**מדידת מרחק בעזרת WIFI**

לפי (להוסיף קישור) ומקורות נוספים, ניתן לקבל RSSI ובאמצעותו למדוד את המרחק (d) בהתאם לנוסחה הבאה:

כאשר הוא ה-RSSI המתקבל בנקודת "ייחוס" הנמצאת במרחק ממקור ה-WIFI.

עבור *, נסמן*

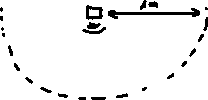
*ונקבל*

ניסינו לבצע מספר ניסוייים על מנת למצוא את הפרמטרים A, ו-n ולחשב את המרחק.

**ניסוי ראשון:**

"מקור" ה-WIFI הוחזק במקום אחד, ובקר נוסף שמודד RSSI, "הסתובב" סביבו בכחצי מעגל ברדיוס מסוים, על מנת לבדוק מה ה-RSSI המתקבל במרחק זה מכמה כיוונים. המדידות מתבצעות במרווחי זמן קבועים של 20 מילישניה.

שרטוט להמחשה:



חזרנו על הניסוי במרחקים: 0.5 מטר, 1 מטר, 2 מטרים, 3 מטרים, 4 מטרים ו-5 מטרים.

התוצאות שהתקבלו (התעלמנו מחלק מהדגימות האחרונות בחלק מהמרחקים, על מנת לשמור על אחידות במספר הדגימות):

התקבלו תוצאות לא מאוד יציבות ועם הרבה קפיצות, למרות שהמרחק היה פחות או יותר קבוע עבור כל עקומה. ניסיתי לחפש דרכים להפוך את המדידה ליותר "יציבה".

כאן: <https://journals.sagepub.com/doi/pdf/10.1155/2015/195297> מופיעות מספר אפשרויות לשיפור התוצאות. נכון לעכשיו, ולפי הצעות שמצאתי באינטרנט, ניסיתי להשתמש בקוד שמופיע כאן: <https://github.com/bachagas/Kalman/blob/master/README.md> (מימוש של Kalman filter).

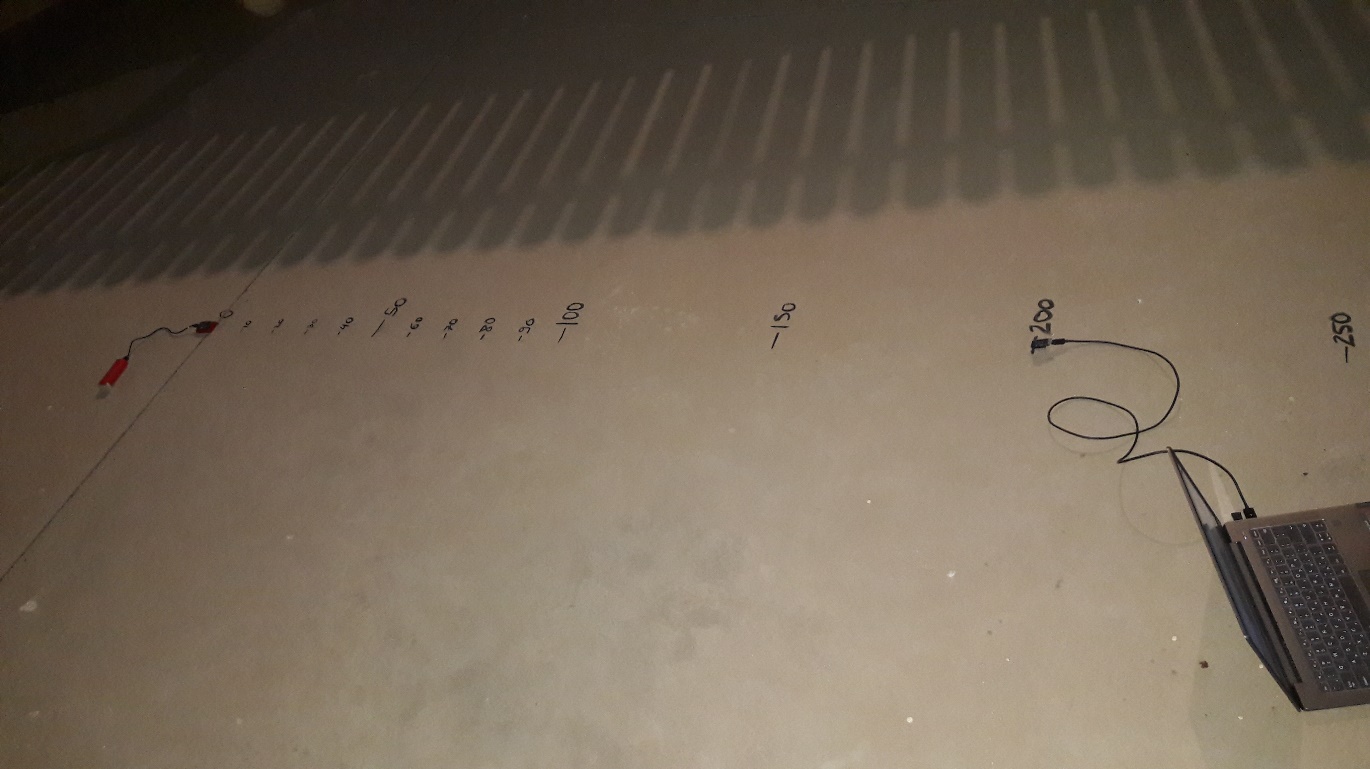
כך נראית ה­­השפעה שלו, למשל עבור מרחק של מטר אחד:

