu databáze dat. Využívá se tedy ve fázi trénování. Základním prvkem layoutu, tvořícího vizuální stránku aktivity, je tzv. *ImageView*. Je to v podstatě prvek, který vytváří vrstvu na základním layoutu, který je navržen k účelu zobrazení obrázků. V aplikaci je proto využit pro zobrazení mapy. Jelikož se jedná o jednoduchý prvek, tak díky němu není standardně možné s mapou pohybovat, či oddalovat a přibližovat. Pro tuto funkcionalitu jsem musel vytvořit speciální pomocnou třídu *Map.java*. Ta implementuje rozhraní *OnTouchListener*, které detekuje doteky prstů na displej zařízení. Rozpozná tedy dotek a uvolnění jednoho prstu z displeje, popřípadě i dvou prstů zároveň, a rozpozná i posun prstů po obrazovce. Tyto elementy jsou následně využity pro konfiguraci (přepočítání hodnot) transformační matice, která popisuje vrstvu již zmíněného *ImageView*. Výsledkem je plynulý posun mapy, či její přibližování a oddalování. Důležité také bylo vytvořit podmínky, které zabraňují uživateli, aby nemohl posunout mapu mimo daný zobrazovaný prvek. Konkrétně hrany mapy nikdy nepřesáhnou polovinu displeje. Kupříkladu, levá hrana obrazové mapy se může pohybovat pouze v levé polovině displeje, resp. *ImageView*. Je to způsobeno způsobem zadávání nových bodů na mapu, protože uprostřed displeje (horizontálně i vertikálně) je zobrazen středový kříž. Ten definuje, kam se daný bod zakreslí. Kříž je tedy statický a pomocí posunu mapy nastavíme aktuální pozici pod zobrazený kříž. Tím se nastaví souřadnice obrazových bodů dané mapy a na volání dalších metod jsou zpracovány a slouží jako identifikátor pozice, který se ukládá do databáze. Samozřejmě, vše je naprogramováno takovým způsobem, aby pro různé hustoty obrazových bodů jak zařízení, tak samotných obrázků, aby tato funkce byla zcela nezávislá. Souřadnice jsou poté uloženy do statických proměnných ve třídě *Global.java*.

Vraťme se ale k funkci aktivity. Její layout, krom *ImageView*, dále obsahuje dvě tlačítka. Méně využívané je tlačítko, na krok zpět, při zadávání bodů na mapu. Maže tedy poslední zadaný bod. Druhé tlačítko je však důležitější, volá metodu pro zakreslení bodu na mapu, ale krom toho také volá třídu *MultiScan.java*. Třída je blíže popsána v následujícím popisu další aktivity. Jen zde upřesním jednu záležitost. Volání této třídy se neprovádí časovačem, jak bude popsáno u lokalizace, ale pouze jednou po zmáčknutí tlačítka pro nový bod. Nic to však nemění na tom, že je skenování hodnot signálu prováděno také osmkrát. V konečném důsledku je tedy tímto tlačítkem vytvořen záznam pro databázi, avšak ještě není do ní uložen. Tím se dostáváme k hlavnímu menu aktivity, jež obsahuje osm položek. Kromě položky pro návrat do hlavního menu



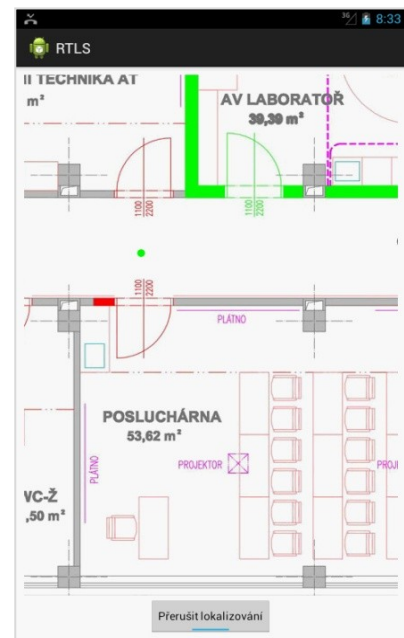
*Obr. 5.5: Aktivita Survey*



aplikace, všechny ostatní využívají metody definované v pomocné třídě `IOFile.java`. Položka *Nové měření*, v podstatě zakládá nový projekt, který je definován mapou a databází. Po výběru položky se vyvolá dialog na výběr mapy, jehož obsah vychází z obsahu adresáře pro uložení obrazových map (adresář `/rtls/maps`). Po výběru mapy se zobrazí další dialog pro vložení jména nového měření. Nyní se už může vytvářet databáze hodnot pomocí tlačítka *Nový bod*, která se vytvoří až po uložení měření (adresář `/rtls/databases`). To můžeme provést položkou v menu *Uložit* anebo *Uložit jako*, kde druhá zmiňovaná samozřejmě vyvolá dialog pro zadání nového jména měření. Po uložení vznikne textový soubor podle jména měření, na jehož prvním řádku je jméno mapy, ke které se soubor vztahuje. Jméno souboru také definuje jméno databáze, která se tedy vztahuje ke konkrétní mapě, jež je uložena v souboru měření. Toho je využito v případě, když z menu vybereme položku *Načíst měření*. Zde se objeví dialog, který obsahuje názvy všech uložených měření (adresář `/rtls/surveys`). Mezi další položky menu patří mazání všech bodů na mapě a mazání celých souborů měření. Důležitá je však zejména položka poslední, a to *Aktivovat měření*. Poté, co byl vytvořen a uložen nový projekt, měření, tak je nutné dané měření aktivovat. Výsledkem je vytvoření textového souboru v adresáři `/rtls/localization`. Soubor tam je vždy pouze jeden, protože pouze jedno měření, tedy mapa prostoru s databází hodnot, může být aktivní. Toho je posléze využito pro uživatelskou aplikaci, kterou se zabývá kolega. Co se týče pohledu programátorského, tak je vše o něco více složitější, jak tedy podrobný popis funkce nejen této aktivity, ale celé aplikace. Cílem práce je však seznámení se základními principy aplikace.

#### 5.5.4 Aktivita `Localization.java`

Třetí položka menu, a zároveň druhá klíčová funkce aplikace, je spuštění lokalizace zařízení. Vzhled aktivity je jednoduchý, celý layout tvoří jeden `ImageView` (viz předchozí aktivita) a tlačítko pro zahájení lokalizace. Aktivita je koncipována tak, že nejprve pomocí hlavního menu aktivity načteme mapu prostoru, která je zobrazena právě v `ImageView`. Jelikož ve fázi trénování byl vytvořen soubor díky položce v menu *Aktivovat měření*, tak po načtení mapy se konkrétní soubor (dle jména mapy) přečte a zjistí se, která databáze bude využita pro účely lokalizace. Poté už jen stačí aktivovat tlačítko pro spuštění lokalizování a na mapě se zobrazí konkrétní pozice v podobě barevné tečky. Z programátorského hlediska je však tato aktivita daleko složitější. Pominu-li práci s mapou a dialogy, třída `Map.java`, resp. `IOFile.java`, tak mezi další pomocné třídy patří `MultiScan.java` a `Scan.java`. Druhá jmenovaná má funkci skenování okolních přístupových bodů. Nejdříve tedy naváže komunikaci s bezdrátovým adaptérem konkrétního zařízení a poté přejímá výsledky, které tento adaptér zjistil z tzv. *beacon rámců*. To jsou rámce, jež přístupový bod vysílá do okolí proto, aby zařízení mohla dané AP „vidět“. Z množství poskytovaných parametrů nám postačí síla signálu a MAC adresa přístupového bodu. Tyto hodnoty jsou filtrovány z naskenovaného listu záznamů



Obr. 5.6: Aktivita `Localization`

všech dostupných AP. Nejprve se tedy vyfiltrují záznamy, jejichž název sítě odpovídá té, ke které je zařízení připojeno. Poté se ze získaných záznamů vyfiltrují potřebná data. Aby byly splněny požadavky pro lokalizační metodu *fingerprinting* (viz kap. *Návrh lokalizačního systému*), je nutné tyto hodnoty seřadit sestupně dle síly signálu. Výsledkem třídy `Scan.java` jsou tedy dvě pole hodnot, síla signálu a MAC adresa, jež každé obsahuje čtyři záznamy. Samozřejmě, jednotlivé pozice polí patří k sobě, tedy popisují jedno AP. Z počátku vývoje aplikace se tato třída, resp. datová pole, využila pro vytvoření záznamu do databáze hodnot. Poté se však objevil faktor ovlivňující přesnost lokalizace, a to kolísání síly signálu. To je pro přístupové body vcelku normální jev, protože využívají různé regulace vyzářeného výkonu a i samotná modulace OFDM přispívá tím, že v závislosti na využití AP, vypíná či zapíná přenosové subkanály v 20 MHz pásmu jednotlivých kanálů. To způsobí situaci, že v různých časových intervalech bude různá síla signálu, a to na stejném místě a od stejného AP. Z tohoto pohledu nastává problém, protože využívaný algoritmus (*Nejbližší soused*, podkap. 3.4), na výpočet pozice, předpokládá, že pokud je vektor z databáze ze všech nejvíce podobný aktuálnímu naměřenému vektoru, tak ho prohlásí za výslednou pozici. Pokud si tedy spojíme pojmy kolísání síly signálu a porovnávání dvou vektorů na stejném místě, avšak v jiný časový interval, tak můžeme usoudit, že vznikne určitá chyba lokalizace. Částečnou kompenzací může být několikanásobné měření síly signálu pro dané místo.

To je implementováno v další pomocné třídě s názvem `MultiScan.java`. Tato třída je odvozená od třídy `AsyncTask`, jež je třída, která umožňuje běh náročnějších procedur na pozadí. Tím je myšleno vytvoření nového vlákna, které běží asynchronně s vláknem uživatelským. Uživatelské vlákno, respektive uživatelské rozhraní, je konkrétně například aktivita, která reaguje na stisk tlačítek a od toho odvozené úkony. Jelikož v našem případě do uživatelského rozhraní spadá práce s mapou, tak při dalším vytížení tohoto vlákna, by mohlo dojít i k pádu aplikace. Proto skenování okolí a zpracování dat je vykonáváno v jiném vlákně na pozadí. Základní princip lokalizace je takový, že po aktivaci tlačítka pro spuštění lokalizace se ve třídě `Localization.java` spustí časovač. Ten je nastaven na šest vteřin, to znamená, že se každých šest vteřin obnoví vypočtená pozice na mapě. Časovač však navíc spouští vykonání třídy vícenásobného skenování tak, že se pomocí cyklu a jeho určitého zpoždění vykonává volání třídy `Scan.java`. Konkrétně se jedná o osm volání. Z tohoto počtu volání se poté získá datové pole hodnot sil signálů a MAC adres. Toto pole je následně přeskládáno tak, aby ve stejném sloupci byly stejné MAC adresy, kde se využívá referenčního záznamu, který je získán jako řádek obsahující nejvíce adres. Po přeskupení pole se pro jednotlivé sloupce sil signálů vypočítá průměrná hodnota signálu, jež vychází z měření osmi hodnot. Tento vektor, který už obsahuje jen čtyři hodnoty signálu a k nim danou MAC adresu, je použit pro porovnání s databází. Hodnoty v databázi byly vytvořeny stejným způsobem, jen s tím rozdílem, že interval skenování je delší (šestnáct vteřin).

