



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

## NEJISTOTA MĚŘENÍ ROZMĚRŮ VOZIDEL POMOCÍ 2D LIDARU

UNCERTAINTY OF VEHICLE DIMENSION MEASUREMENT USING 2D LIDAR

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

Bc. David Želinský

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Soňa Šedivá, Ph.D.

BRNO 2022

# Diplomová práce

magisterský navazující studijní program **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

**Student:** Bc. David Želinský

**ID:** 203429

**Ročník:** 2

**Akademický rok:** 2021/22

**NÁZEV TÉMATU:**

## Nejistota měření rozměrů vozidel pomocí 2D lidarů

### POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je popsat a kvantifikovat zdroje nejistot při měření rozměrů vozidel pomocí 2D lidarů. Práce je vypisována ve spolupráci s firmou CROSS Zlín, a.s.

1. Nastudujte problematiku měření profilů pomocí liniového lidarů. Popište princip liniového lidarů a jeho použití.
2. Na dané konfiguraci měřicího stanoviště, která je zadána firmou CROSS Zlín, a.s., definujte množinu parametrů senzoru, které mají potenciál významně ovlivňovat přesnost měření výšky, šířky a délky vozidel.
3. Vliv jednotlivých parametrů senzoru na základě teoretických předpokladů ideálně matematicky kvantifikujte, proveďte odhad nejistot měření. Když to nebude kvůli složitosti možné, tak jej určete alespoň kvalitativně.
4. Na poskytnutých datech ověřte předpokládané vlivy.
5. Navrhněte a realizujte algoritmus pro stanovení rozměrů vozidel z dat získaných se senzorů. Součástí programu bude vyhodnocení nejistoty měření.
6. Dosažené výsledky práce zhodnotte.

### DOPORUČENÁ LITERATURA:

[1] Evaluation of measurement data — Guide to the expression of uncertainty in measurement [online]. GUM 1995 with minor corrections, 2008. Dostupné z: [https://www.bipm.org/utls/common/documents/jcgm/JCGM\\_100\\_2008\\_E.pdf](https://www.bipm.org/utls/common/documents/jcgm/JCGM_100_2008_E.pdf)

**Termín zadání:** 7.2.2022

**Termín odevzdání:** 18.5.2022

**Vedoucí práce:** Ing. Soňa Šedivá, Ph.D.

**doc. Ing. Petr Fiedler, Ph.D.**  
předseda rady studijního programu

### UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **Abstrakt**

Tato diplomová práce se zabývá vysvětlením pojmů chyba měření a nejistota měření. Také představuje pojem LiDAR, popisuje jeho konstrukci, jak pracuje, a k čemu se dá použít. Informace z teoretické části této práce jsou pak použity pro analýzu nejistot měření rozměrů vozidel pomocí 2D LiDARu. Jednotlivé nejistoty měření jsou vysvětleny a na vzorovém příkladu vypočítány. Dále je vytvořen program v LabVIEW, který z naměřených dat rozměry vozidla a nejistoty těchto rozměrů počítá. Na závěr jsou srovnány výsledky programu s reálnými rozměry. Práce je vypracována pro firmu Cross Zlín.

## **Klíčová slova**

LiDAR, metoda měření TOF, chyba měření, nejistota měření, databáze, LabVIEW, stavový automat

## **Abstract**

This diploma thesis deals with the explanation of terms such as measurement error and uncertainty of measurement. It also introduces the term LiDAR, describes its construction, how it works and what it can be used for. The information from the theoretical part of this thesis is then used to analyze the uncertainties of measuring the dimensions of vehicles using 2D LiDAR. The individual measurement uncertainties are explained and calculated on an example. Furthermore, a LabVIEW program is created, which calculates the dimensions of the vehicle and the uncertainties of these dimensions from measured data. Finally, the results of the program are compared with real dimensions. The thesis is elaborated for the company Cross Zlín.

## **Keywords**

LiDAR, TOF measurement method, measurement error, uncertainty of measurement, database, LabVIEW, state machine

## **Bibliografická citace**

Citace tištěné práce:

ŽELINSKÝ, David. Nejistota měření rozměrů vozidel pomocí 2D lidarů. Brno, 2022. Dostupné také z: <https://www.vutbr.cz/studenti/zav-prace/detail/142148>. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce Soňa Šedivá.

Citace elektronického zdroje:

ŽELINSKÝ, David. Nejistota měření rozměrů vozidel pomocí 2D lidarů [online]. Brno, 2022 [cit. 2022-05-13]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/142148>. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce Soňa Šedivá.



# Prohlášení autora o původnosti díla

**Jméno a příjmení studenta:** *David Želinský*

**VUT ID studenta:** *203429*

**Typ práce:** *Diplomová práce*

**Akademický rok:** *2021/22*

**Téma závěrečné práce:** *Nejistota měření rozměrů vozidel pomocí 2D lidarů*

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: 17. května 2022

-----  
podpis autora

## **Poděkování**

Děkuji vedoucí diplomové práce Ing. Soni Šedivé, Ph.D. a konzultantovi diplomové práce Ing. Janovi Klečkovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne: 17. května 2022

-----  
podpis autora

# Obsah

<b>SEZNAM OBRÁZKŮ .....</b>	<b>9</b>
<b>SEZNAM TABULEK.....</b>	<b>10</b>
<b>1. ÚVOD.....</b>	<b>11</b>
<b>2. LIDARY .....</b>	<b>12</b>
2.1 KONSTRUKCE 2D LIDARŮ .....	12
2.2 PRINCIP MĚŘENÍ 2D LIDARŮ.....	12
<b>3. NEJISTOTY MĚŘENÍ.....</b>	<b>17</b>
3.1 CHYBY MĚŘENÍ .....	17
3.1.1 Rozdělení chyb .....	18
3.1.2 Chyby měřicích přístrojů .....	18
3.1.3 Chyby metody.....	19
3.1.4 Chyby nepřímých měření .....	19
3.2 NEJISTOTY MĚŘENÍ.....	20
3.3 NEJISTOTA TYPU A.....	20
3.4 NEJISTOTA TYPU B .....	21
3.5 KOMBINOVANÁ NEJISTOTA .....	23
3.6 ROZŠÍŘENÁ NEJISTOTA .....	23
3.7 NEJISTOTA NEPŘÍMÉHO MĚŘENÍ .....	23
3.7.1 Nejistota nepřímého měření nekorelovaných veličin .....	24
3.7.2 Nejistota nepřímého měření navzájem korelovaných veličin .....	24
3.8 ZAOKROUHLOVÁNÍ NEJISTOT MĚŘENÍ .....	25
3.9 ZÁPIS NEJISTOT MĚŘENÍ .....	25
3.9.1 Zápis s kombinovanou nejistotou $u_c(x)$ .....	25
3.9.2 Zápis s rozšířenou nejistotou $U(x)$ .....	26
3.9.3 Bilanční tabulka.....	26
<b>4. POUŽITÍ 2D LIDARU V REÁLNÉ SITUACI .....</b>	<b>28</b>
4.1 SROVNÁNÍ KONKURENČNÍCH LIDARŮ.....	28
4.2 MĚŘICÍ STANOVISŤE.....	29
4.3 METODY MĚŘENÍ JEDNOTLIVÝCH ROZMĚRŮ.....	30
4.3.1 Měření délky vozidla .....	31
4.3.2 Měření šířky vozidla.....	31
4.3.3 Měření výšky vozidla.....	32
<b>5. ANALÝZA NEJISTOT MĚŘENÍ 2D LIDAREM .....</b>	<b>33</b>
5.1 DATABÁZE FIRMY CROSS .....	33
5.1.1 Schéma Raw .....	33
5.1.2 Schéma Data .....	34
5.1.3 Čtení dat z databáze.....	35
5.2 UVAŽOVANÉ NEJISTOTY .....	37
5.2.1 Nejistoty ovlivňující měření délky .....	37
5.2.2 Nejistoty ovlivňující měření šířky.....	37
5.2.3 Nejistoty ovlivňující měření výšky.....	38

5.3	VÝPOČET NEJISTOT .....	38
5.3.1	<i>Pracovní podmínky LiDARu</i> .....	38
5.3.2	<i>Nejistota dána systematickou chybou LiDARu</i> .....	39
5.3.3	<i>Nejistota dána statistickou chybou LiDARu</i> .....	39
5.3.4	<i>Tvar vozidla</i> .....	39
5.3.5	<i>Délka vozidla</i> .....	41
5.3.6	<i>Měření rychlosti</i> .....	41
5.3.7	<i>Zaokrouhlování rychlosti</i> .....	42
5.3.8	<i>Šířka vozidla</i> .....	43
5.3.9	<i>Výška vozidla</i> .....	44
5.4	VÝSLEDEK MĚŘENÍ.....	44
<b>6.</b>	<b>VYTVOŘENÝ ALGORITMUS .....</b>	<b>47</b>
6.1	STAVOVÝ DIAGRAM .....	47
6.2	STROM PROJEKTU .....	48
6.3	BLOKOVÝ DIAGRAM V LABVIEW.....	49
6.3.1	<i>Stav inicializace (Initialize)</i> .....	51
6.3.2	<i>Stav zobrazování dat (Display Data)</i> .....	51
6.3.3	<i>Stav výpočtu rozměrů (Dimensions)</i> .....	53
6.3.4	<i>Stav výpočtu nejistot (Uncertainties)</i> .....	53
6.3.5	<i>Klidový stav (Wait for Event)</i> .....	53
6.3.6	<i>Stav ukončení (Exit)</i> .....	54
6.3.7	<i>Podprogramy (SUBVI) stavového automatu</i> .....	54
6.4	OVLÁDACÍ PANEL V LABVIEW .....	64
6.5	OVLÁDÁNÍ PROGRAMU ZA BĚHU .....	65
<b>7.</b>	<b>ZHODNOCENÍ VÝSLEDKŮ .....</b>	<b>66</b>
<b>8.</b>	<b>ZÁVĚR.....</b>	<b>70</b>
<b>9.</b>	<b>LITERATURA .....</b>	<b>71</b>

## SEZNAM OBRÁZKŮ

Obrázek 2.1: Schéma principu měření TOF [13].....	13
Obrázek 2.2: Měření fázového posuvu u principu nepřímého TOF [18].....	14
Obrázek 2.3: Schéma srovnání principů měření TOF: přímý TOF nalevo, nepřímý TOF napravo [18] ....	15
Obrázek 2.4: Ukázka měření LiDAREm na stanovišti firmy Cross.....	15
Obrázek 2.5: Ukázka měření LiDAREm [35] .....	16
Obrázek 3.1: Zobrazení jednotlivých rozdělení [24] .....	22
Obrázek 4.1: Názorné schéma umístění instrumentace na stanovišti .....	30
Obrázek 4.2: Záběr na měřicí stanoviště firmy Cross u Tlumačova na D55 [36].....	30
Obrázek 4.3: Ukázka měření délky vozidla [37] .....	31
Obrázek 4.4: Ukázka měření šířky vozidla [31] .....	32
Obrázek 5.1: Základní struktura dat JSON .....	35
Obrázek 5.2: Schéma rozměrů vozidla Škoda Octavia iV (sedan) [30].....	36
Obrázek 5.3: Odchylka způsobená tvarem osobního automobilu [31] .....	40
Obrázek 5.4: Odchylka způsobená tvarem krytých návěsů a přívěsů .....	40
Obrázek 5.5: Schéma pro výpočet odchylky měření šířky vozidla .....	43
Obrázek 6.1: Stavový diagram programu .....	48
Obrázek 6.2: Strom projektu LabVIEW .....	49
Obrázek 6.3: Blokový diagram SUBVI <i>Nejistota zaokrouhlení</i> .....	55
Obrázek 6.4: Blokový diagram SUBVI <i>Nejistota měření šířky</i> .....	56
Obrázek 6.5: Blokový diagram SUBVI <i>Měření délky</i> .....	57
Obrázek 6.6: Blokový diagram SUBVI <i>Hex na uint16</i> .....	57
Obrázek 6.7: Blokový diagram SUBVI <i>Filtr</i> .....	60
Obrázek 6.8: Blokový diagram SUBVI <i>Filtrování s JSON</i> .....	61
Obrázek 6.9: Blokový diagram SUBVI <i>Vyrovnání roviny</i> .....	62
Obrázek 6.10: Náhled na ovládací panel vytvořeného programu .....	64
Obrázek 6.11: 3D graf naskenovaného vozidla s ID 1582094.....	65