ועבור כל המרחקים:

העקומות יותר "חלקות" בדרך הזו.

ה-RSSI המתקבל אמור להיות מונוטוני כפונקציה של המרחק, וניתן לראות שזה לא קרה הפעם (העקומה עבור מרחק חצי מטר נמצאת בין 1 ל-2, למשל).

כלומר עדיין, ניתן לראות שיש הרבה רעש והמדידה לא מדויקת כל כך, ולכן ביצענו ניסוי נוסף:

"מקור" ה-WIFI מונח במקומו, ואנו מודדים את ה-RSSI ממרחקים שונים (באותו הכיוון), כך שבכל שלב הבקר המודד נמצא בנקודה קבועה ולא זז. תמונה להמחשה: 

חישבנו את הפרמטר n בתור ה-n הממוצע שהתקבל מהמדידות כאשר המרחק ידוע, בהתאם לנוסחה, ולאחר מכן השתמשנו ב-n שהתקבל כדי לחשב את המרחקים.

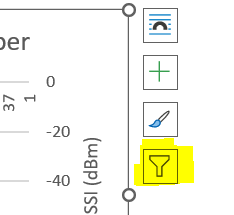
התוצאות שהתקבלו:

התוצאות שהתקבלו יחד עם הfilter-, והתוצאות שהתקבלו מה-RSSI הרגיל מאוד קרובות במקרה זה, שבו האות נקלט בלי תזוזה של הבקר (המדידות היו די יציבות עבור מרחק קבוע ללא תזוזה). השגיאה היחסית ( ) הממוצעת שקיבלנו עבור RSSI רגיל היא 32.3%, לעומת 32.2% עבור הfiltered RSSI.

כמו כן, עבור מרחקים גדולים יותר, החל מ-2 מטרים, השגיאה שהתקבלה היא 43.1% ו-42.9% בהתאמה. כלומר יש שיפור קטן (מאוד), אך עדיין שגיאה גדולה.

נרצה לבחון את ההשפעה של ה-filter שניסינו, וגם אפשרויות נוספות, כאשר הבקר לא לגמרי במנוחה, למשל מוחזק ביד במקום, כפי שיכול להיות במשחק עצמו.

ניסוי נוסף: התקרבות בקצב פחות או יותר קבוע למקור ה-WIFI, כאשר בכל מרחק מבין כמה מרחקים שנקבעו מראש, לוחצים על כפתור על מנת לדעת לשייך את המדידות שמתקבלות למרחק הנוכחי.

בבדיקת הנתונים השתמשנו בכמה אופציות לטיפול ברעשים, מתוכן ממוצע, חציון, ממוצע נע, exponential smoothing, ו-Kalman filter, חלקם עם כמה פרמטרים שונים. ניתן לראות באקסל (להוסיף), וכאן ניתן להציג רק חלק מהעקומות בעזרת הכפתור: . שמופיע כשלוחצים על הגרף.

השתמשנו במדידות אלה על מנת לחשב את המרחק, יחד עם פרמטרים n שנקבע במדידה נפרדת עי ממוצע של שלושה Nים 20 80 100 ממרחק של d0=50, ואילו P(d0)=B חושב כממוצע מניסוי זה (נבחרו בעזרת ניסוי וטעיה, כי זה מה שהקטין את השגיאה -ערכים אחרים יכולים להוביל לשגיאות של יותר מ-50%).