Pro výpočet pozice je využita další pomocná třída `DatabaseHandler.java`, jež je kolegovým dílem, ale implementoval jsem zde návrh algoritmu nejbližšího souseda. Jelikož díky fluktuaci signálu se stává, že část naskenovaných hodnot je občas nulová, tak jako první krok je nutné porovnat u vektorů, aktuálně změřeného a z databáze, jejich počet nenulových hodnot, konkrétně MAC adres. Pokud se tyto hodnoty rovnají, tak dochází k porovnávání hodnot MAC adres. Podle počtu shodných adres se nastaví priorita výsledku, který je počítán podle (3.12). Pokud je tedy priorita větší než u předchozího záznamu a pokud je výsledek, tedy rozdílová vzdálenost dvou

vektorů, menší než předchozí, tak dojde k uložení nových pozic, které se vztahují k danému porovnávanému vektoru z databáze. Tímto způsobem se prochází celá databáze a dojde se k výsledným souřadnicím. Ty se poté ve třídě `Localization.java` použijí pro umístění značky na mapu. Ve výsledku je tedy takto obnovována pozice každých šest vteřin. Ještě bych se vrátil k důležitosti zmíněných priorit. Pokud bychom neporovnávali žádným způsobem MAC adresy, docházelo by k situacím, kde by se našlo určité množství stejných výsledků. Je to dáno tím, že se počítá odlišnost vektorů, kde se v databázi mohou najít dva stejné vektory, které se však liší zcela svou pozicí na mapě. Proto je využit nejprve filtr adres, aby se zjistilo, jestli pořadí a popřípadě počet MAC adres je u obou vektorů shodný. Tím se značně sníží šance na chybu metodiky. Priority tedy zaručují situaci, kdy budeme brát v potaz co nejvyšší možný počet shodných MAC adres a tedy jeden z nejbližších vektorů.

### 5.5.5 Aktivita `Settings.java`

Poslední položkou hlavního menu je aktivita pro nastavení parametrů aplikace. Jednotlivé parametry jsou vytvářeny pomocí aktivit. Mezi ně se řadí aktivity pro nastavení jazyka, nastavení měřítka mapy a nastavení serveru se sdílenou složkou. Zaměřím se nejprve na nastavení jazyka aplikace. Je si možno vybrat mezi dvěma variantami jazyka, a to čeština a angličtina. Angličtina je však preferována, z pohledu, když je aplikace na daném zařízení poprvé nainstalována, tak je automaticky vybrán právě anglický jazyk. Pro dosažení různých jazykových lokalizací bylo možno využít tři různá řešení. Prvním je vytvoření duplicitních řetězců (z angl. *string* - datový typ program. jazyka pro uložení znaků) v různých jazycích, které popisují jednu danou věc. Podle nastaveného jazyka prostředí v zařízení, který si každý uživatel sám volí, se poté automaticky využívají české nebo anglické řetězce. Nevýhoda je však v tom, že je aplikace striktně vázána na jazyk zařízení. Druhá možnost je změna jazyka aplikace tak, že se změní jazyk zařízení přímo z aplikace. Tím by však byly postiženy všechny ostatní aplikace a samozřejmě to ani není doporučováno. Využil jsem tedy možnost třetí. Ta je obtížnější, avšak aplikace je zcela nezávislá na jazyku prostředí v zařízení, uživatel si tedy může libovolně jazyk měnit, a také není nijak zasahováno do nastavení zařízení. Princip je takový, že každá aktivita je definována jak tedy samotnou logikou, tak zároveň soubory (layouty) XML, o nichž jsem se už také zmínil. Proto pro každou aktivitu jsou vytvořeny dva layouty, český a anglický. Za zmínku také stojí to, že pokud uživatel vybere jazyk, tak jeho hodnota se zapíše do proměnné, odvozené od třídy *SharedPreferences*, která si hodnotu pamatuje i po ukončení aplikace. Tato hodnota se porovnává na začátku každé aktivity a není tedy nutno po opětovném spuštění aplikace nastavovat jazyk, anebo aplikaci po nastavení jazyka restartovat.

Druhou položkou menu v nastavení je konfigurace měřítka. Ta je momentálně funkční, avšak její výsledek se prozatím nepoužívá, protože bude mít smysl až pro případný další vývoj, a to navigace uživatele v prostředí budov. Ta by se měla lišit od



Obr. 5.7: Aktivita `Settings`

stávající lokalizace hlavně v tom, že bude určovat reálnou vzdálenost k nějakému referenčnímu bodu. Pokud tedy například chceme najít konkrétní zboží v prostoru obchodu, tak navigace zobrazí cestu a její délku. Samozřejmě takovýchto příkladů je mnoho. Nicméně zadání měřítka je založeno na principu, kdy se načte požadovaná mapa. Poté se zadá první bod a po zadání druhého se zobrazí dialog pro vložení reálné vzdálenosti těchto bodů, která je poté vztažena ke vzdálenosti bodů na mapě. Poslední položkou menu je nastavení sítě, která slouží pro nastavení IP adresy, portu, uživatelského jména, hesla daného FTP serveru a poslední je SSID bezdrátové sítě.

Zde je však nutno dodat, že je aplikace stále ve vývoji, a proto je stále obtížnější aktualizovat její podobu v tomto textu. Proto se ve výsledku může popis a návod v příloze E v určitých částech odlišovat od verze na přiloženém CD, kde se nachází poslední možná verze k datu odevzdání této práce. Je také nutno dodat, že ve výsledku vznikly dvě aplikace správce, a to *RTLS Demo* a *RTLS Admin*. Základem je prvně zmiňovaná, jejíž princip jsem také výše popsal. Je tedy nezávislá na konektivě se serverem, a proto může sloužit jako měřicí či optimalizační pomůcka. Samozřejmě plní funkci tvorby databáze a následné lokalizace. Její hlavní výhodou je, že po příchodu do daných prostor, stačí zadat SSID sítě a hned může být vytvořena databáze a spuštěna lokalizace. Druhá aplikace je doplněna o komunikaci s FTP serverem a dohledovou částí (lokalizace uživatelů na dálku), její nasazení je tedy podmíněno existencí FTP serveru a jeho správné konfigurace.

## 5.6 Zhodnocení přesnosti lokalizačního systému

### 5.6.1 Optimalizace přesnosti lokalizace

Cílem kapitoly bylo zjistit, jaký výpočetní algoritmus dosahuje nejpřesnějších výsledků. Měření vychází ze třiceti početních intervalů pozice pro každý algoritmus. Skutečná pozice byla zaujata v bodě  $[0,0]$ , viz *Graf 5.1*. Největší hustota bodů kolem skutečné pozice je v zastoupení algoritmu WKNN, jehož průměrná chyba vzdálenosti (vztah 5.16) dosahuje nejmenší hodnoty, viz *Tab. 5.1*. Z tohoto pohledu bude dosavadní metoda NN nahrazena právě metodou WKNN. Databáze hodnot byla tvořena sítí měřících bodů, jejichž interval vzdálenosti byl jeden metr v obou osách. Z tohoto pohledu bylo měření ovlivněné hustotou bodů, protože při její větší hustotě (menší vzdálenost mezi body) by byla dosažena větší přesnost. Jeden metr jsem však zvolil jako kompromis, protože si musíme uvědomit, že při velice husté síti (řády decimetrů) je rozlišitelnost pro technologii Wi-Fi velice nízká. Samozřejmě, z pohledu počtu vzorků databáze, by měl algoritmus k porovnání více hodnot, avšak z pohledu geografického, tedy například čtverce  $(20 \times 20)$  cm, je signál pro každý jeho bod téměř konstantní, a můžeme ho tedy nahradit jedním bodem.

$$E = \sqrt{(x - \bar{x})^2 + (y - \bar{y})^2} \quad (5.16)$$

kde  $x, y$  je skutečná pozice a  $\bar{x}, \bar{y}$  je pozice určená systémem.

Rozptyl vypočtených hodnot je dán časovým kolísáním hodnot signálu, které slouží jako vstupní parametry výpočtu. Problematikou kolísání signálu se zabývá ve své práci kolega. Nicméně, samotné určování pozičních bodů vykazovalo znaky konvergence ke skutečné pozici. Ta však byla způsobena pouze změnou hladin signálů, tedy pokud tyto hodnoty v daném časovém intervalu více odpovídaly hodnotám v databázi, tak bylo postupně dosahováno lepších výsledků. Konvergence však nebyla stabilní, protože i po

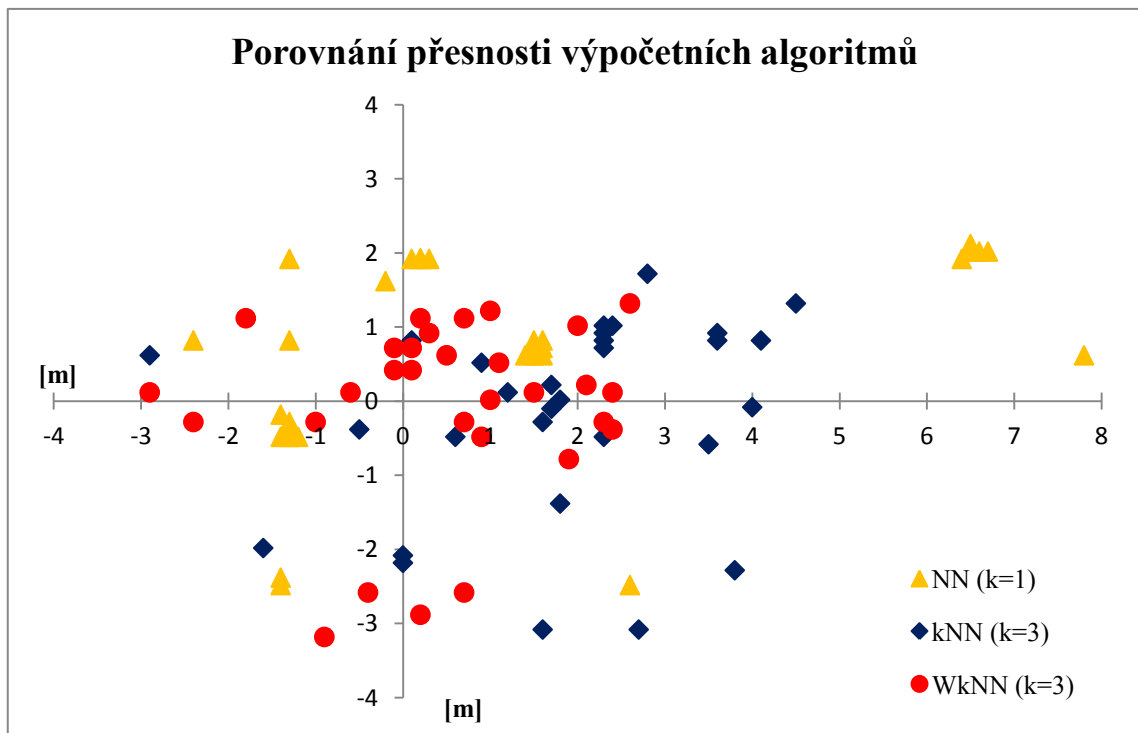
nastavení konstantního výkonu na přístupových bodech docházelo k fluktuacím signálu důsledkem vícecestného šíření. Díky tomu, že metody KNN a WKNN pracují s větším množstvím nejmenších sousedů, ne tedy pouze s jedním jako metoda NN, tak obecně dosahují lepších výsledků.