## SEZNAM TABULEK

Tabulka 3.1: Hodnoty $k_s$ na základě počtu měření $n$ [21] .....	21
Tabulka 3.2: hodnoty $\varkappa$ na základě typu rozdělení .....	22
Tabulka 3.3: Hodnoty $k_r$ na základě pravděpodobnosti výskytu hodnot [21] .....	23
Tabulka 3.4: Obecná podoba bilanční tabulky [24] .....	27
Tabulka 4.1: Srovnání parametrů vybraných LiDARů [25], [26], [27], [28], [29] .....	29
Tabulka 5.1: Vyfiltrovaná data z databáze pro výpočet nejistot .....	36
Tabulka 5.2: Údaje o prostředí, ve kterých může LiDAR Sick LMS511 pracovat [25] .....	38
Tabulka 7.1: Výsledné hodnoty pro délku vozidla Škoda Octavia IV sedan .....	66
Tabulka 7.2: Výsledné hodnoty pro šířku vozidla Škoda Octavia IV sedan .....	66
Tabulka 7.3: Výsledné hodnoty pro výšku vozidla Škoda Octavia IV sedan .....	67
Tabulka 7.4: Výsledné hodnoty pro délku vozidla Škoda Octavia III combi .....	68
Tabulka 7.5: Výsledné hodnoty pro šířku vozidla Škoda Octavia III combi .....	68
Tabulka 7.6: Výsledné hodnoty pro výšku vozidla Škoda Octavia III combi .....	68

# 1. ÚVOD

Před přibližně 136 lety požádal Karl Benz o patent prvního vozidla se spalovacím motorem. Ten samý rok patent dostal. A i když to byl tenkrát velký technologický skok, nikdo nejspíš netušil, jak moc se automobilismus rozvine. Auta se stala dostupnými, a jsou nabízena ve všech možných tvarech, s různými oblastmi použití i různými pohony. Nicméně v době, kdy se čím dál víc řeší ekologický dopad vozidel na životní prostředí, jsou určitá omezení pohybu velkého množství vozidel nevyhnutelná. Než se ale vyřeší příčiny tohoto problému, je nutné udržet automobilismus alespoň v rozumných mezích. Tímto směrem se vydala například Zlínská firma Cross, která se s pomocí svých měřicích stanovišť snaží omezit alespoň pohyb přetížených vozidel, která nejen že jsou pro životní prostředí škodlivější, ale také vážou ničí komunikace. Firma analyzuje přetížení vozidel pomocí jejich vážení a měření jejich rozměrů za běžného provozu na komunikacích s vysokým vytížením.

Jak napovídá zadání, cílem této práce je analyzovat nejistoty, které výsledky měření rozměrů vozidel LiDARem ovlivňují, tyto nejistoty vyjádřit a dále napsat algoritmus, který tyto nejistoty bude zároveň s rozměry vozidla počítat.

V práci bude popsáno, co to jsou LiDARy, jaké jsou typy LiDARů, kde a jak se používají a na jakých principech pracují. Zobrazen bude i příklad měření LiDARem z reálného měřicího stanoviště firmy Cross, pro kterou je práce vypracována.

Dále v práci bude napsáno, co to jsou chyby měření a nejistoty měření. Popisy budou doplněny vztahy pro výpočet chyb i nejistot měření a jejich vhodné zápisy u výsledku měření včetně pravidel pro správný zápis.

V další kapitole budou srovnány 2D LiDARy, které buď jsou nebo by mohly být použity na měřicích stanovištích pro měření rozměrů vozidel firmou Cross. Popsána bude i konstrukce typického měřicího stanoviště firmy Cross a vysvětleny budou i metody měření jednotlivých rozměrů vozidel s názornými příklady.

V páté kapitole práce bude vysvětleno, jakým způsobem byla získána data z měřicího stanoviště, a na základě těchto dat budou vypočítány nejistoty měření jak společné pro všechny rozměry, tak pro jednotlivé rozměry zvlášť. Kapitola bude doplněna vhodným příkladem z reálného měření.

Další kapitola bude obsahovat detailní popis vytvořeného algoritmu pro výpočet rozměrů a nejistot měření rozměrů vybraného vozidla. Ten bude vytvořen jako stavový automat, bude vysvětlen jeho stavový diagram, dále kód každého stavu a pak jednotlivých vytvořených podprogramů a funkcí. Na konci kapitoly bude popsáno ovládání programu.

Závěrečná kapitola bude věnována zhodnocení výsledků dosažených vytvořeným algoritmem a tyto výsledky budou srovnány s výsledky firmy Cross.

Jak již bylo uvedeno, práce je vypracována pro Zlínskou firmu Cross. Firma se zabývá chytrými dopravními řešeními od měření parametrů vozidel jako jsou rozměry a váha, detekce přestupků, přes parkovací systémy až po silniční meteorologii.

## 2. LIDARY

LiDAR (*Light Detection And Ranging*) je zařízení určené pro měření vzdáleností množiny bodů, které tvoří jeho okolí. Využívá k tomu laserové paprsky, které se po odrazu vrací zpět do zařízení [1].

Historie LiDARů sahá až do třicátých let minulého století, kdy byly provedeny první pokusy měření vzdálenosti pomocí světelného paprsku. To bylo provedeno pomocí reflektorů na analýzu struktury zemské atmosféry. Pulzy světla z těchto reflektorů byly použity pro měření výšky mraků v roce 1938 [2]. Samotný LiDAR bylo možné vytvořit až po příchodu laseru (1960). Prototyp tohoto zařízení vyvinula společnost Hughes Aircraft Company (stejná společnost, co vyvinula první laser) v roce 1961 [3].

LiDARy našly využití ve velkém množství odvětví, např.: astronomie, autonomní mobilita, zemědělství, analýza terénu nebo mořského dna, měření rozměrů objektů, ... [1] [4] [5] [6]

Mezi větší výrobce LiDARů patří německá firma *Sick*, od které používá zařízení i firma Cross (model LiDARu LMS511 bude použit pro zpracování této diplomové práce). Dále jsou to pak například výrobci Hexagon AB, Luminar, Hokuyo, Velodyne Lidar, VanJee Technology... [7], [8], [9], [10], [11], [12]

### 2.1 Konstrukce 2D LiDARů

Typický LiDAR se skládá z těchto částí:

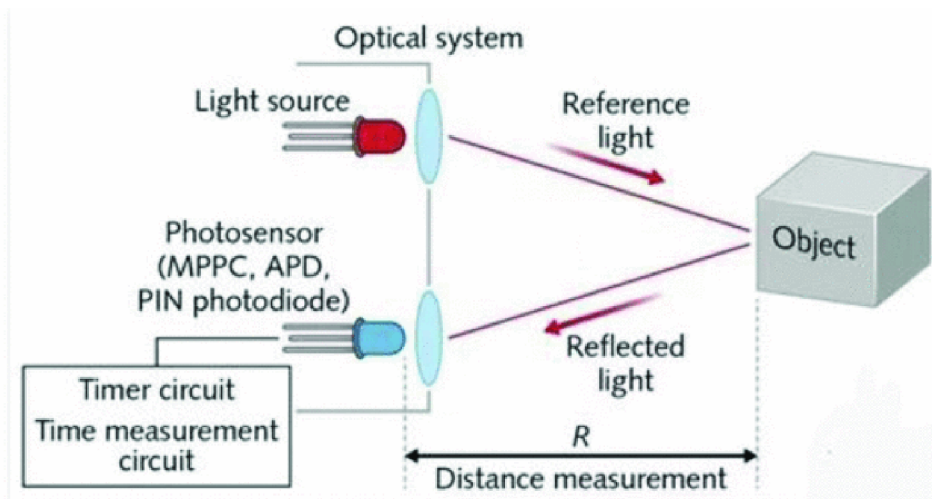
- Zdroj světla (laser s vlnovou délkou v závislosti na použití)
- Senzor dopadajících paprsků (například fotodioda)
- Obvod časovače
- Výpočetní obvody
- Optika LiDARu (zrcátko/broušený skleněný hranol)
- Motor (slouží pro otáčení optiky)

Součástí LiDARu někdy bývá i GPS. Informace z tohoto modulu jsou užitečné například při skenování terénu z letadla [4].

### 2.2 Princip měření 2D LiDARů

Většina LiDARů funguje na základě principu TOF (*Time Of Flight*). Tento princip vysvětluje obrázek 2.1 [13]:





Obrázek 2.1: Schéma principu měření TOF [13]

Zařízení disponuje zdrojem laserového paprsku o vlnové délce typicky 905 nm, případně 1550 nm (bezpečnější pro lidské oko, ale jinak méně efektivní téměř v každém ohledu, jak ukazuje porovnání společnosti Velodyne Lidar [14]). Výjimku tvoří LiDARy snímající mořská dna, které využívají paprsky o vlnové délce 532 nm. Takové paprsky lépe prochází vodním sloupcem [15], [14], [16], [17].