התוצאות שהתקבלו – (כמעט) מכל אופציה לטיפול ברעש, בחרנו את מה שצמצם את השגיאה, כך שנותרנו עם האפשרויות (שצריך להמשיך לבדוק כדי לבחור):

1. ממוצע של כל 20 דגימות נתן שגיאה קטנה יחסית לאחרים, אבל היו רק מעט מדידות שנבדקו.
2. exponential smoothing () עבור נתן תוצאות טובות יחסית לאחרים2.
3. גם ממוצע נע של 20 דגימות *נראה עם שגיאה יחסית (לאחרים) קטנה.*
4. *בדיקה של Kalman filter.*

*חישובי המרחקים בעזרת האפשרויות שנבחרו:*

ממדידות שביצענו על מנת לחשב את הפרמטרים n ו-B, גילינו שקשה למצוא אותם, הם משתנים אפילו באותה הסביבה, בין מדידות שונות, ולכן אולי כדאי להוסיף מדידה שלהם בתחילת המשחק בתור "כיול" הכולל 3 מדידות ממרחקים שונים ולא גדולים כדי שיהיה ניתן לבצע.

**ניסוי בשטח:**

1. ממוצע של 20 דגימות בכל פעם:

גילינו שבחירת n ("כיול") טובה יותר עבור מרחקים של 50,80,100 ס"מ.

עדיין – תוצאות לא נכונות, סיבובים קטנים של הבקר במקום יכולים לשנות מאוד את המרחק, מרגיש קצת איטי (עובר זמן בין הדפסות של המרחק) והתוצאות בין מדידה למדידה יכולות לקפוץ כי מתייחסים לממוצע של מדידות נפרדות בכל הצגה של המרחק.

ה-n שנתן תוצאות הכי טובות: 2.27, עבור מרחק: d0=0.8. וגם כך – מרחק לא מדויק.

1. שימוש בKalman filter- - עדיין יש קפיצות לפעמים, ובחלק מהמקרים המרחק לא מדויק כל כך.

יתכן שהבעיה של המרחק קשורה לנוסחה שבחרנו ולקביעת הפרמטרים שלה. נראה שהנוסחה יכולה להיות נכונה בתחום קטן מאוד, ולאחר מכן מתחילה לגדול בקצב ממש מהיר, או לחלופין להיות קטנה מאוד כל הזמן בלי להשתנות הרבה ובלי לייצג נכון את המרחק כמעט בכלל, תלוי בפרמטרים שבוחרים, גם אם מודדים אותם ממש בתחילת המשחק ("כיול").

מסקנות:

חיפוש באינטרנט הוביל למציאת נוסחה נוספת <https://stackoverflow.com/questions/11217674/how-to-calculate-distance-from-wifi-router-using-signal-strength> המשמשת למדידת המרחק בשטח פתוח. בנוסף, מה שחשוב למציאת המטמון במשחק זה המונוטוניות של המרחק המוצג ולא דווקא המרחק עצמו (כדי לדעת אם מתקרבים או מתרחקים). בגרף האחרון רואים שKalman ו-exponential smoothing נראים יחסית מונוטוניים ולכן ננסה להמשיך איתם בבדיקות הבאות.

ניסוי עם הנוסחה החדשה: עם הנוסחה הזאת לפחות עד מרחק של כ-10 מטרים, אם פונים אל המטמון בקו יחסית ישר, המרחק נראה די קרוב למציאות (כרגע). עם סיבובים ותזוזות קטנות מהקו הישר זה, המרחק יכול להשתנות מאוד, אך ננסה עם סינון:

1. ניסיון: הנוסחה ה"חדשה", עם Kalman. שינויים די מהירים, אך יחסית סבבה. ניסיון להציג בכל פעם ממוצע של מספר **קטן** של מדידות (כלומר להוסיף ממוצע) עזר. יתכן שזהו האופציה שתיבחר.
2. exponential smoothing עם אלפא 0.1 והנוסחה החדשה – יש קפיצות במרחקים, מתעדכן פעם בכמה זמן, קצת יותר איטי, ועדיין שינויים די מהירים כשהמרחק בפועל משתנה לא ככ מהר.
3. ממוצע נע 20 דגימות – שינויים מהירים.

וקצת ניסיונות (שלא נבחרו) לשילובים בניהם...

בינתיים נבחרה אופציה 3.

אחרי בדיקות נוספות בזמנים ומקומות אחרים, המרחק (גם עם הנוסחה החדשה) כבר לא נראה כ"כ נכון, אבל יחסית מונוטוני כשמתקרבים בקו ישר למטמון, וכשלא מכוונים את המחפש לעבר קיר או משהו דומה.

ניסון לשיפור: הוספת parabolic reflector למחפש. זה אכן גרם למרחק להיות קטן יותר כשמצביעים לכוון של המטמון, מאשר לכיוונים אחרים, ובכך גורם ל"מרחק" הכתוב להיות אינדיקציה גם עבור הכיוון.