Důvodem je částečná kompenzace kolísání signálu, protože hodnoty v databázi mohou být ovlivněny kolísáním, tzn. že vektor pro daný bod obsahuje velice odlišné hodnoty oproti vektorům (bodům) sousedním. Pokud se tedy ve výpočtu pracuje s více než jedním bodem pro pomyslné okolí skutečné pozice, může se jeden chybný vektor databáze tímto způsobem eliminovat. Eliminace chyb databáze počtem nejbližších sousedů, které jsou použity pro výpočet, však jde zlepšovat jen do určité míry.

Tab. 5.1: Porovnání přesnosti jednotlivých algoritmů

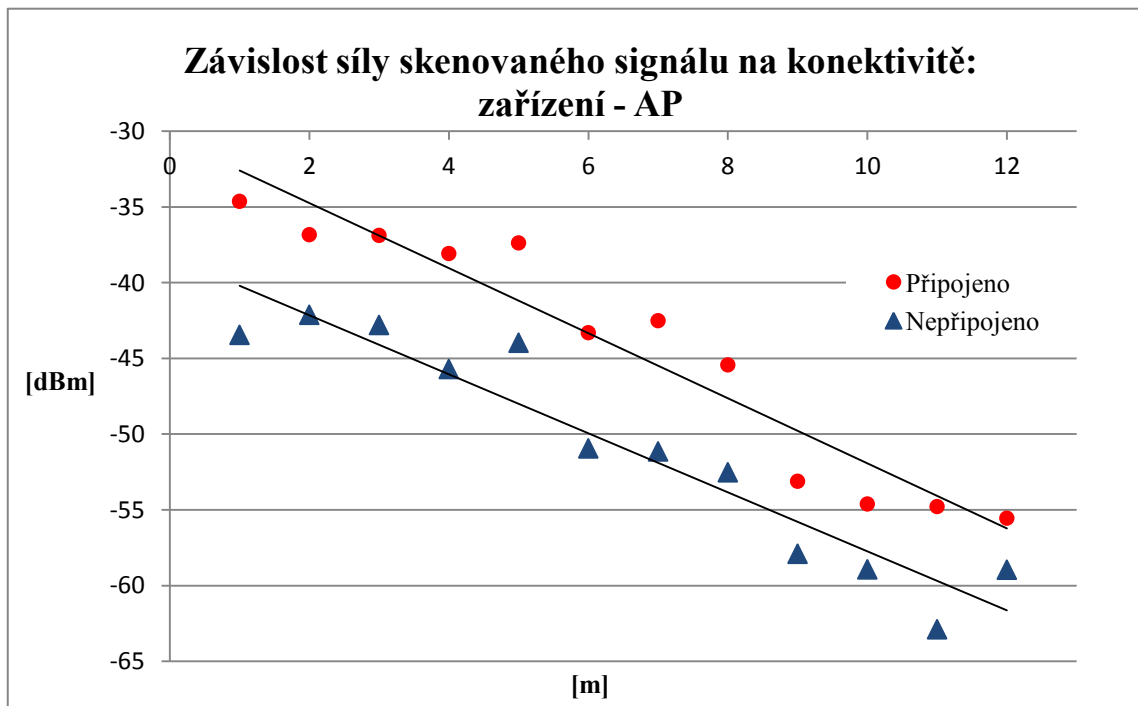
Algoritmus	Průměrná chyba vzdálenosti $E$ [m]
NN (k=1)	2,90
KNN (k=3)	2,58
WKNN (k=3)	1,69

Optimální počet zpracovaných nejbližších sousedů se pohybuje od tří do pěti, protože při větších počtech se počítá i s těmi hodnotami, které jsou vzdálenější skutečné pozici, a mohou tedy způsobovat chybu přesnosti. Zejména to může platit pro menší či užší prostory, jež jsou v databázi popsány menším počtem bodů. Při velkém počtu  $k$ , by v následném výpočtu mohly být brány i vzorky, které odpovídají vedlejší místnosti či místům hodně vzdáleným. Jelikož je  $k$  neměnná konstanta, určitá optimalizace, pro takto členitá prostředí, by spočívala v dynamickém určování  $k$ -nejbližších sousedů. V popisovaném systému je aktuálně využito fixní proměnné  $k = 3$  (viz Graf. 5.1).



Graf 5.1: Porovnání výpočetních algoritmů výsledné pozice

Další optimalizace systému spočívala v konektivě mezi přenosným zařízením a přístupovým bodem (viz *Graf. 5.2*). Graf zobrazuje sílu signálu a počet metrů od AP. Konkrétně se jedná o řez prostoru, kde jsou patrné tři místnosti, jež jsou rozpoznatelné dle podobných hladin signálu. Důležitější je však rozdíl mezi hladinou signálu pro připojení a nepřipojení k síti. Pokud se zařízení připojí k síti, tedy k AP s nejvyšší dostupnou hladinou signálu, tak dochází do jisté míry k automatickému zvyšování výkonu vyslaného signálu samotným AP. Toto je však nežádoucí z pohledu tvorby databáze, protože takto je zařízení připojeno po celém prostoru k jednomu AP, protože přepojení na jiné AP je podmíněno velmi nízkou dostupnou hladinou signálu. Pokud později dojde k připojení k síti, avšak k jinému AP, hodnoty pro konkrétní místo, z pohledu konkrétního AP, jsou zcela odlišné od hodnot zaznamenaných v databázi. Aplikace tedy zapíná Wi-Fi adaptér, avšak nepřipojuje se k dané síti.



*Graf 5.2: Závislost skenovaného signálu na konektivě: zařízení - AP*

### 5.6.2 Výsledná přesnost lokalizačního systému

V obecném prostředí je přesnost navrhovaného systému ovlivněna těmito faktory:

- 1) kolísání signálu z pohledu regulace výkonu samotným AP
  - pokud není možno regulaci vypnout
- 2) kolísání signálu z pohledu vícecestného šíření
- 3) hustota naměřených bodů
  - závisí na správci systému
- 4) dynamická změna prostředí
  - pohyb osob, věcí, otevřené dveře apod.
- 5) různá kalibrace Wi-Fi adaptérů v přenosných zařízeních
  - dvě různá zařízení naskenují různé hladiny signálu
- 6) členitost prostředí
  - z pohledu výskytu více či méně rozlehlých prostor (místnosti, chodby)

Stanovení výsledné přesnosti lokalizačního systému jsem provedl pomocí čtyř měření, kde byly jednotlivé body databáze z trénovací fáze vytvářeny s metrovým odstupem. Přístupové body měly nastaveny konstantní vysílací výkon. Z výše zmíněných faktorů bylo zčásti omezeno vícecestné šíření signálu zavřenými dveřmi a byl použit jeden typ mobilního zařízení. Samotné měření probíhalo ve čtyřech etapách, kde každá obsahovala čtyřicet intervalů výpočtu pozice. Výsledky jsou uvedeny v *Tab. 5.2* a vizualizovány v *Příloze C*. Nicméně, prvním výsledkem je úspěšnost systému v určování správné místnosti (prostoru), odkud bylo měření prováděno. Ta byla stanovena jako podíl úspěšných určení a celkového počtu pokusů (40). Dílčí výsledky si jsou velice podobné, což částečně nepotvrdilo můj předpoklad, že pro velké prostory, jako je celý Smart Home a komplex tří propojených místností, kam patří JIP, bude systém určovat pozice ve větší vzdálenosti od skutečné pozice, avšak ve stejném prostoru. V těchto rozlehlých členitých prostorech se signál šíří velice podobně, a proto mohla být ovlivněna rozlišitelnost konkrétní místnosti v daném prostoru. Naopak se potvrdilo, že rozlišitelnost byla velice nízká pro místnost posluchárny. Ta není členitá, avšak řadí se také mezi větší prostory, a proto zde byl rozptýl určených bodů také nejvyšší.

Co se týče průměrné chyby vzdálenosti mezi skutečnou a určenou pozicí, tak pro posluchárnu, z výše zmíněného, vyplívá i největší chyba. Ostatní chyby vzdálenosti jsou srovnatelné. Na základě dostupných výsledků mohu usoudit, že vytvořený lokalizační systém spadá spíše do stupně lokalizace na úrovni pokoje. Ze zkušeností získaných při tvorbě této práce, bych usoudil, že vytvořit systém na bázi WLAN pokrytí je vcelku náročný úkol. Jedná se o množství faktorů, které výsledky lokalizace mohou ovlivnit. Jak už bylo zmíněno u popisu RSSI metody pro získávání dat, tato metoda je lehce ovlivnitelná změnou signálu v daném místě a v daném časovém intervalu. Nejlepší volbou pro zpřesnění, ale hlavně pro zkvalitnění časové stálosti, by bylo vhodné tento systém propojit s jinou technologií, pracující na podobném principu, avšak s využitím jiných metod. Jedná se hlavně o zpřesnění lokalizace v samotném pokoji. Jak jsem už také jednou zmínil, bezdrátové pokrytí WLAN sítě neumožňuje tak dokonalou rozlišitelnost v řádu do dvou až tří metrů v prostoru jednoho pokoje, proto se systémy na tomto základě mohou spíše použít pro určování pokoje, části pokoje, ale prozatím není možné očekávat precizní lokalizaci, aniž by jednotlivé místnosti nebyly vybaveny dalšími podpůrnými zařízeními. Na druhou stranu je tento systém schopný využít stávající síťové infrastruktury, ovšem musí být splněna podmínka dostupnosti nejméně tří vysílačů pro každý bod prostoru.