Při měření LiDAR vysílá paprsek směrem k objektu, jehož vzdálenost od zdroje se měří. Zároveň se spustí časovač LiDARu. Paprsek se od objektu odrazí a vrátí se zpět k zařízení. To ho detekuje pomocí světlo-citlivého senzoru. Z časů vyslání a zpětného detekování paprsku se určí doba letu paprsku pomocí následující rovnice:

$$L = \frac{1}{2} v \Delta t \text{ [m]} \quad (2.1)$$

$L$ : měřená vzdálenost [m]

$v$ : rychlost světla ve vzduchu [m/s]

$\Delta t$ : doba mezi vysláním paprsku a detekcí senzorem odraženého paprsku [s]

Čas je v obvodu časovače měřen pomocí oscilátoru a čítače. Po odeslání paprsku začne oscilátor vysílat pulzy s danou frekvencí  $f_o$ . Čítač tyto pulzy počítá až od chvíle, kdy dorazí paprsek zpět. Pomocí frekvence  $f_o$  a počtu pulzů  $n$  lze původní rovnici upravit na:

$$L = \frac{1}{2} c \frac{n}{f_o} \text{ [m]} \quad (2.2)$$

$f_o$ : frekvence pulzů oscilátoru [Hz]

$n$ : počet pulzů oscilátoru mezi vysláním paprsku a detekcí jeho odrazu [-]

Metoda měření TOF může fungovat v některých LiDARech i na jiném principu, označovaném jako nepřímý TOF princip. Takto pracující LiDAR vysílá laserový paprsek s definovanou frekvencí a fází. Odražený paprsek, který se vrátí zpět do senzoru, je pak srovnáván s původním a u frekvenčních průběhů se zjišťuje fázový posun. Frekvence se používá taková, aby nevznikl fázový posun větší než jedna perioda. Maximální vzdálenost při definované frekvenci lze určit z následujícího vzorce [18]:

$$L = \frac{vT}{2} = \frac{v}{2f} [m] \quad (2.3)$$

$L$ : maximální vzdálenost překážky od LiDARu [m]

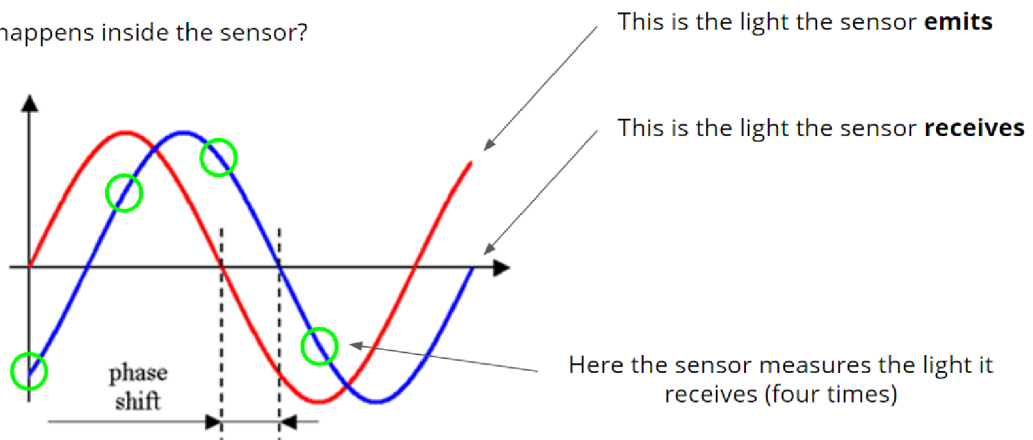
$v$ : rychlost světla ve vzduchu [m/s]

$T$ : perioda signálu paprsku [s]

$f$ : frekvence signálu paprsku [Hz]

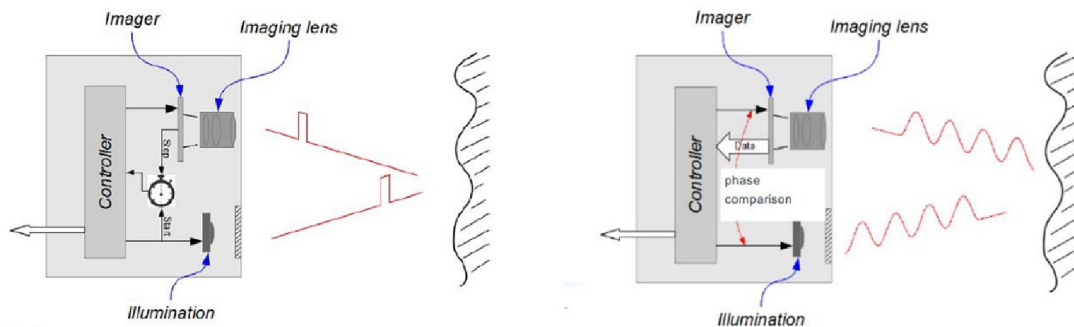
Při použití frekvence například 10 MHz je možné měřit překážku vzdálenou maximálně 15 m. Při umístění překážky do vzdálenosti 7,5 m od senzoru by vznikl fázový posun  $180^\circ$ , při vzdálenosti 15 m pak  $360^\circ$  a signály by se tak úplně překryly [18]. LiDAR srovnává průběhy ve 4 časech, kdy původní signál roste a protíná časovou osu, dosahuje maxima, klesá a protíná časovou osu a dosahuje minima, jak zobrazuje obrázek 2.2.

What happens inside the sensor?



Obrázek 2.2: Měření fázového posuvu u principu nepřímého TOF [18]

Princip nepřímého TOF není zdaleka tak rozšířený jako první uvedený princip, ale nabízí lepší přesnost měření. Srovnání obou metod nabízí obrázek č. 2.3 [18].

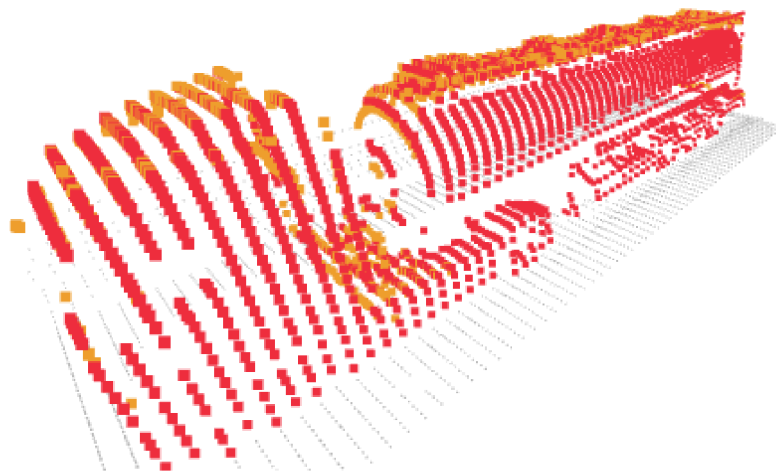


Obrázek 2.3: Schéma srovnání principů měření TOF: přímý TOF nalevo, nepřímý TOF napravo [18]

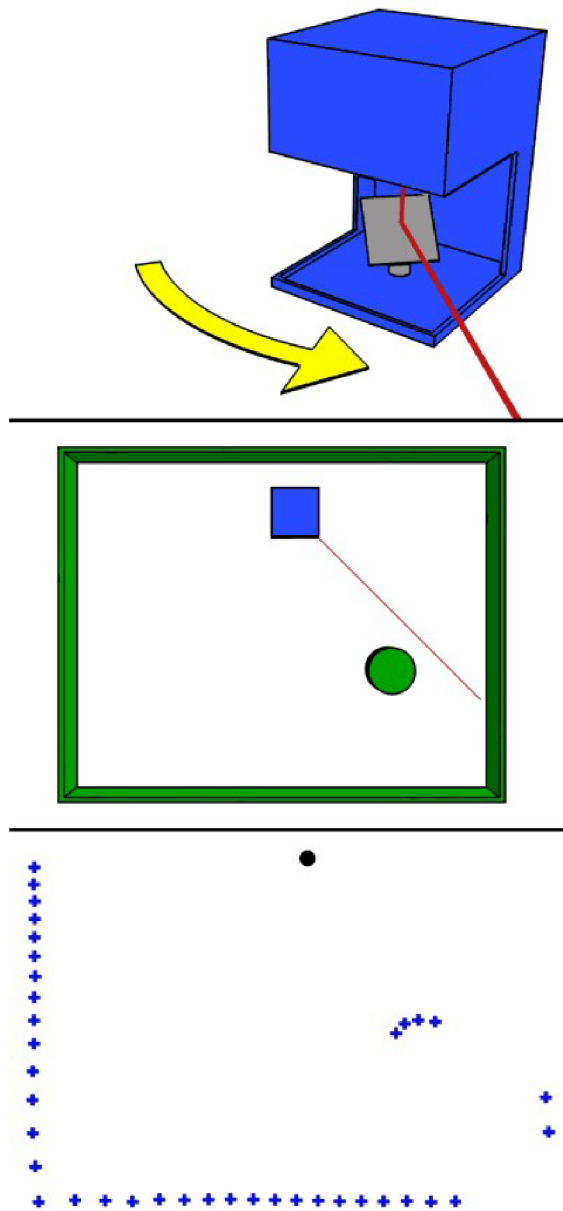
2D LiDAR kombinuje jeden z těchto principů s rotujícím zrcátkem, a vysílá tak množství paprsků, které jsou navzájem otočeny o definovaný úhel. Tyto paprsky se v případě úspěchu vrací do LiDARu, který pak počítá vzdálenost bodu odrazu.

Skenování 2D LiDARu probíhá vždy v definované rovině určené umístěním samotného LiDARu, jak je vidět na obrázku 2.5. Skenování pak probíhá s určitou frekvencí, typicky desítky Hz.

Na obrázku 2.4 je pak možné vidět už naměřená data projíždějícího vozidla pod měřicím stanovištěm firmy Cross, kde oranžové body jsou zachyceny LiDAREm ve stejném pruhu jako vozidlo, červené body jsou zachyceny LiDAREm z vedlejšího pruhu (schéma stanoviště je na obrázku 4.1).



Obrázek 2.4: Ukázka měření LiDAREm na stanovišti firmy Cross



Obrázek 2.5: Ukázka měření LiDARem [35]

### 3. NEJISTOTY MĚŘENÍ

Nejistota měření je definována jako „parametr přidružený k výsledku měření, který charakterizuje rozptyl hodnot, které by mohly být důvodně přisuzovány k měřené veličině s určitou pravděpodobností a které lze pokládat za hodnoty veličiny, které je objektem měření.“ [19] Je to metodika, díky které je možné vyjadřovat přesnost měření. Nejistoty měření tak postupně nahrazují starou metodiku chyb měření [20].

Historie nejistot měření sahá až do osmdesátých let minulého století, kdy Mezinárodní výbor pro váhy a míry (CIPM – *Comité International des Poids et Mesures*) na 70. a 75. zasedání přijal doporučení na náhradu koncepce chyb měření právě nejistotami. Po množství dalších přijatých doporučení byl vytvořen v roce 1990 dokument Západoevropského kalibračního sdružení WECC č. 19. Na základě tohoto dokumentu byly přijímány národní předpisy s cílem zajistit jednotné vyjadřování nejistot. Nejdůležitější dokument byl poté vytvořen v roce 1993 v Ženevě. Byla to směrnice s názvem *Guide to expression of the Uncertainty of Measurement (GUM)*, kterou vydaly mezinárodní metrologické orgány [20], [21].

V současné době jsou v České republice používány definice z normy ČSN EN 60359:2003 *Elektrická a elektronická měřicí zařízení – vyjadřování vlastností* [21].

#### 3.1 Chyby měření

I přes postupné nahrazování chyb nejistotami, se s chybami měření lze stále setkat. Proto budou v této podkapitole shrnuty základní vztahy pro výpočet chyb měření.

Chyba měření je odchylka měřené hodnoty  $X_M$  od konvenčně pravé hodnoty měřené veličiny  $X_S$ . Absolutní chyba  $\Delta_x$  měření je pak uváděna v jednotkách měřené veličiny a definována podle vztahu:

$$\Delta_x = X_M - X_S \quad (3.1)$$

Na zhodnocení měření se místo absolutní chyby často používá relativní chyba  $\delta_x$  dána následovně:

$$\delta_x = \frac{\Delta_x}{X_S} = \frac{X_M - X_S}{X_S} [-] \quad (3.2)$$

Relativní chybu můžeme také vyjádřit procentuálně dle vztahu:

$$\delta_x = \frac{\Delta_x}{X_S} \cdot 100 [\%] \quad (3.3)$$

### 3.1.1 Rozdělení chyb

Chyby lze rozdělit 3 různými způsoby [22]:

- 1) Podle způsobu výskytu
  - Systematické (soustavné) chyby – při opakovaném měření zůstávají stálé nebo se mění s předvídatelně a lze je tak korigovat
  - Náhodné chyby – při opakovaném měření se nahodile mění a nelze je korigovat
  - Hrubé chyby
- 2) Podle místa vzniku
  - Chyba metody měření
  - Chyba měřicího zařízení
  - Chyba použitých etalonů
- 3) Podle příčiny vzniku
  - Chyby experimentátora
  - Chyby způsobené rušivými vlivy
  - Vliv přístroje na měřený objekt
  - Vliv napájení

Vzhledem k zaměření této práce budou popsány detailněji chyby měřicích přístrojů, chyby metody a chyby nepřímých měření. Tyto chyby úzce souvisí se stanovováním nejistoty měření, což je jeden z cílů práce.