*Tab. 5.2: Výsledný charakter přesnosti lokalizačního systému*

Měřená místnost*	Úspěšnost [%]**	Průměrná chyba vzdálenosti $E$ [m]
Smart Home	82,50	2,11
JIP	80,00	2,77
AV laboratoř	70,00	2,49
Posluchárna	80,00	3,24
Výsledný průměr	<b>78,75</b>	<b>2,65</b>
<i>*Vybrané místnosti z prostor univerzity, kde probíhal vývoj systému</i>		
<i>** Úspěšnost lokalizace z pohledu správného stanovení místnosti</i>		

## 6. Závěr

Osobním přínosem této práce je zejména získání praktických zkušeností z oboru analýzy pokrytí, konkrétně metodikou měření spektrálním analyzátozem. Další praktické zkušenosti jsem získal i při návrhu a následném programování mobilní aplikace, jejíž funkčnost byla potvrzena testy v kapitole 5.6.

Abych se ale konkrétněji vyjádřil k hlavním cílům práce. Analýza pokrytí byla založena na metodice proměření prostoru spektrálním analyzátozem. Základní a jedinou měřenou hodnotou byla síla signálu pro každý bod prostoru. Jednotlivé body byly definovány maticovou mřížkou a jejich rozestupy v obou osách byly dva metry. Základním smyslem analýzy pokrytí bylo ověření parametrů pro nasazení lokalizačního systému. Mezi tyto parametry patří zejména plocha, kterou jednotlivé přístupové body (vysílače) pokrývají signálem, a s tím související síla signálu v daném místě (viz *Příloha A, B*). Z plochy pokrytí vychází parametr, který udává počet dostupných vysílačů pro každý bod prostoru. Například z *Obr. 4.12* můžeme usoudit, že podmínka dosažení minimálně třech vysílačů je splněna v celém prostoru, kromě části toalet. Vizualizace však bere v úvahu sílu signálu  $-65$  dBm a vyšší, nicméně i v těchto místech je dostupný potřebný počet vysílačů, avšak s nižší hodnotou signálu. Co se týče zbývajících prostor, jelikož se pokryté oblasti od jednotlivých vysílačů překrývají, vznikají tím místa, kde spolu kanály vysílačů interferují (viz *Obr. 4.10*). Zde je vyneseno počet dvojic vysílačů, jež se vzájemně ovlivňují. Jelikož je využito pásmo 2,4 až 2,5 GHz, kde se nacházejí pouze tři kanály, které se nepřekrývají, a se skutečností, že musíme využít šest vysílačů, není tedy reálně dosažitelné, aby k interferencím nedocházelo. Navíc daný prostor je silně zarušen okolními sítěmi, proto jsem měřil prostor i pomocí Wi-Fi USB adaptéru, abych mohl upřesnit závěrečné vyhodnocení. I přes tyto komplikace je však prostor využitelný pro navrhovaný lokalizační systém.

Je využitelný hlavně díky principu, na kterém lokalizační systém pracuje. Zvolil jsem metodu pozičních otisků. To znamená, že se v trénovací fázi, pomocí aplikace správce, projde prostor a pro jeho každý bod (maticová mřížka) se do databáze uloží vektor (sada hodnot), který je definován čtyřmi hodnotami RSSI, příslušnými MAC adresami a souřadnicemi měřeného bodu vztažených k mapě prostoru. Ve fázi vlastní lokalizace se poté porovnávají aktuálně naměřené vektory uživatelskou aplikací s vektory z databáze. Výsledkem porovnání je získání třech nejméně odlišných vektorů (nejbližších sousedů), které jsou posléze metodou WKNN zpracovány ve výslednou pozici.

Výsledná přesnost lokalizace je uvedena v *Tab. 5.2*. Systém tedy dosahuje téměř 80% úspěšnosti ve stanovení správné místnosti, kde se uživatel nachází. Průměrná chyba vzdálenosti je poté 2,65 metru. Jak už jsem ale zmínil v předchozí kapitole, rozlišitelnost pozice na bázi WLAN pokrytí je v řádu do dvou až tří metrů velmi nízká. Nicméně, systém bych zařadil do stupně lokalizace na úrovni pokoje, v lepším případě i části pokoje, kde jeho výhodou je snadná implementace do prostředí, pokud je tedy každý bod prostoru pokryt nejméně třemi vysílači.

Výsledkem této práce jsou také dvě aplikace pro správce, a to *RTLS Demo* a *RTLS Admin*. První je nezávislá na existenci serveru, a proto může sloužit jako měřicí či optimalizační pomůcka. Nicméně, plní funkci tvorby databáze a následné lokalizace. Její hlavní výhodou je, že po příchodu do daných prostor, se pouze zadá SSID síť a ihned se může vytvářet databáze a započít samotná lokalizace. Druhá aplikace je doplněna o komunikaci s FTP serverem a dohledovou částí (autorem kolega). Její nasazení je však podmíněno existencí FTP serveru a jeho správné konfigurace. Tato



aplikace je základním prvkem lokalizačního systému, protože umožňuje tvorbu databáze a její distribuci na FTP server. Dále umožňuje i dohled, tedy vzdálenou lokalizaci uživatelů. Pro úplnost celého řešení, jsem na přiložené CD umístil i zbývající uživatelské aplikace, a to *RTLS Localization Service*, což je aplikace v podobě systémové služby na pozadí (pro následný dohled), a aplikace *RTLS Localization Client*, která zobrazuje polohu přímo v uživatelském zařízení, podobně jako u *RTLS Demo (RTLS Admin)*.

Vývoj mobilní aplikace *RTLS Admin*, resp. celého lokalizačního systému, jehož je součástí, není zcela dokončen. Osobně uvažuji o zdokonalení výpočetního algoritmu za pomoci dalších metod, v tomto ohledu se nabízí například využití pravděpodobnosti, respektive její kombinace s aktuálně využívanými metodami. Je také nutné vytvořit algoritmus, který by eliminoval různorodost bezdrátových adaptérů z hlediska jejich zisku přijímané úrovně signálu. S tím souvisí i algoritmus pro kvalitnější tvorbu databáze, který by mohl část vzorků, na základě parametrů prostředí, dopočítat, mazat duplicitní hodnoty a předcházet chybným hodnotám v konkrétních bodech prostoru.

## Seznam použité literatury

- [1] CHANDRA, P. - M. DOBKIN, D. Basics of Wireless Local Area Networks. V: *Wireless networking: Know It All*. Burlington: Elsevier/Newnes, ©2008, str. 47 - 95. ISBN 0750685824.
- [2] Wi-Fi: definition. *Oxford Dictionaries* [online]. [cit. 2014-03-05]. Dostupné z: <http://www.oxforddictionaries.com/definition/english/Wi-Fi>
- [3] RECOMMENDATION SM. 1056-1. *Limitation of radiation from industrial, scientific and medical (ISM) equipment*. ITU-R, 1997 - 2007. Dostupné z: <http://www.itu.int/rec/R-REC-SM.1056-1-200704-I/en>
- [4] ERC RECOMMENDATION (70-03). *Relating to the Use of Short Range Devices (SRD)*. Electronic Communications Committee, 2014. [online]. [cit. 2014-03-05]. Dostupné z: <http://www.erodocdb.dk/docs/doc98/official/pdf/rec7003e.pdf>
- [5] LEE, B. Gi - CHOI, S. *Broadband wireless access and local networks: Mobile WiMax and WiFi*. London: Artech House, ©2008, str. 36. ISBN 1596932937.
- [6] LEE, B. Gi - CHOI, S. *Broadband wireless access and local networks: Mobile WiMax and WiFi*. London: Artech House, ©2008, str. 394. ISBN 1596932937.
- [7] VO-R/12/09.2010-12. *Využívání rádiových kmitočtů a k provozování zařízení pro širokopásmový přenos dat v pásmech 2,4 GHz až 66 GHz*. Praha: Český telekomunikační úřad, 2010. [online]. [cit. 2014-03-05]. Dostupné z: [http://www.ctu.cz/cs/download/oop/rok\\_2010/vo-r\\_12-09\\_2010-12.pdf](http://www.ctu.cz/cs/download/oop/rok_2010/vo-r_12-09_2010-12.pdf)
- [8] Report M. 2227. *Multiple Gigabit Wireless Systems in frequencies around 60 GHz*. Geneva: ITU-R, 2011. [online]. [cit. 2014-03-05]. Dostupné z: [http://www.itu.int/dms\\_pub/itu-r/opb/rep/R-REP-M.2227-2011-PDF-E.pdf](http://www.itu.int/dms_pub/itu-r/opb/rep/R-REP-M.2227-2011-PDF-E.pdf)
- [9] ETSI EN 302 567. *Broadband Radio Access Networks (BRAN); 60 GHz Multiple-Gigabit WAS/RLAN Systems: Harmonized EN covering the essential requirements of article 3.2 of the R&TTE Directive*. ETSI, 2011. [online]. [cit. 2014-03-05]. Dostupné z: [http://www.etsi.org/deliver/etsi\\_en/302500\\_302599/302567/01.02.01\\_30/en\\_302567v010201v.pdf](http://www.etsi.org/deliver/etsi_en/302500_302599/302567/01.02.01_30/en_302567v010201v.pdf)
- [10] MALIK, A. *Rtls for dummies(r)*. 1st ed. Indianapolis, Wiley Pub., Inc., 2009. ISBN 04-703-9868-X.
- [11] Prof. Ing. MAZÁNEK, M., CSc. - Doc. Ing. PECHAČ, P., Ph.D. *Šíření elektromagnetických vln a antény*. Vyd. 2. přepracované. Praha: Vydavatelství ČVUT, 2004. ISBN 8001030326.
- [12] Prof. Ing. MAZÁNEK, M., CSc. - Prof. Ing. PECHAČ, P., Ph.D. - Prof. Ing. Vrba, J., CSc. *Základy antén, šíření vln a mikrovlnné techniky*. Vyd. 1. Praha: Vydavatelství ČVUT, 2008. ISBN 8001039978.

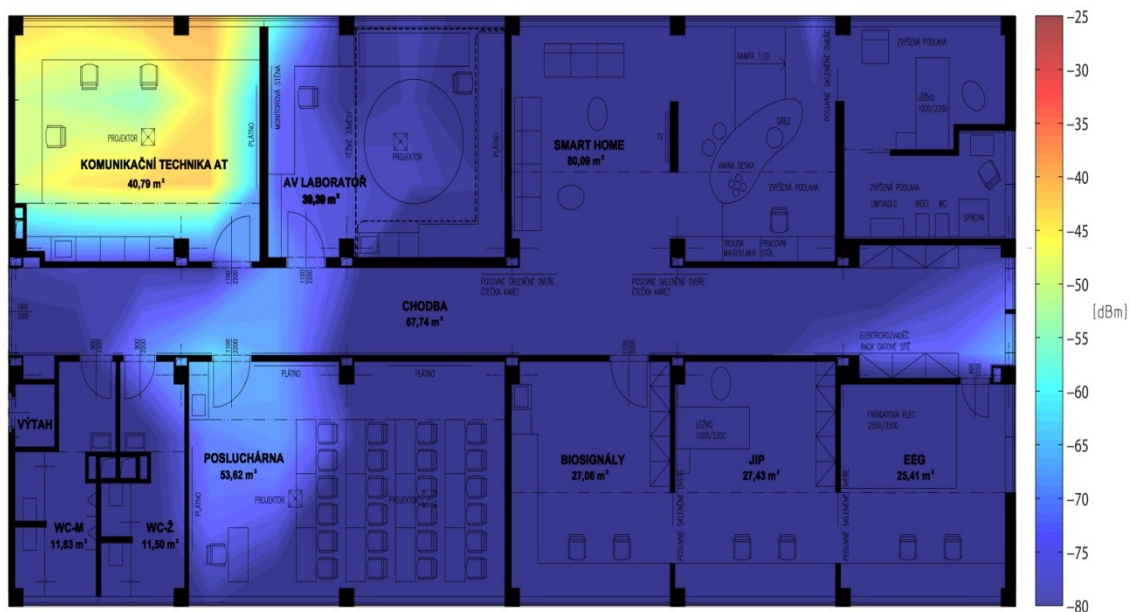
- [13] A-level Physics (Advancing Physics)/ Radar and Triangulation. *WIKIBOOKS* [online]. 2012 [cit. 2014-03-09]. Dostupné z: [http://en.wikibooks.org/wiki/A-level\\_Physics\\_\(Advancing\\_Physics\)/Radar\\_and\\_Triangulation](http://en.wikibooks.org/wiki/A-level_Physics_(Advancing_Physics)/Radar_and_Triangulation)
- [14] KHODAYARI, S. - MALEKI, M. - HAMED, E., *A RSS-based fingerprinting method for positioning based on historical data, Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2010 International Symposium on*, str. 306, 310. [online]. ©2014 [cit. 2014-03-09]. Dostupné z: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=5589221&queryText%3D+RSS-based+fingerprinting+method+for+positioning+based+on+historical+data>
- [15] *Dokumentace Ekahau*. [online]. [cit. 2014-03-12]. Dostupné z: <http://www.ekahau.com/>
- [16] DING, G. - ZHANG, J. - ZHANG, L. - Tan, Z., *Overview of received signal strength based fingerprinting localization in indoor wireless LAN environments, Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications (MAPE), 2013 IEEE 5th International Symposium on*, str. 160, 164, 29-31 Oct. 2013. doi: 10.1109/MAPE.2013.6689973
- [17] Dashboards. *Developer Android* [online]. [cit. 2014-03-14]. Dostupné z: <http://developer.android.com/about/dashboards/index.html>

## Seznam příloh

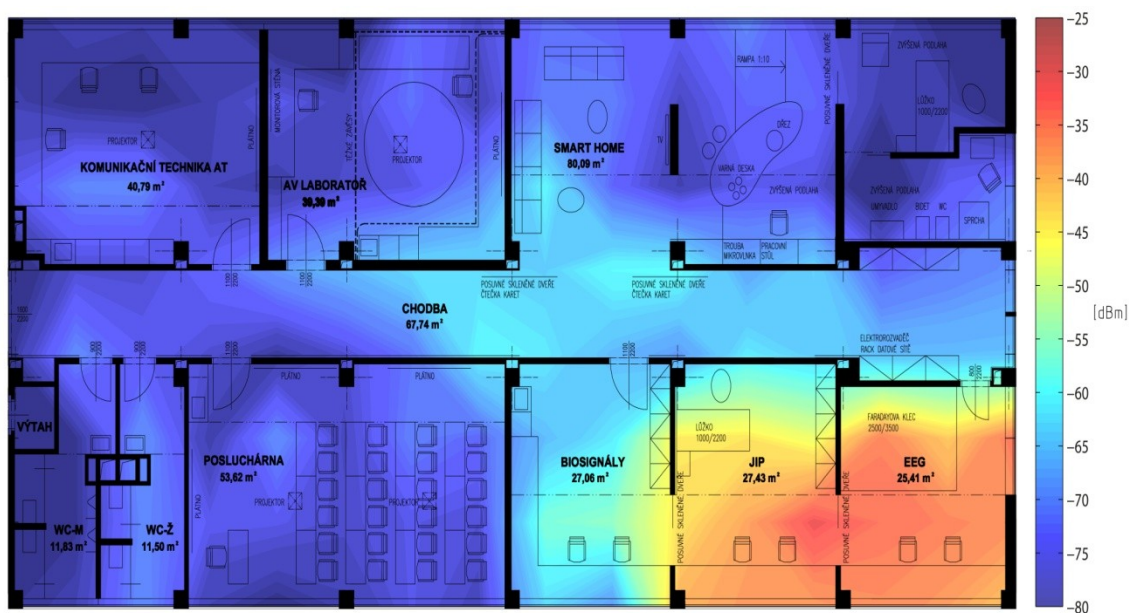
Příloha A - Vizualizace pokrytí jednotlivých vysílačů z měření spektr. analyzátozem	45
Příloha B - Vizualizace pokrytí jednotlivých vysílačů z měření USB adaptérem .....	48
Příloha C - Vizualizace přesnosti stanovení pozice lokalizačním systémem .....	51
Příloha D - Obsah příloženého CD .....	52
Příloha E - Návod pro obsluhu mobilní aplikace <i>RTLS Demo (RTLS Admin)</i> .....	53

# Příloha A

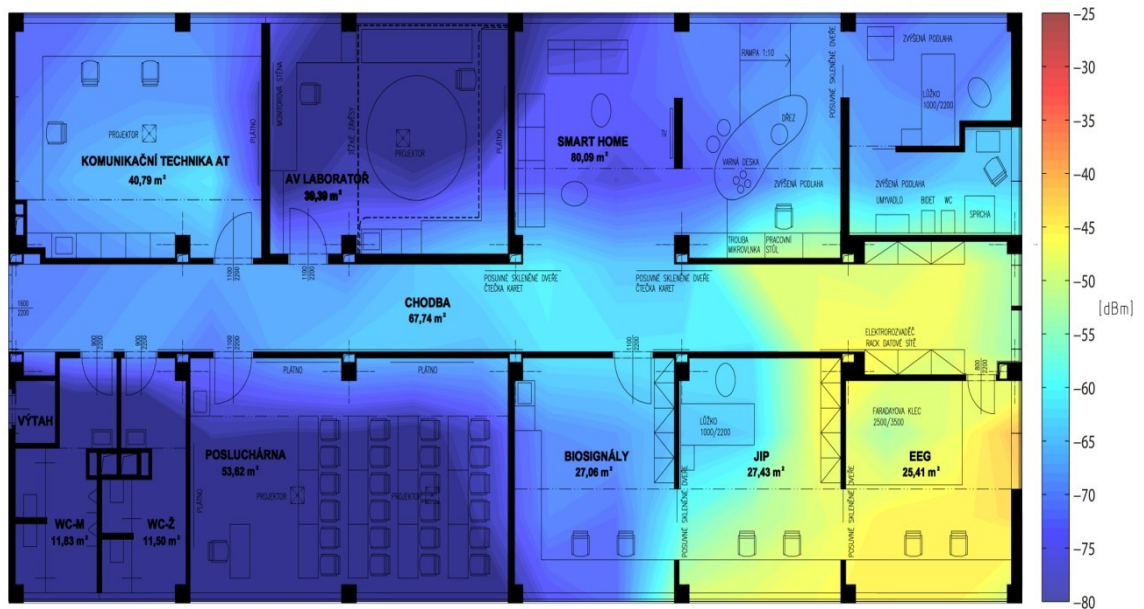
Vizualizace pokrytí jednotlivých vysílačů z měření spektr. analyzátořem