### 3.1.2 Chyby měřicích přístrojů

Při výpočtu chyb je třeba zohlednit i ty, které způsobují nepřesnosti samotných měřicích přístrojů. Nazývají se také chyby údaje a dají se vyjádřit v závislosti na tom, jestli je přístroj analogový nebo digitální. V případě analogových přístrojů lze absolutní chybu údaje nejčastěji vyjádřit z třídy přesnosti přístroje udávané výrobcem pomocí následujícího vzorce [19]:

$$\Delta_{MA} = \pm \frac{X_R}{100} \delta_{TP} \quad (3.4)$$

$X_R$ : hodnota měřicího rozsahu

$\delta_{TP}$ : třída přesnosti v procentech

Největší možná relativní chyba údaje se pak určí ze vztahu:

$$\delta_{MA} = \frac{\Delta_{MA}}{X_M} 100 [\%] \quad (3.5)$$

$X_M$ : měřená hodnota

U digitálních přístrojů se nejčastěji udává chyba jako součet mezní chyby z měřené hodnoty  $\delta_M$  a chyby z největší hodnoty měřicího rozsahu  $\delta_R$ . Výsledný vztah pro chybu číslicového přístroje  $\delta_{CMP}$  pak vypadá následovně [19]:

$$\delta_{CMP} = \pm(\delta_M + \delta_R) [\%] \quad (3.6)$$

Jiní výrobci udávají chybu přístroje v jiném formátu (3.7), kde používají opět mezní chybu měřené veličiny doplněnou o chybu udanou v počtu jednotek posledního místa zobrazovače  $d$  [19].

$$\delta_{CMP} = \pm(\delta_M) + |d| [\%] \quad (3.7)$$

Hodnota chyby  $d$  závisí na zvoleném měřicím rozsahu a počtu indikovaných míst zobrazovače  $D$ . Přepočtení chyby  $d$  na chybu z největší hodnoty měřicího rozsahu  $\delta_R$  je [19]:

$$\delta_R = \frac{d}{D} 100 [\%] \quad (3.8)$$

Absolutní chybu údaje číslicového měřicího přístroje v jednotkách měřené veličiny je pak možné z předchozích chyb vyjádřit vztahem [19]:

$$\Delta_{MC} = \pm \frac{\delta_M X_M + \delta_R X_R}{100} \quad (3.9)$$

A relativní chybu údaje digitálního měřicího přístroje vyjadřuje vztah [19]:

$$\delta_{MC} = \pm \frac{\Delta_{MC}}{X_M} \cdot 100 = \pm \left( \delta_M + \delta_R \cdot \frac{X_R}{X_M} \right) [\%] \quad (3.10)$$

### 3.1.3 Chyby metody

Chyby metody vznikají například nezohledněním všech známých vlivů, zjednodušením vztahu pro výpočet nebo zjednodušením zapojení měřicích přístrojů. Chyby metody patří mezi systematické chyby a lze je eliminovat (korigovat) [22].

### 3.1.4 Chyby nepřímých měření

Hodnota hledané veličiny  $Y$  je funkcí  $n$  veličin s označením  $X_1 - X_n$ , a tak je možné veličinu  $Y$  zapsat jako:

$$Y = f(X_1, X_2, \dots, X_n) \quad (3.11)$$

Pokud u veličin  $X_1 - X_n$  jsou známé absolutní chyby údajů  $|\Delta_{X_1}| - |\Delta_{X_n}|$ , je možné vyjádřit maximální chybu hledané veličiny  $Y$  zákonem o hromadění chyb následovně [22]:

$$|\Delta_Y| = \left| \frac{\partial Y}{\partial X_1} \right| |\Delta_{X_1}| + \left| \frac{\partial Y}{\partial X_2} \right| |\Delta_{X_2}| + \dots + \left| \frac{\partial Y}{\partial X_n} \right| |\Delta_{X_n}| \quad (3.12)$$

V jednodušších případech, kdy je hledaná veličina  $Y$  určena jen základními matematickými operacemi mezi dvěma nebo více veličinami, je možné použít následující pravidla [22]:

$$Y = X_1 + X_2 \rightarrow |\Delta_Y| = |\Delta_{X_1}| + |\Delta_{X_2}| \quad (3.13)$$

$$Y = X_1 - X_2 \rightarrow |\Delta_Y| = |\Delta_{X_1}| + |\Delta_{X_2}| \quad (3.14)$$

$$Y = X_1 \cdot X_2 \rightarrow |\delta_Y| = |\delta_{X_1}| + |\delta_{X_2}| \quad (3.15)$$

$$Y = \frac{X_1}{X_2} \rightarrow |\delta_Y| = |\delta_{X_1}| + |\delta_{X_2}| \quad (3.16)$$

## 3.2 Nejistoty měření

Nejistoty jsou oproti chybám měření výpočetně náročnější a stejně jako u chyb na sebe výpočty navazují.

Nejistoty se dělí následovně:

- Nejistota typu A –  $u_A(x)$
- Nejistota typu B –  $u_B(x)$
- Kombinovaná standardní nejistota –  $u_C(x)$
- Rozšířená nejistota –  $U(x)$  [21]

## 3.3 Nejistota typu A

Nejistota typu A se zjišťuje statistickým zpracováním hodnot z opakovaných měření veličiny  $x$  pomocí výběrového rozptylu hodnot. Mírou nejistoty typu A je pak výběrová směrodatná odchylka výběrového průměru. Pokud bylo naměřeno alespoň 10 hodnot, je nejistota typu A dána vztahem [19]:

$$u_A(x) = s_{\bar{x}} = \sqrt{\frac{1}{n(n-1)} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (3.17)$$



Pokud bylo naměřeno méně než 10 hodnot, byla by hodnota nejistoty typu A nespolehlivá. Je proto nutné tento vztah upravit konstantou  $k_s$ . Nejistota se pak počítá podle následujícího vztahu [21]:

$$u_A(x) = k_s s_{\bar{x}} = k_s \sqrt{\frac{1}{n(n-1)} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (3.18)$$

kde hodnota  $k_s$  je určena podle tabulky 3.1:

Tabulka 3.1: Hodnoty  $k_s$  na základě počtu měření  $n$  [21]

$n$	$\geq 10$	9	8	7	6	5	4	3	2
$k_s$	1	1,2	1,2	1,3	1,3	1,4	1,7	2,3	7

Ve vztahu pro nejistotu typu A je  $\bar{x}$  průměrná hodnota z počtu  $n$  měření veličiny  $x_i$ . Počítá se vzorcem:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (3.19)$$

### 3.4 Nejistota typu B

Hodnota nejistoty typu B je určena jiným než statistickým přístupem. Odhaduje se pomocí dostupných informací o měření. Tím jsou například:

- Údaje výrobce měřicího zařízení
- Zkušenosti z předchozích sérií měření
- Zkušenosti s vlastnostmi chování materiálů a techniky a poznatky o nich
- Údaje získané z kalibrace a z certifikátů
- Nejistoty referenčních údajů v příručkách [20]

Výsledná hodnota nejistoty typu B se určuje z dílčích částí  $u_{BZ}(x)$ , které se určují podle následujícího vztahu [20]:

$$u_{BZ}(x) = \frac{D_{max}}{\kappa} \quad (3.20)$$

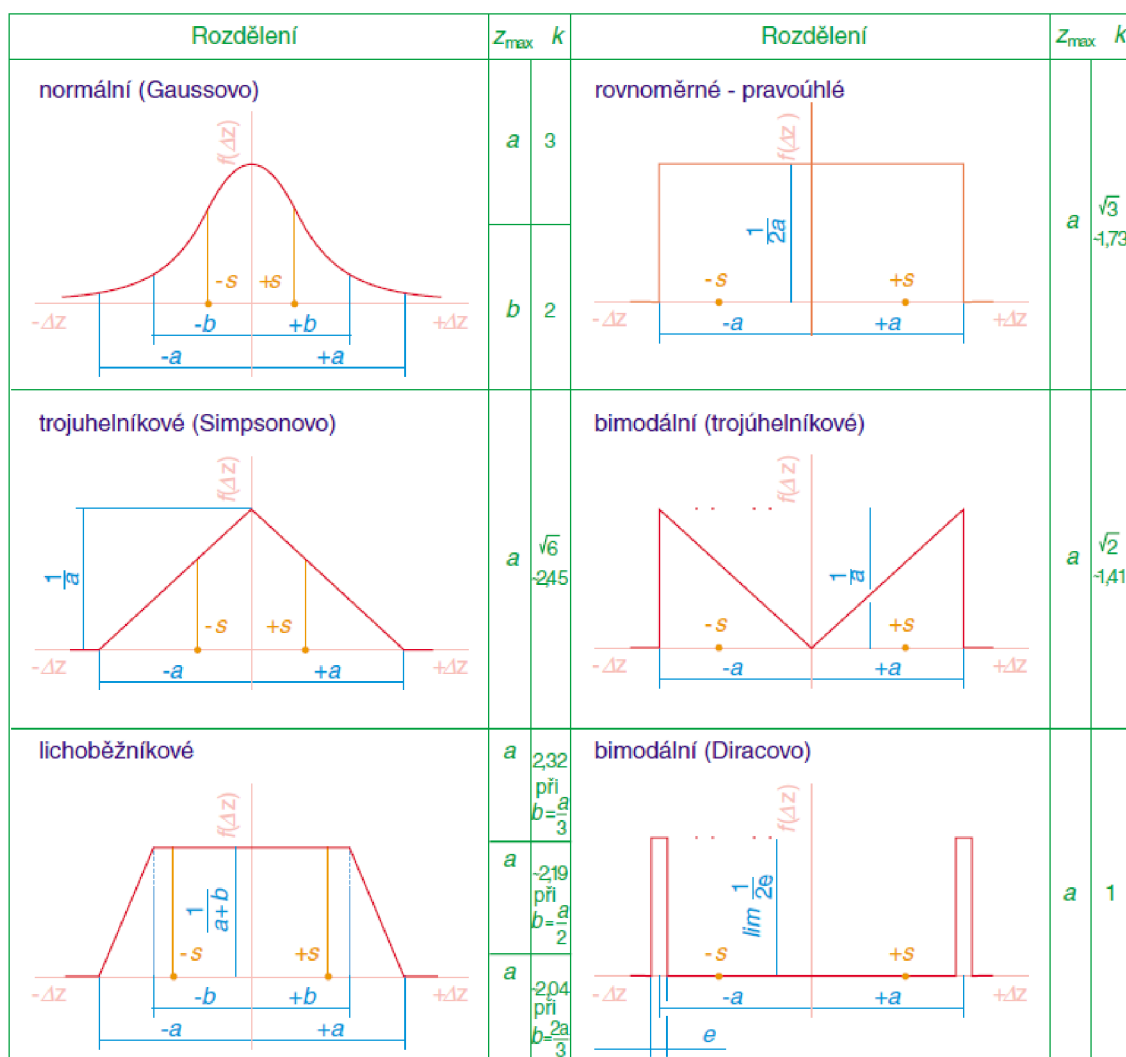
$D_{max}$ : maximální odchylka daného zdroje nejistoty

$\kappa$ : koeficient rozdělení náhodné veličiny

Hodnota koeficientu rozdělení náhodné veličiny je dána tabulkou 3.2 a obrázkem 3.1 [19]:

Tabulka 3.2: hodnoty  $\kappa$  na základě typu rozdělení

Rozložení	$\kappa$	Vhodné pro aplikaci
Normální	2	Přesné přístroje
Rovnoměrné	$\sqrt{3}$	Základní výstupní kontrola výrobce
Trojúhelníkové	$\sqrt{6}$	Vyspělá technologie výrobce
Bimodální Dirac	1	Hystereze
Lichoběžníkové $b = \frac{a}{2}$	2,19	Možnost přesahu
Bimodální Trojúhelník	$\sqrt{2}$	Nonia (posuvka...)



Obrázek 3.1: Zobrazení jednotlivých rozdělení [24]

Celková nejistota typu B  $u_B$  je určena geometrickým součtem jednotlivých zdrojů nejistot  $u_{BZ}$  podle následujícího vztahu [19]:

$$u_B(x) = \sqrt{\sum_{Z=1}^n u_{BZ}(x)^2} \quad (3.21)$$

### 3.5 Kombinovaná nejistota

Vzhledem k tomu, že je u výsledku potřeba vyjádřit nejistotu jednou hodnotou, tak je možné přepočítat nejistoty typu A a B (pokud jsou dostupné) na jednu hodnotou s názvem kombinovaná nejistota  $u_c$ . Je určena geometrickým součtem nejistoty typu A a typu B. [21] Výsledný vztah pro výpočet tedy je:

$$u_c(x) = \sqrt{u_A^2(x) + u_B^2(x)} \quad (3.22)$$

### 3.6 Rozšířená nejistota

V případě, že nestačí mít výslednou hodnotu uvnitř intervalu daného kombinovanou nejistotou s 68% pravděpodobností (v případě normálního rozdělení), je možné z kombinované nejistoty  $u_c$  vypočítat rozšířenou nejistotu  $U$ , která tuto pravděpodobnost výskytu hodnot v intervalu upravuje konstantou  $k_r$  [20]. Výsledná hodnota je pak určena vztahem:

$$U(x) = k_r u_c(x) \quad (3.23)$$

Za konstantu  $k_r$  se dosazují hodnoty dány tabulkou 3.3 [21]:

Tabulka 3.3: Hodnoty  $k_r$  na základě pravděpodobnosti výskytu hodnot [21]

$k_r$	1	2	2.58	3
$p$ [%]	68	95	99	99,7

### 3.7 Nejistota nepřímého měření

Stejně jako u chyb nepřímých měření, i zde je hledaná veličina  $Y$  funkcí  $n$  veličin  $X_1 - X_n$ . Veličinu  $Y$  lze pak vyjádřit jako [19]:

$$Y = f(X_1, X_2, \dots, X_n) \quad (3.24)$$

Veličiny  $X_1 - X_n$  lze přímo měřit nebo známe jejich nejistoty, kovariance nebo odhady z jiných zdrojů. Odhad  $y$  hledané veličiny  $Y$  lze pak vyjádřit vztahem [23]:

$$y = f(x_1, x_2, \dots, x_n) \quad (3.25)$$

Kde  $x_1 - x_n$  jsou odhady veličin  $X_1 - X_n$ .

### 3.7.1 Nejistota nepřímého měření nekorelovaných veličin

Nejistota nepřímého měření  $u(\bar{y})$  odhadu veličiny  $Y$ , která je funkcí nekorelovaných veličin  $X_1 - X_n$  lze určit ze vztahu pro zákon šíření nejistot následovně [23]:

$$u(\bar{y}) = \sqrt{\sum_{i=1}^N \left( \frac{\partial f(X_1, X_2, \dots, X_N)}{\partial X_i} u(\bar{x}_i) \right)^2} \quad (3.26)$$

$f$ : funkční závislost výstupní veličiny  $Y$

$u(\bar{x}_i)$ : nejistota odhadu vstupní veličiny  $X_i$

Vztah lze zjednodušit použitím citlivostních koeficientů  $A_i$ . Nejistota nepřímého měření  $u(\bar{y})$  je pak dána vztahem [23]:

$$u(\bar{y}) = \sqrt{\sum_{i=1}^N A_i^2 u^2(\bar{x}_i)} \quad (3.27)$$

Citlivostní koeficienty  $A_i$  lze pak určit samostatně jednotlivými parciálními derivacemi z funkční závislosti hledané veličiny  $Y$  ze vztahu [23]:

$$A_i = \left. \frac{\partial f(X_1, X_2, \dots, X_N)}{\partial X_i} \right|_{X_1=x_1, \dots, X_N=x_N} \quad (3.28)$$

### 3.7.2 Nejistota nepřímého měření navzájem korelovaných veličin

V případě, že vstupní veličiny  $X_1 - X_n$  funkce hledané veličiny  $Y$  nebo jejich odhady  $x_1 - x_n$  jsou vzájemně korelované, je třeba uvažovat i kovariance mezi vstupními veličinami nebo jejich odhady a vhodně je zohlednit ve vztahu pro výpočet nejistoty  $u(\bar{y})$  [23]:

$$u(\bar{y}) = \sqrt{\sum_{i=1}^N A_i^2 u^2(\bar{x}_i) + 2 \sum_{i=2}^N \sum_{k<i}^{N-1} A_i A_k u^2(\bar{x}_i, \bar{x}_k)} \quad (3.29)$$

$u(\bar{x}_i, \bar{x}_k)$ : kovariance mezi korelovanými odhady  $x_i$  a  $x_k$

$A_i, A_k$ : citlivostní koeficienty definované podle vztahu 3.28

Kovariance mezi korelovanými odhady  $x_i$  a  $x_k$  je možné spočítat například vztahem [23]:

$$u(\bar{x}_i, \bar{x}_k) = \frac{1}{n(n-1)} \sum_{k=1}^n (x_{ik} - \bar{x}_i)(x_{jk} - \bar{x}_j) \quad (3.30)$$

Pro určení kovariance existuje i další metoda, ale vzhledem k tomu, že v práci se s korelací veličin počítat nebude, nebude kapitola dále rozebírána. Další informace jsou k nalezení v článku *Nejistoty v měření III: nejistoty nepřímých měření* v časopise *Automa* (12/2001) [23].

### 3.8 Zaokrouhlování nejistot měření

Nejistoty se zaokrouhlují ve většině případů na 2 platné číslice téměř vždy nahoru. Například pokud hodnota nejistoty z výpočtu je 0,132, bude i tak zaokrouhlena na 0,14, i když podle běžných zaokrouhlovacích pravidel by bylo číslo zaokrouhleno dolů na 0,13. Výjimku tvoří hodnoty, kde by zaokrouhlování snížilo hodnotu nejistoty o méně než 5 %. U takových čísel je možné použít běžná zaokrouhlovací pravidla [21].

### 3.9 Zápis nejistot měření

Při zápisu nejistot je možné použít 3 typy zápisu (3. typ vždy doplňuje jeden z předchozích dvou) [24]:

- Zápis s kombinovanou nejistotou  $u_c(x)$
- Zápis s rozšířenou nejistotou  $U(x)$
- Bilanční tabulka

#### 3.9.1 Zápis s kombinovanou nejistotou $u_c(x)$

Pokud při zápisu výsledku stačí v případě normálního rozložení 68% pravděpodobnost výskytu výsledku v intervalu daným rozšířenou nejistotou, může se tato nejistota použít v zápisu výsledku. V takovém případě je nutno uvést tyto informace [24]:

- Definici měřené veličiny  $Y$
- Odhad  $y$  měřené veličiny  $Y$  a hodnotu kombinované nejistoty  $u_c(y)$  a jejich jednotku

Je možné také případně uvést relativní standardní kombinovanou nejistotu  $u_c(y)/|y|$ ,  $|y| \neq 0$  a také bilanční tabulku.

Příkladem bude měření hmotnosti, kdy zjištěná hmotnost závaží  $m$  o nominální hmotnosti 100 g je 100,02147 g a kombinovaná nejistota  $u_c$  je 0,35 mg. Samotný zápis je pak možné napsat 4 různými způsoby [24]:

- $m = 100,021\ 47\ \text{g} \pm 0,35\ \text{mg}$
- $m = 100,021\ 47\ (35)\ \text{g}$ , kde číslo v závorce je hodnota kombinované nejistoty s dekadickým řádem shodným s řádem posledních dvou číslic hodnoty výsledku
- $m = 100,021\ 47\ (0,000\ 35)\ \text{g}$ , kde číslo v závorce je hodnota kombinované nejistoty ve stejné jednotce jako je výsledná hodnota
- $m = (100,021\ 47 \pm 0,000\ 35)\ \text{g}$ , kde číslo po znaku  $\pm$  je hodnota kombinované nejistoty, ale moc se nepoužívá, protože tento zápis se nejčastěji používá s rozšířenou nejistotou

### 3.9.2 Zápis s rozšířenou nejistotou $U(x)$

Pokud zápis s kombinovanou nejistotou nestačí pro danou situaci, využívá se zápis s rozšířenou nejistotou  $U(x)$ , kde je možné zvolit, s jakou pravděpodobností se bude výsledek v intervalu daným nejistotou vyskytovat. V takovém případě je nutné uvést [24]:

- Podrobnou definici měřené veličiny  $Y$
- Výsledek ve formě  $Y = y \pm U$  včetně jednotky
- Hodnotu koeficientu rozšíření  $k_r$  použitou u výpočtu nejistoty  $U(x)$

Stejně jako u kombinované nejistoty, je možné i zde uvést v případě potřeby relativní rozšířenou nejistotu a bilanční tabulku [24].

Pokud tedy bude použit znova příklad pro zápis, kde hmotnost závaží  $m$  je 100,02147 g a rozšířená nejistota  $U$  je 0,70 mg, bude zápis [24]:

$$m = (100,02147 \pm 0,00070)\ \text{g}, \text{ kde číslo za znakem } \pm \text{ je hodnota rozšířené nejistoty s použitím koeficientu rozšíření } k_r = 2$$

### 3.9.3 Bilanční tabulka

Někdy zápisy kombinovaných nebo rozšířených nejistot může doplňovat i bilanční tabulka. Taková tabulka (3.4) pak vypadá následovně [21]:

Tabulka 3.4: Obecná podoba bilanční tabulky [24]

Veličina $X_q; Y$	Odhad $x_q; y$	Standardní nejistota $u_q(x)$	Typ rozdělení	Koeficient citlivosti $A_q$	Příspěvek ke standardní nejistotě $u_q(y)$ ; nejistota $u(y)$
$X_1$	$x_1$	$u_1(x)$	Podle situace	$A_1$	$u_1(y)$
$X_2$	$x_2$	$u_2(x)$	Podle situace	$A_2$	$u_2(y)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$X_m$	$x_m$	$u_m(x)$	Podle situace	$A_m$	$u_m(y)$
<b><math>Y</math></b>	<b><math>y</math></b>	<b>-</b>	<b>-</b>	<b>-</b>	<b><math>u(y)</math></b>

Zároveň platí vztahy [24]:

$$u_q(y) = A_q u_q \quad (3.31)$$

$$u(y) = \sqrt{\sum_{q=1}^m u_q^2(y)} \quad (3.32)$$

## **4. POUŽITÍ 2D LIDARU V REÁLNÉ SITUACI**

V současné době firma Cross pro měření rozměrů (šířky, výšky a délky) vozidel na svých stanovištích při běžném provozu na používá dva typy LiDARů – Sick LMS511 [25] a Sick LMS111 [26]. První LiDAR je použit zejména na dálnicích, kde by LiDARy s frekvencí snímání nižší než 100 Hz neposkytovaly použitelné výsledky. Model LMS111 je tedy použit pouze na komunikacích s nižší rychlostí průjezdu pod LiDARem.