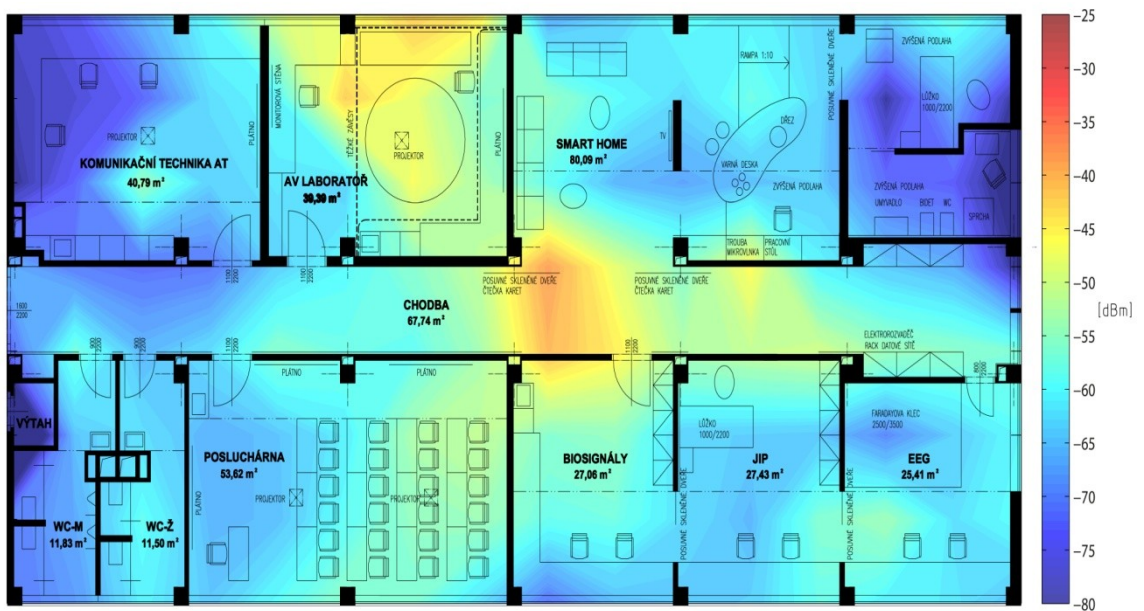
Obr. A. 1: Pokrytí pro kanál 2



Obr. A. 2: Pokrytí pro kanál 3

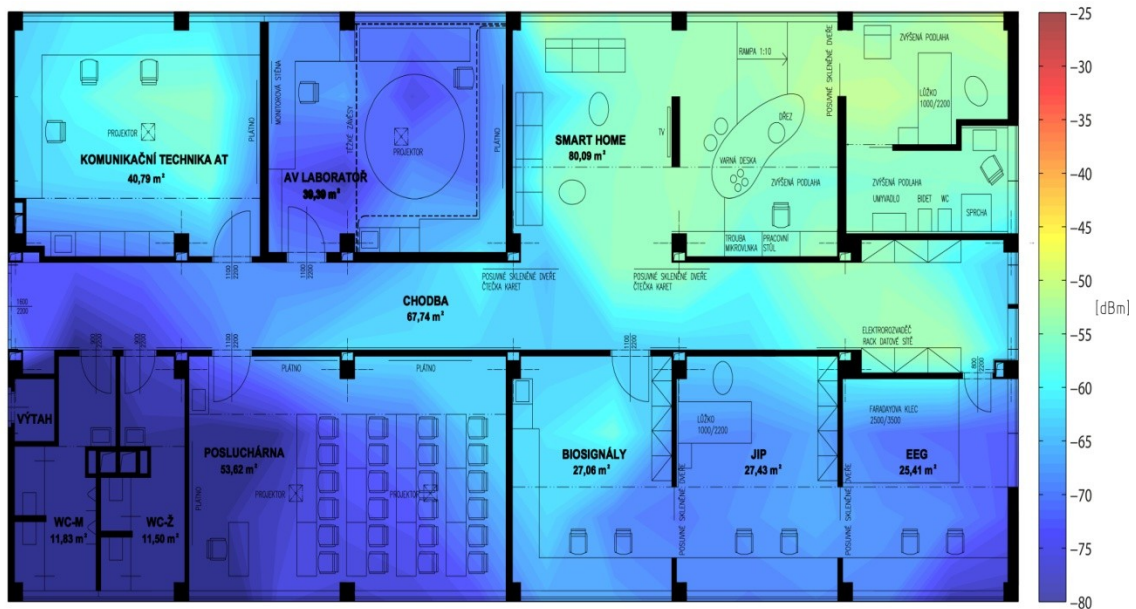


Obr. A. 3: Pokrytí pro kanál 6

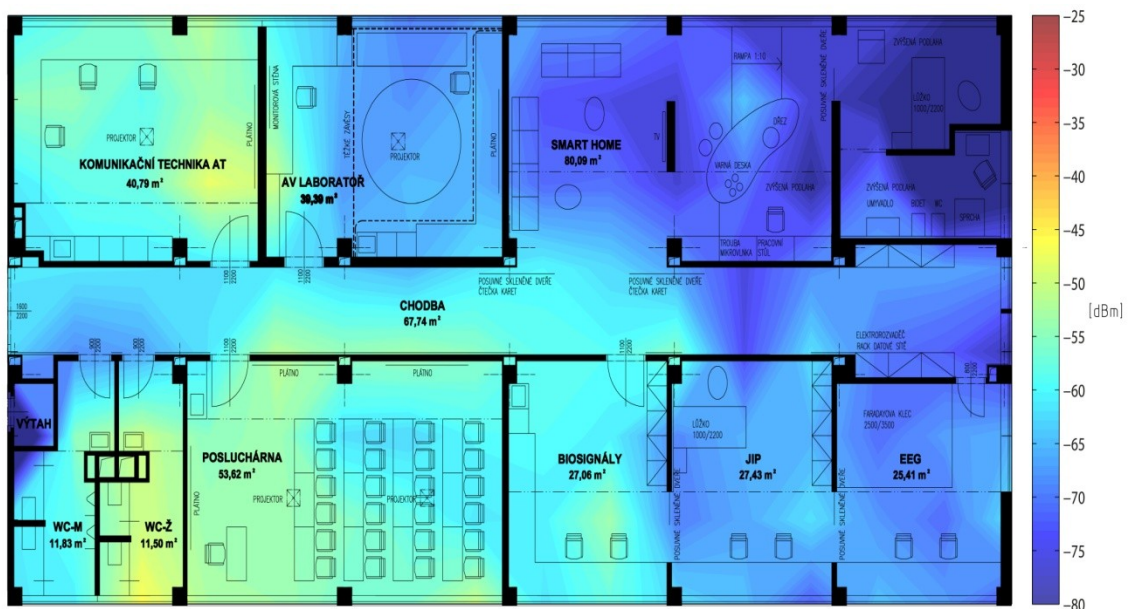


Obr. A. 4: Pokrytí pro kanál 9





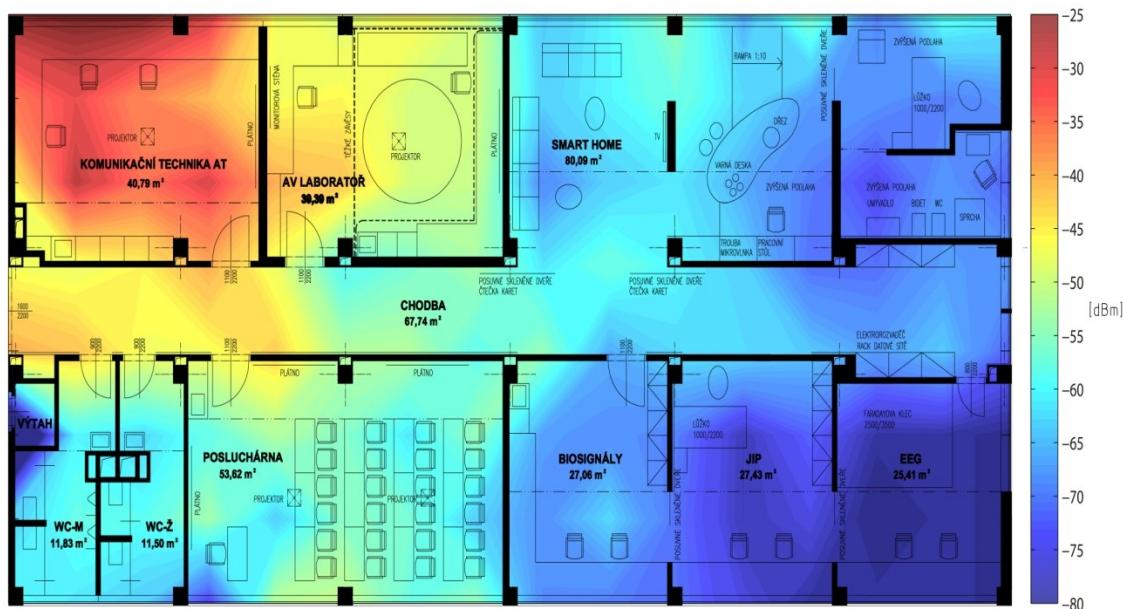
Obr. A. 5: Pokrytí pro kanál 12



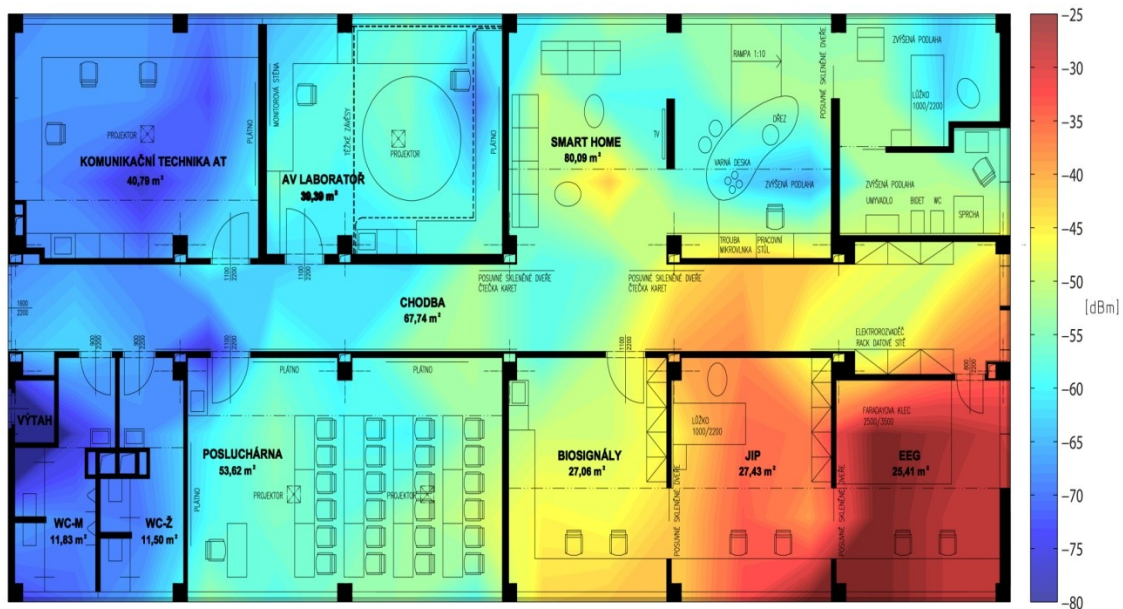
Obr. A. 6: Pokrytí pro kanál 13

# Příloha B

Vizualizace pokrytí jednotlivých vysílačů z měření USB adaptérem

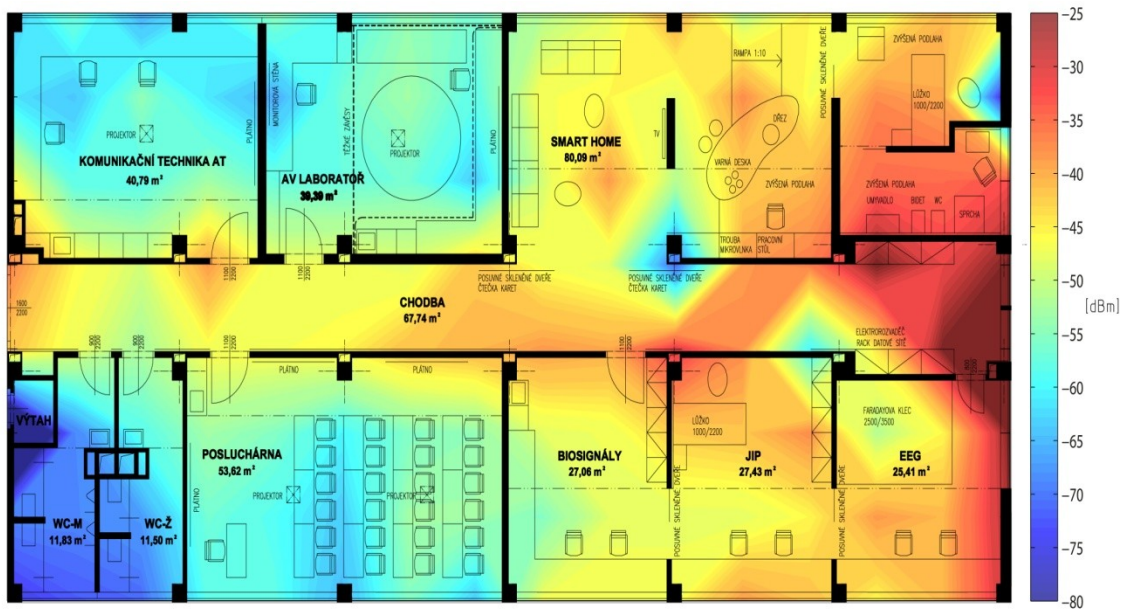


Obr. B. 1: Pokrytí pro kanál 2

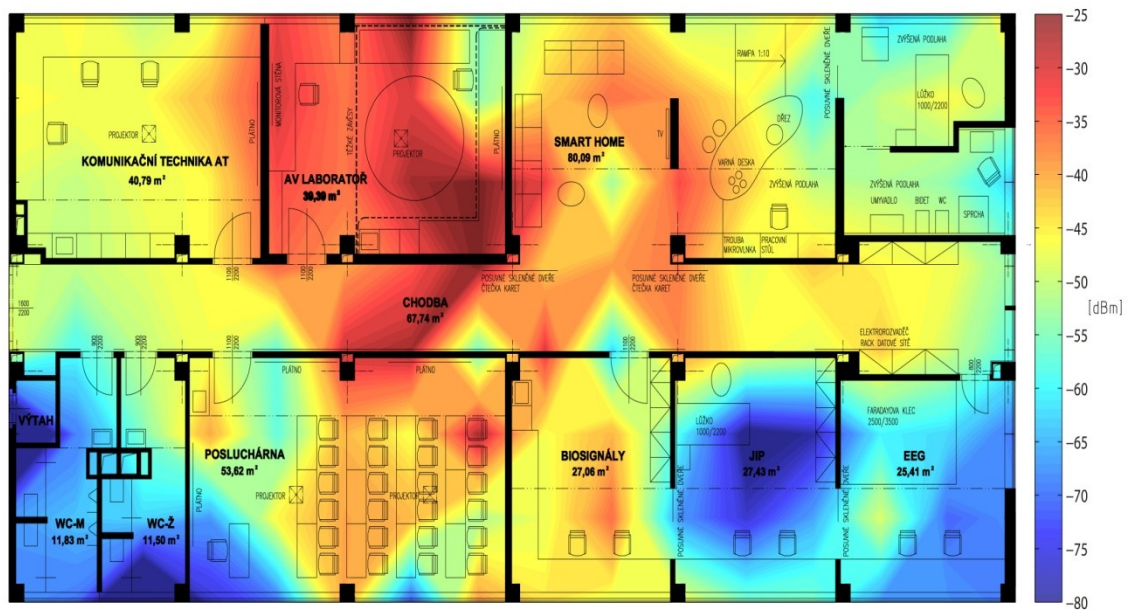


Obr. B. 2: Pokrytí pro kanál 3