### **4.1 Srovnání konkurenčních LiDARů**

Jako alternativy byly vybrány LiDARy od čínské firmy VanJee a japonské firmy Hokuyo, které nabízí široký výběr 2D LiDARů. VanJee WLR-711 [27] parametrově odpovídá požadavkům na snímač, ale nemá tak dobré snímací schopnosti ve zhoršených podmínkách (sněžení, déšť, mlha). Zajímavou možností je také mnohem lehčí LiDAR Hokuyo UTM-30LX-FEW [28]. Na komunikace s nižšími rychlostmi průjezdu (do 90 km/h) by se dal použít i poslední LiDAR UGM-50LAP [29] od firmy Hokuyo. Oba LiDARy od firmy Hokuyo ale nedisponují vlastnostmi korekce při nepříznivém počasí.

Z tohoto srovnání LiDARů není tedy možné doporučit firmě jiné zařízení, než momentálně používají. Nicméně nebylo možné zjistit cenu těchto LiDARů, která by mohla zásadně tento výběr ovlivnit.

V tabulce 4.1 budou tyto dva LiDARy srovnány s alternativami od jiných výrobců.



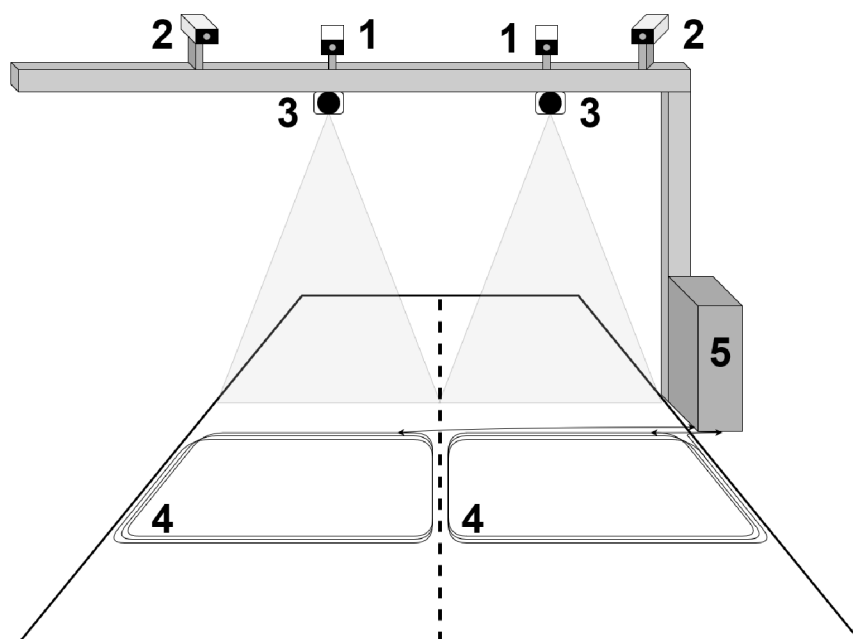
Tabulka 4.1: Srovnání parametrů vybraných LiDARů [25], [26], [27], [28], [29]

Parametr	Sick LMS511	Sick LMS111	VanJee WLR-711	Hokuyo UTM-30LX-FEW	Hokuyo UGM-50LAP
Úhel zorného pole	190°	270°	180°	270°	190°
Frekvence snímání	25/35/50/75//100 Hz	25/50 Hz	50/100 Hz	100 Hz	12,5/25/50 Hz
Úhlové rozlišení	0,6667°	0,25/0,5°	0,25/0,5°	0,65°	0,08/0,16/0,32°
Pracovní oblast	1-80 m	0,5-20 m	0,5-40 m	0,1-30 m	0,1-120 m
Hmotnost	3,7 kg	1,1 kg	4,4 kg	0,3 kg	4,2 kg
Třída ochrany	IP67	IP67	IP68	IP67	IP67
Korekce při snížené viditelnosti	Ano (5 pulzů)	Ano (2 pulzy)	Ano (3 pulzy)	Ne	Ne

## 4.2 Měřicí stanoviště

Firma Cross má po České republice momentálně několik nezávislých stanovišť schopných měřit rozměry vozidel pomocí LiDARů. Tyto stanoviště se mohou lišit, ale ve většině případů má stanoviště v jednom směru tuto instrumentaci (obrázek 4.1):

- 1) Na kovové konstrukci nad vozovkou
  - 2 kamery na rozpoznávání SPZ (1 pro každý pruh) (1)
  - 2 přehledové kamery (1 pro každý pruh) (2)
  - 2 LiDARy ve výšce 6-7,5 m pro měření rozměrů vozidla (1 pro každý pruh) (3)
- 2) Ve vozovce
  - indukční smyčky (4)
- 3) Rozvaděč (5)



Obrázek 4.1: Názorné schéma umístění instrumentace na stanovišti

Na obrázku č. 4.2 je zobrazeno stanoviště u Tlumačova na D55 před instalací nové instrumentace. Současný snímek se bohužel nepodařilo pořídit.



Obrázek 4.2: Záběr na měřicí stanoviště firmy Cross u Tlumačova na D55 [36]

### 4.3 Metody měření jednotlivých rozměrů

Každý rozměr vozidla je měřen LiDARem jiným způsobem a vznikají u nich různé odchylky, které budou shrnuty v další kapitole. Nyní budou představeny jednotlivé metody měření, které budou ve výsledném algoritmu použity.

Měření jednotlivých rozměrů bude předvedeno na situaci průjezdu osobního automobilu na dálnici pod branou s LiDARem Sick LMS511 ve výšce 7,1 metru.

### 4.3.1 Měření délky vozidla

Pro měření délky projíždějícího vozidla měřicím stanovištěm je potřeba znát rychlost vozidla, která se vypočítává pomocí dat o projetí vozidla z indukčních smyček, a frekvence snímání LiDARem, která je zvolena na pevně pro jednotlivá stanoviště. V případě umístění LiDARů s měřicí rovinou jinou než kolmou k vozovce a zároveň kolmou ke směru jízdy, by byly potřeba další informace jako například úhel natočení LiDARu, nicméně firma Cross využívá tento způsob měření, kde tyto informace stačí.

Samotná délka je měřena podle počtu skenů LiDARem, ve kterých je vozidlo zachyceno, jak ukazuje obrázek 4.3, kde svíslé černé úsečky jsou jednotlivé skenované roviny a červené šipky zobrazují prostor, kde se může nacházet měřené vozidlo, které bude mít stejnou vypočtenou délku. Z počtu těchto skenů vozidla  $n$  navýšených o 1 (délka vozidla bude počítána jako mezi středy skenů, které zachytily vozidlo, a skeny, které nezachytily vozidlo), frekvence snímání  $f$  a rychlosti vozidla  $v$  je možné určit délku podle následujícího vzorce:

$$l = (n + 1) \frac{v}{f} [m] \quad (4.1)$$



Obrázek 4.3: Ukázka měření délky vozidla [37]

### 4.3.2 Měření šířky vozidla

Měření šířky je stejně jako měření délky závislé na počtu naměřených detekcí vozidla. Ale na rozdíl od měření délky, kde bylo vozidlo detekováno pomocí celé roviny, zde je vozidlo detekováno pomocí jednotlivých laserových paprsků, jak je vidět na obrázku 4.4. Pokud naměřený bod má kratší vzdálenost od LiDARu než při měření silnice bez vozidla, tak daný bod detekoval vozidlo a je zahrnut do výpočtu šířky. Pokud jsou pak dané body

převedeny z polárních souřadnic do kartézských (LiDAR měří v polárních souřadnicích), je možné určit šířku ze vztahu:

$$w = |d_{n_x} - d_{1_x}| [m] \quad (4.2)$$

$d_{1_x}$ : souřadnice  $x$  prvního bodu detekovaného na vozidle

$d_{n_x}$ : souřadnice  $x$  posledního bodu detekovaného na vozidle

$n$ : počet bodů detekovaných na vozidle



Obrázek 4.4: Ukázka měření šířky vozidla [31]

### 4.3.3 Měření výšky vozidla

Ze stejných dat o vzdálenostech jednotlivých bodů detekovaného vozidla jako u měření šířky, je měřena i výška, ale ta využívá jen výše položené detekované body (tj. body detekované na střeše vozidla, obrázek 4.4). Polární souřadnice detekovaných bodů se opět převedou na kartézské a hodnoty výšky se vhodným filtrem vyfiltrují. Tak zůstanou pro výpočet pouze vhodné hodnoty výšek, ze kterých se maximem určí výška vozidla  $h$ :

$$h = \max d_{i_y} [m] \quad (4.3)$$

$d_{i_y}$ : vyfiltrované výšky střechy vozidla

## 5. ANALÝZA NEJISTOT MĚŘENÍ 2D LIDAREM

Pro analýzu nejistot je nutné získat naměřená data. Vzhledem k velkému množství dat, která stanoviště při každém projetí vozidla sbírají, už není možné používat pro uchovávání jednoduchou tabulku. Data v ní by byla nepřehledná a bylo by složité v tabulce data filtrovat. Bylo proto nezbytné vytvořit databázi. Firma Cross použila řešení uložení dat do relační databáze, pro kterou je možné následné filtrování pomocí jazyka SQL.

### 5.1 Databáze firmy Cross

Pro účely této práce byla vytvořena ve firmě Cross záloha databáze časového úseku od 6. 9. 2021 do 6. 10. 2021 stanoviště na 24,9 km D55 ve směru na Otrokovice, které využívá LiDAR *Sick LMS511* se vzorkovací frekvencí 100 Hz a úhlovým rozlišením  $0,6667^\circ$ . Tato záloha byla obnovena do prázdné databáze *DPdata* ve firmou doporučeném nástroji pro práci s databázemi *PostgreSQL*. Následující vyčítání hodnot už probíhalo v nástroji *pgAdmin 4*. Alternativní možností byla konzolová aplikace *SQL shell (psql)*, což je opět nástroj *PostgreSQL*.

Databáze obsahuje 2 schémata, obsahující navzájem propojené tabulky. Schémata jsou:

- *Raw* – obsahuje tabulky surových dat naměřených na stanovišti
- *Data* – obsahuje tabulky zpracovaných a vypočítaných dat

#### 5.1.1 Schéma Raw

Toto schéma obsahuje 17 tabulek:

- Accelerometer\_event
- Adc\_data\_event
- Adc\_signal\_event
- Adc\_signal\_event\_additional
- Adr\_event
- Dimension\_data\_event
- Dimension\_signal\_event
- Distance\_adc\_signal
- Event
- Image\_event
- Loop\_data\_event
- Loop\_occupancy\_event
- Loop\_signal\_event
- Lpn\_event
- Temperature\_event
- Traffic\_jam\_event
- Version

Z těchto tabulek jsou pro tuto práci použity tabulky *Event*, *Dimension\_signal\_event*. Tabulka *Event* je pro tuto práci použita jako převodní tabulka mezi vozidly a skeny. Pro každé ID skenu (*id*) je zde uvedena časová známka odeslání dat (*stamp*), typ (*type*), ID pruhu silnice (*lane*), status (*status*) a ID vozidla (*vehicle\_id*).

Druhá tabulka použitá pro tuto práci je *Dimension\_signal\_event*. Pro ID skenu (*id*) je zde pole naměřených binárních dat LiDARu (*signal*), ID LiDARu (*sensor\_id*), pořadí

v celkovém počtu skenů (*scan\_count*), první měřený úhel (*start\_angle* – ve stupních), úhlové rozlišení (*resolution* – ve stupních) informace o tom, zda je sken statickou scénou nebo ne (*dc*).

### 5.1.2 Schéma Data

Schéma obsahuje 18 tabulek:

- Adr\_plate
- Axle
- Axle\_group
- Axle\_wheel
- Dimension
- Flag
- Image
- Lane
- License\_plate
- Malenovice
- Measurement\_detail
- Temperature
- Vehicle
- Vehicle\_annotation
- Version
- Violation
- Violation\_axle
- Wheel\_base

Z tohoto schématu jsou v práci využity data z tabulek *dimension*, *vehicle\_annotation*, *vehicle*, *measurement\_detail*. První zmíněná tabulka (*dimension*) obsahuje data pro jednotlivá ID vozidel (*vehicle\_id*) naměřené hodnoty rozměrů v milimetrech (*value*) pro dané rozměry (*type* – 3 = šířka, 4 = výška, 5 = délka).

Tabulka *vehicle\_annotation* obsahuje dopňující informace o vozidlech daného ID (*vehicle\_id*). Mezi tyto informace patří značka a model vozidla (*model*), barva (*color*) a počasí, za kterého vozidlo stanovištěm projelo (*weather*).

Další tabulkou využitou v práci je *vehicle*. Pro každé ID vozidla (*vehicle\_id*) obsahuje informace o časové značce vytvoření záznamu (*timestamp*), ID pruhu silnice (*lane*), rychlosti vozidla v kilometrech za hodinu (*speed*), jeho váze v kilogramech (*total\_weight*), informace o směru vozidla (*direction*), informace o typu vozidla (*ucid* – např. 1 = osobní vozidlo, 23 = tahač s hranatým návěsem, ...), počet os kol vozidla (*axle\_count*) a další informace.

Poslední použitou tabulkou schématu je tabulka *measurement\_detail*. Ta pro každé ID vozidla (*vehicle\_id*) obsahuje detailní informace o měření uložených jako řetězec znaků JSON (*data*). Hlavní struktura těchto dat je zobrazena na obrázku 5.1.

```

1  {
2  "dimensions": {
3    "height": 1423,
4    "length": 5078,
5    "width": 1692
6  },
7  "newSickData": {
8    "height_detail": {...
32  },
33    "height": {
34      "mean": 1423.4208984375,
35      "std": 9.0524320602417
36    },
37    "length": {
38      "mean": 5078.58203125,
39      "std": 129.48689270019531
40    },
41    "length_detail": {...
65  },
66    "width": {
67      "mean": 1692.843505859375,
68      "std": 17.415807723999023
69    },
70    "width_detail": {...
1202  }
1203  },
1204  "pointCloud": {
1205    "1": {...
2850  },
2851    "2": {...
4388  }
4389  },
4390  "sickProcessorVersion": "726c6481 2021-08-20"
4391  }

```

Obrázek 5.1: Základní struktura dat JSON

Kromě dat ve zmíněných tabulkách databáze uchovává i další informace, jako například data indukčních smyček ve vozovce, data kamer pro rozpoznávání poznávacích značek, informace o čase, datu, počasí, teplotě a další.

### 5.1.3 Čtení dat z databáze

Jako ukázkový příklad čtení z databáze poslouží naměřené rozměry vozidel Škoda Octavia iV v konfiguraci sedan. Pro účely diplomové práce jsou uvažována pouze data naměřená za pěkného počasí (tj. slunečno/polojasno, ve dne, suchá vozovka, nesnížená viditelnost). Při uvažování pěkného počasí a vybraného modelu vozidla jsou vyčteny z databáze 3 záznamy o projetí vozidla. Základní informace záznamů (ID vozidla, model, rychlost v km/h, typ rozměru a hodnota naměřeného rozměru) je možné zobrazit pomocí SQL příkazu:

```

SELECT d.vehicle_id, va.model, v.speed, d.type, d.value
FROM data.dimension AS d
INNER JOIN data.vehicle_annotation AS va
ON d.vehicle_id = va.vehicle_id
INNER JOIN data.vehicle AS v
ON va.vehicle_id = v.id
WHERE va.model = 'Skoda_Octavia_IV_Sedan' AND d.type > '2' AND va.weather
= 'Clear'
ORDER BY va.vehicle_id

```



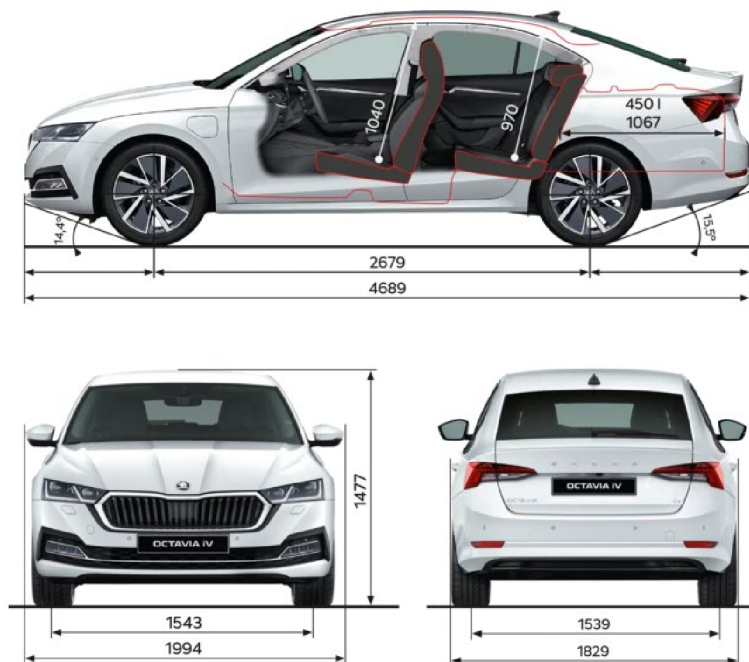
Po zadání zmíněného SQL dotazu se zobrazí následující tabulka (5.1) měřených hodnot:

Tabulka 5.1: Vyfiltrovaná data z databáze pro výpočet nejistot

	vehicle_id	model	speed	type	value
	bigint	text	smallint	smallint	integer
1	1271784	Skoda_Octavia_IV_Sedan	150	3	1636
2	1271784	Skoda_Octavia_IV_Sedan	150	4	1457
3	1271784	Skoda_Octavia_IV_Sedan	150	5	5416
4	1271822	Skoda_Octavia_IV_Sedan	159	3	1692
5	1271822	Skoda_Octavia_IV_Sedan	159	4	1423
6	1271822	Skoda_Octavia_IV_Sedan	159	5	5078
7	1582094	Skoda_Octavia_IV_Sedan	88	3	1721
8	1582094	Skoda_Octavia_IV_Sedan	88	4	1447
9	1582094	Skoda_Octavia_IV_Sedan	88	5	3422

Ze sloupce vehicle\_id je možné vyčíst, že vozidla projela jen 3, protože každému vozidlu je přiřazeno jedinečné identifikační číslo.

Za použití těchto SQL dotazů se zpřístupní hodnoty pro výpočet nejistot. Pro závěrečné porovnání, zda se vypočítané rozměry nachází v intervalu daného nejistotou, je nutné najít oficiální rozměry výrobce. Ten udává rozměry na následujícím schématu (5.2) [30]:



Obrázek 5.2: Schéma rozměrů vozidla Škoda Octavia iV (sedan) [30]



## 5.2 Uvažované nejistoty

Jak již bylo zmíněno, měření rozměrů probíhá tak, že pod branou (měřicím stanovištěm) projede vozidlo v libovolném pruhu a nemusí přitom nijak zpomalovat nebo úplně zastavovat. To sice dává dostatek času LiDARu pro provedení několika měření, ale tato měření není možné brát jako opakované. O opakovaném měření by se teoreticky dalo uvažovat, pokud by byly vozidla dokonalé krychle nebo kvádry, ale vozidla mají k těmto tvarům daleko. Každá série měření LiDARu probíhají na jiné části vozu, která má jiný tvar. Proto není možné uvažovat o nejistotě typu A, která je založena na opakovaném měření. Výpočet bude tedy proveden pouze pro nejistoty typu B.

Některé z nejistot nebudou v diplomové práci řešeny. Jsou jimi nejistoty vzniklé různou odrazivostí různých barev a povrchových úprav vozidel, které nemusí být zahrnuty ve výrobcem daných pracovních podmínkách použitého LiDARu. Další nejistoty, které nebudou v práci řešeny, jsou způsobeny počasím a denní dobou.

Nejistoty, kterými se bude práce zabývat můžou být rozděleny do tří kategorií, podle toho, kterého rozměru se týkají.

### 5.2.1 Nejistoty ovlivňující měření délky

První nejistota ovlivňující měření délky je nejistota dána systematickou chybou měření LiDARu. Ovlivňuje každé měření LiDAREm a hodnota maximální odchylky je dostupná v datovém listu daného LiDARu [25]. Stejně tak každé měření ovlivňuje nejistota dána statistickou chybou a její maximální odchylka je opět uvedena v datovém listu LiDARu.

Nejistotou, která částečně ovlivňuje i měření délky je nejistota dána rozměrem vozidla. Zakřivení vozidla může v krajních případech způsobit, že se odražené paprsky nevrátí zpět do LiDARu. Je to ale poměrně nepravděpodobné. Navíc nejistotu není možné bez rozsáhlé analýzy vyčíslit.

Nejdůležitější nejistotou je nejistota způsobená metodou měření. Jak již bylo zmíněno v podkapitole 4.3.1 Měření délky vozidla, může se kraj přední i zadní části vozidla nacházet kdekoliv mezi skenem, který vozidlo detekoval, a skenem, který vozidlo nedetekoval. Vzhledem k tomu, že pomyslný kraj vozidla je určen jako střed mezi těmito skeny, bude nejistota rovna polovině délky mezi skeny.

Dalšími nejistotami, které budou mít vliv na měření délky jsou nejistoty měření rychlosti a zaokrouhlování rychlosti.

### 5.2.2 Nejistoty ovlivňující měření šířky

Stejně jako u délky, budou i měření šířky ovlivňovat nejistoty systematické chyby, statistické chyby. Dále bude měření šířky ovlivněno nejistotou danou tvarem vozidla. Ta bude u měření šířky podstatnější, protože zde se jedná o odraz jednotlivých paprsků místo celé roviny. Navíc budou paprsky dopadat na kraje vozidla pod úhlem, který ještě zvýší pravděpodobnost, že se neodrazí zpět do LiDARu. Opět ale nebude možné tuto nejistotu vyjádřit kvantitativně.