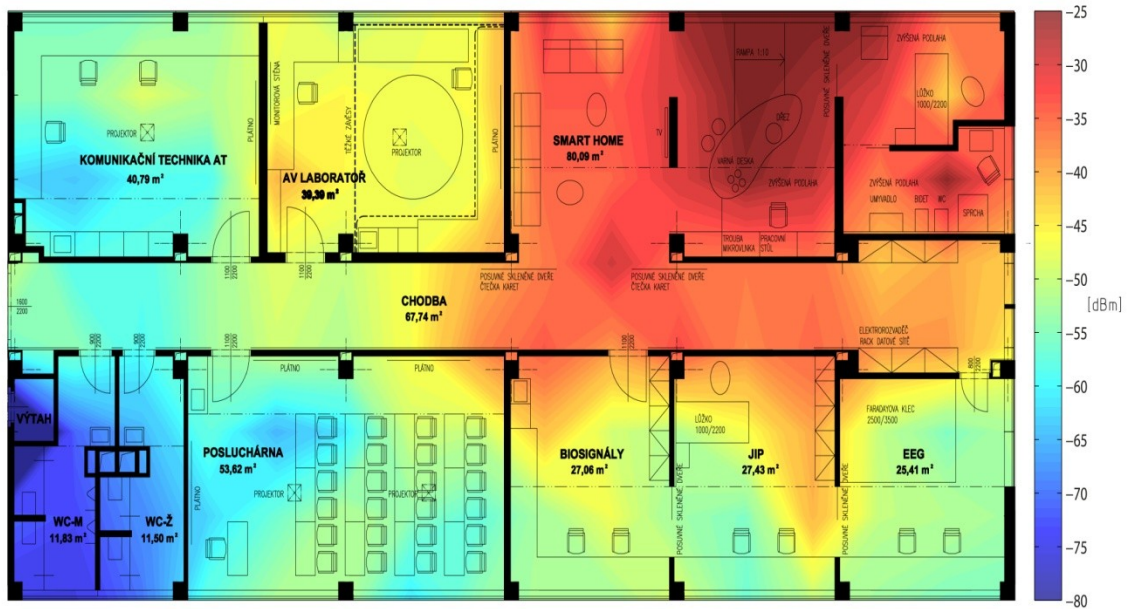




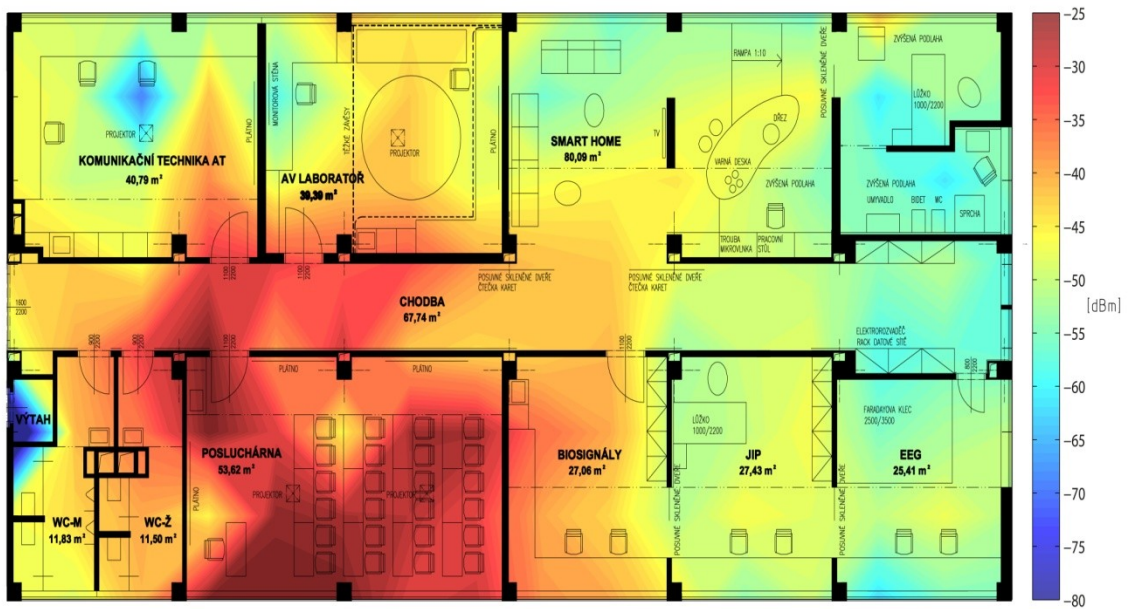
Obr. B. 3: Pokrytí pro kanál 6



Obr. B. 4: Pokrytí pro kanál 9



Obr. B. 4: Pokrytí pro kanál 12

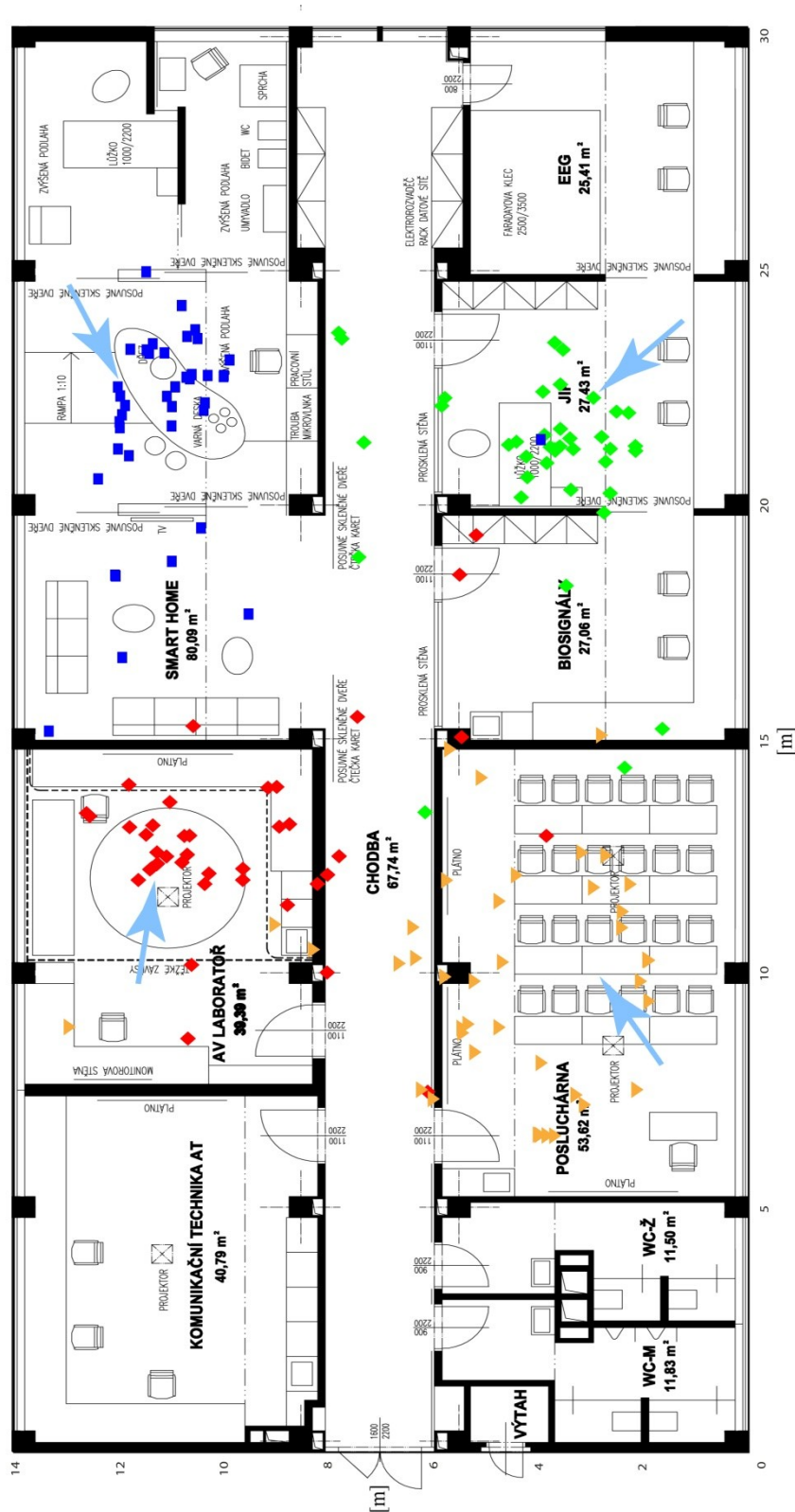


Obr. B. 4: Pokrytí pro kanál 13

# Příloha C

## Vizualizace přesnosti stanovení pozice lokalizačním systémem

- šipky udávají skutečnou pozici při jednotlivých (čtyřech) měření



Obr. C. 1: Výsledný charakter přesnosti stanovení pozice mobilního zařízení

# Příloha D

## Obsah přiloženého CD

- 1) Readme.txt - informace o obsahu CD
- 2) Elektronická verze této práce ve formátu PDF
- 3) Zdrojové kódy aplikací
- 4) Mapové podklady

ad 3) Adresář obsahuje podadresáře zdrojových kódů jednotlivých aplikací. Mezi ně patří:

*RTLS Demo,*  
*RTLS Admin,*  
*RTLS Localization Client a*  
*RTLS Localization Service.*

Toto je zároveň kompletní výčet aplikací, jež tvoří navržený lokalizační systém. Cílem této práce byla tvorba aplikace *RTLS Demo*, která byla doplněna o komunikaci s FTP serverem a byla pojmenována jako *RTLS Admin*. Obě tyto aplikace jsou určeny pro správce systému. První může sloužit pro měřicí a optimalizační účely, protože je zcela nezávislá na konektivitě s FTP serverem. Druhá zmiňovaná aplikace slouží pro tvorbu a distribuci databáze, kterou využívají zbylé dvě uživatelské aplikace. Navíc také obsahuje funkci vzdáleného dohledu.

ad 4) Adresář obsahuje podadresáře s mapovými podklady prostředí, vizualizacemi pro jednotlivé metodiky měření pokrytí a případných zkoušek lokalizačního systému.



# Příloha E

## Návod pro obsluhu mobilní aplikace *RTLS Demo (RTLS Admin)*

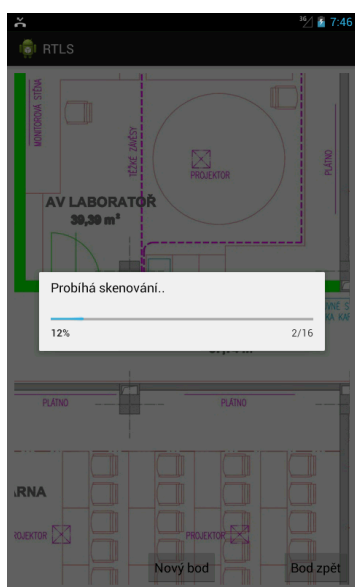
Návod je popisován pro *RTLS Demo*, *RTLS Admin* se liší jen částečně. Pro správný chod aplikace je nutný Android OS verze 3.0 (Honeycomb) a vyšší. Úvodní menu mobilní aplikace správce na Obr. E. 1, spouští čtyři základní funkce, **Dohled**, **Proměření prostoru**, **Spustit lokalizaci** a **Nastavení**. Pod tlačítkem **Dohled** je zobrazení lokalizace uživatelů na dálku (v *RTLS Demo* neaktivní). Do aplikace je potřebné přidat mapu prostoru, ve kterém chceme provést měření databáze. Nejjednodušší způsob je vložení mapy přímo do složky vytvořené aplikací. Poté co byla aplikace nainstalována a poprvé spuštěna, se na SD kartě zařízení objeví adresář */rtls* a v něm složka */maps*, do které stačí mapu zkopírovat. Je však nutné dodržet maximální velikost mapy dva miliony obr. bodů (2Mpx).



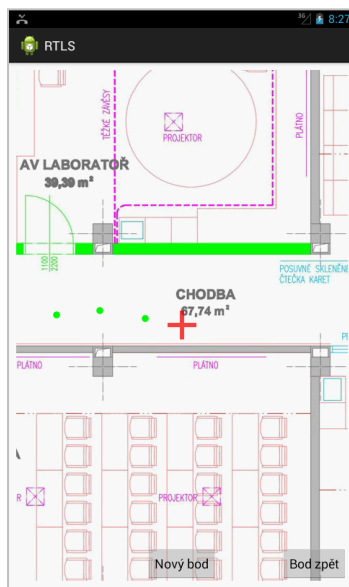
Obr. E. 1: Úvodní menu

### 1. Proměření prostoru

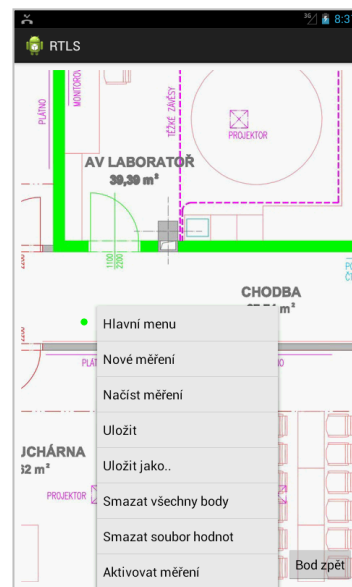
V prvním kroku, tvorby databáze prostoru, tedy vybereme položku **Proměření prostoru**. Poté se zobrazí prázdná plocha aplikace, která bude mít deaktivovaná tlačítka. Hardwarovým či softwarovým tlačítkem (umístěným vpravo na hlavní liště aplikace) vyvoláme hlavní menu (Obr. E. 4). Z něj vybereme položku **Nové měření**. Tím začne průvodce vytvoření nového měření (projektu). Nejprve se zobrazí dialog pro vybrání potřebné mapy, poté se zobrazí dialog pro pojmenování a uložení nového měření. Stiskem tlačítka **Potvrdit** měření uložíme. Na pozadí aplikace se zobrazí vybraná mapa a aktivují se tlačítka **Nový bod** a **Bod zpět**. Nyní už můžeme vytvářet databázi hodnot. Na středu obrazovky je zobrazen poziční křížek, který je statický. Posunem, oddalováním či přibližováním mapy, dosáhneme toho, že naši pozici můžeme pod daný křížek nastavit. Nyní stiskem tlačítka



Obr. E. 2: Skenování



Obr. E. 3: Tvorba databáze



Obr. E. 4: Menu měření

**Nový bod** zadáme na mapu novou značku (Obr. E. 3). Tím byla vytvořena první hodnota v databázi. Každá hodnota je pečlivě skenována, proto se zobrazí dialog o průběhu skenování (Obr. E. 2). Tímto způsobem můžeme projít celý prostor a tím vytvořit plnohodnotnou databázi dat. Doporučení pro její tvorbu je takové, abychom si v daném prostoru vytvořili síť bodů, kde každý bude od sebe vzdálen v obou osách například dva metry. *Důležité je podotknout, že čím bude hustější síť bodů (více naměřených bodů), tím bude výsledná lokalizace přesnější.* Tlačítko **Bod zpět** nám dává možnost, abychom se o jeden bod vrátili, když například zadáme špatnou pozici.

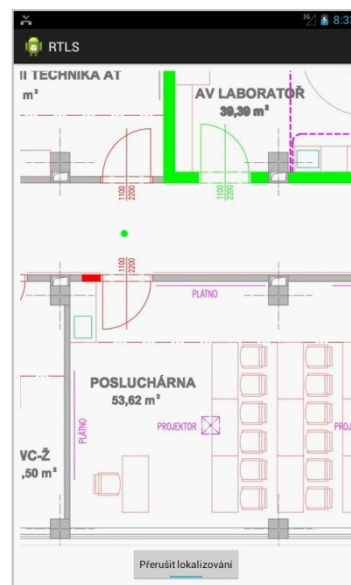
Menu měření dále obsahuje položky pro ukládání **Uložit**, **Uložit jako** a načítání uložených měření **Načíst měření**. Dále obsahuje položku **Smazat všechny body**, jež maže všechny naměřené hodnoty pro konkrétní měření, a položku **Smazat soubor měření**, jež smaže kompletně celý soubor měření. Nejdůležitější je však položka poslední, **Aktivovat měření**, která aktuálně otevřený projekt nastaví jako aktivní. Tím je umožněna lokalizace, tedy k dané mapě je specifikována jedna aktuální databáze. Po vytvoření a uložení měření (databáze) můžeme začít s lokalizací.

## 2. Spustit lokalizaci

Tlačítko **Spustit lokalizování**, po spuštění lokalizace z hlavního menu, bude nejprve neaktivní. Je nutno zvolit položku v menu **Načíst mapu**. Po načtení mapy se tlačítko na displeji aktivuje a poté můžeme tlačítkem spustit lokalizování. Vaše pozice je poté zobrazena bodem na mapě (Obr. E. 5). Tato pozice se každých šest vteřin obnovuje. Lokalizaci přerušíme opětovným poklepem na tlačítko.

## 3. Nastavení

Položka **Nastavení** z hlavního menu, zobrazí hlavní menu nastavení. Zde můžeme najít položky **Konfigurace**, **Jazyk**, **Nastavení sítě** a **O aplikaci**. Položka konfigurace slouží pro nastavení měřítka mapy. Tato funkce však neslouží pro momentální účely aplikace, ale pro její další vývoj. Další položkou menu nastavení je **Jazyk**. Zde je na výběr český a anglický jazyk, kde druhý zmiňovaný je standardně nastaven po prvním spuštění aplikace. Poslední položkou menu nastavení je **Nastavení sítě**. Zde se zadávají konfigurační parametry pro komunikaci s bezdrátovou sítí. *Pro RTLS Demo se zde nastavuje SSID sítě, což je základní podmínkou pro chod aplikace.* Pro *RTLS Admin* se zde nastavují parametry pro komunikaci se serverem, a to uživatelské jméno, heslo, IP adresa, port, který je primárně nastaven na 21, a SSID sítě, ze které je FTP server přístupný. *Pro správný chod RTLS Admin je nutné tyto parametry nastavit.*



Obr. E. 5: Lokalizace