Poslední uvažovanou nejistotou ovlivňující měření šířky bude samotná metoda měření šířky. Stejně jako u délky, bude i šířka určena na obou stranách vozidla jako střed mezi paprskem, který zachytil vozidlo, a paprskem, který jej nezachytil. Maximální odchylka pak bude polovina tohoto intervalu na obou stranách.

### 5.2.3 Nejistoty ovlivňující měření výšky

I měření výšky budou ovlivňovat nejistoty dané systematickou chybou a statistickou chybou LiDARu. Nejistota dána tvarem vozidla bude mít také určitý vliv, nebude ale zdaleka tak výrazný, jako u měření šířky. Bohužel jej opět nepůjde vyjádřit. A nakonec bude měření ovlivňovat nejistota dána metodou měření výšky. Výška vozidla je určena z mnoha bodů v přibližně stejné výšce. Nejistota měření výšky tedy bude určena ze směrodatné odchylky těchto bodů.

## 5.3 Výpočet nejistot

Názorné výpočty nejistot budou provedeny pro Škodu Octavii iV sedan s ID 1582094. Vztahy pro nejistoty týkající se měření více rozměrů budou vyjádřeny pro obecný rozměr  $x$ .

### 5.3.1 Pracovní podmínky LiDARu

Aby bylo možné provést výpočet nejistot, je nutno ověřit, jestli LiDAR pracuje v podmínkách daných výrobcem. Ty jsou k nalezení v datovém listu, v tomto případě modelu Sick LMS511. Ty jsou přepsány do tabulky 5.2 [25].

Tabulka 5.2: Údaje o prostředí, ve kterých může LiDAR Sick LMS511 pracovat [25]

Odrazivost objektu	2 % ... > 1.000 % (odrazky)
Elektromagnetická kompatibilita (EMC)	IEC 61000-6-2:2016-08 / IEC 61000-6-4:2006+A1:2010
Odolnost proti vibracím	IEC 60068-2-6:2007-12
Provozní okolní teplota	-30 °C ... +50 °C
Odolnost vůči okolnímu světlu	70.000 lx

### 5.3.2 Nejistota dána systematickou chybou LiDARu

Jako první nejistotou typu B bude přímo chyba LiDARu. Tu lze vypočítat ze systematické chyby LiDARu, kterou uvádí výrobce v datovém listu. Tato systematická chyba je maximální odchylkou, která bude přímo použita pro výpočet. Pro Sick LMS511 systematická chyba, a tak i maximální odchylka  $D_{max}$  je  $\pm 25$  mm v rozsahu od 1 do 10 metrů [25]. Tato výška je dána umístěním LiDARů na měřicím stanovišti, což je typicky 6 až 7,5 metru nad vozovkou. Zápis potom vypadá takto:

$$D_{max} = \pm 25 \text{ mm (1 m ... 10 m)} \quad (5.1)$$

Pro LiDAR jako přesný přístroj je možné dle tabulky 3.2 uvažovat normální rozdělení. Při tomto rozdělení  $\kappa$  je hodnota nejistoty typu B:

$$u_{B1}(x) = \frac{D_{max}}{\kappa} = \frac{25}{2} = 12,5 \text{ [mm]} \quad (5.2)$$

Tato nejistota se týká všech rozměrů vozidel.

### 5.3.3 Nejistota dána statistickou chybou LiDARu

Další chybou LiDARu je statistická chyba (v tomto případě maximální odchylka). Ovlivňuje každý rozměr a v rozsahu 1 až 10 metrů nabývá hodnoty 6 mm. Hodnota je opět dána výrobcem v datovém listu [25]. Zápis odchylky tedy je:

$$D_{max} = 6 \text{ mm (1 m ... 10 m)} \quad (5.3)$$

Při normálním rozdělení  $\kappa$  je hodnota nejistoty typu B:

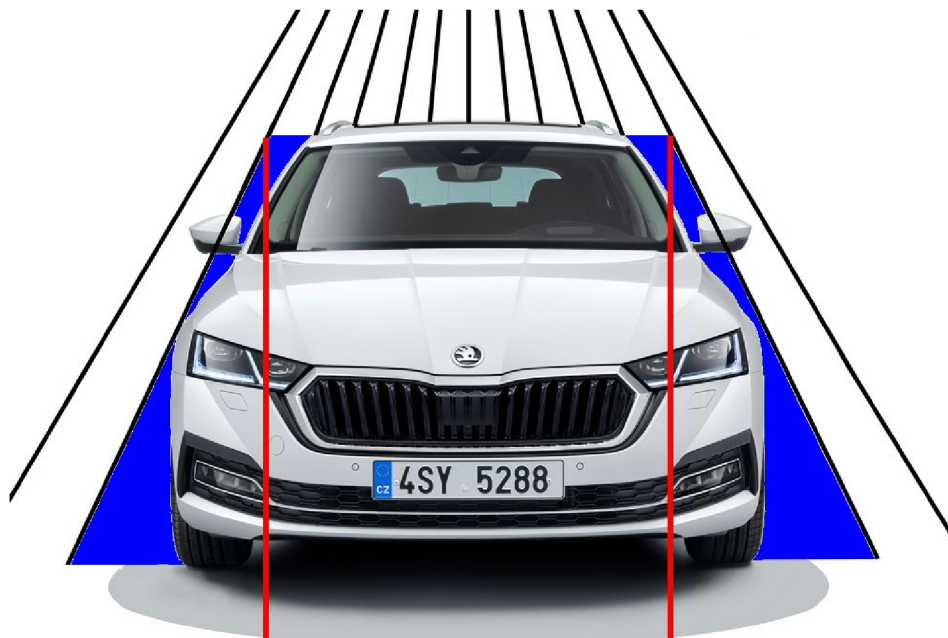
$$u_{B2}(x) = \frac{D_{max}}{\kappa} = \frac{6}{2} = 3 \text{ [mm]} \quad (5.4)$$

### 5.3.4 Tvar vozidla

Velmi důležitým parametrem měření, který tvoří nejistotu, je tvar vozidla. Tento parametr bohužel nejde číselně vyjádřit bez analýzy na předem definovaných tvarech. Taková analýza by musela být velmi rozsáhlá, pokud by měly výsledky být prezentovatelné před zákazníkem.

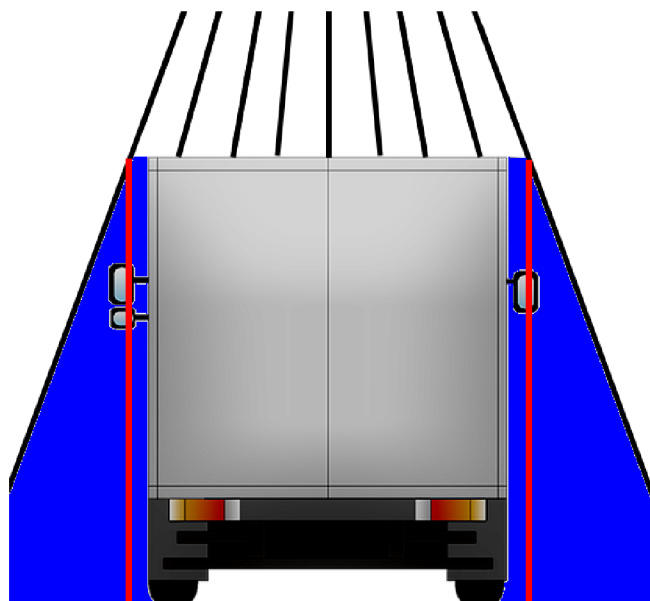
V této práci postačí rozdělit tvary aut na 3 hlavní tvary, u kterých budou načrty s modře znázorněnými oblastmi, kam až může vozidlo sahat a LiDAR by tyto části nezachytil. Červené úsečky budou znázorňovat vypočítanou maximální chybu šířky vozidla. Je potřeba ale počítat s tím, že v reálné situaci jsou paprsky mnohem blíže sebe (svírají úhel definovaného úhlového rozlišení LiDARu) a LiDAR je mnohem výše, než ukazuje náčrt. Hlavními tvary tedy jsou:

- 1) Obecný tvar osobního automobilu se zaoblenými kraji (obrázek 5.3, [31])
  - Při tomto tvaru vznikají malé až střední odchylky při měření šířky. Měření výšky relativně bezproblémové.



Obrázek 5.3: Odchylka způsobená tvarem osobního automobilu [31]

- 2) Obdélníkový tvar s ostrými kraji (kryté návěsy a přívěsy, obrázek 5.4)
  - Odchylky při měření těchto tvarů jsou velmi malé



Obrázek 5.4: Odchylka způsobená tvarem krytých návěsů a přívěsů

- 3) Cisterny s kruhovým nebo elipsovým průřezem
- Měření výšky je relativně bezproblémové, nicméně je průměrováno z menšího množství hodnot. Větší odchylky vznikají při měření šířky. Vzhledem ke špatně nakloněné odrazové ploše paprsků na cisternách vznikají odchylky, které mohou tvořit i polovinu celkové šířky cisterny.
- 4) Obecný navěs nebo přívěs
- Nejkomplikovanější tvary pro určení nejistoty. Mohou to být například otevřené přívěsy/návěsy se sypkými látkami, u kterých je většinou správně detekována šířka a délka, ale velmi komplikované je měření výšky. Dalším případem mohou být přívěsy a návěsy například s vertikálně postavenými deskami, sklem a podobnými předměty. U takových případů je téměř nemožné spolehlivě určit nejistotu a musely by být řešeny jednotlivě případ od případu.

### 5.3.5 Délka vozidla

Pro zjištění maximální odchylky při měření délky vozidla jsou potřeba 2 parametry – rychlost vozidla a frekvence snímání LiDARu. Rychlost je uvedena v databázi v kilometrech za hodinu, takže je nutné ji převést na metry za sekundu. A frekvence snímání je na zvoleném stanovišti 100 Hz. Odchylku pro nejistotu měření délky lze pak vypočítat podle vztahu:

$$D_{max} = \frac{v_{mps}}{f} = \frac{\frac{v_{kph}}{3,6}}{f} = \frac{v_{kph}}{3,6} = \frac{159}{100} = 0,24 \bar{4} [m] \quad (5.5)$$

Při uvážení rovnoměrného rozdělení  $\kappa$  je možné vypočítat nejistotu měření následovně:

$$u_{B3}(l) = \frac{D_{max}}{\kappa} = \frac{0,441\bar{6}}{\sqrt{3}} = 0,141 [m] = 141 [mm] \quad (5.6)$$

### 5.3.6 Měření rychlosti

Jak je vidět na vztahu 5.5, měření délky je ovlivněno měřením rychlosti vozidla. Z dostupných informací z firmy Cross je měření rychlosti realizováno přes vážní senzory ve vozovce a maximální odchylka, která tak může vzniknout je 0,1 km/h. Maximální odchylka délky při frekvenci snímání LiDARu 100 Hz bude:

$$D_{max} = \frac{\frac{v_{kph}}{3,6}}{f} = \frac{\frac{v_{kph}}{3,6}}{f} = \frac{0,1}{\frac{3,6}{100}} = 0,0002\bar{7} [m] \quad (5.7)$$

Při uvážení rovnoměrného rozdělení vzniká nejistota:

$$u_{B4}(l) = \frac{D_{max}}{\kappa} = \frac{0,0002\bar{7}}{\sqrt{3}} = 0,0001604 [m] = 0,16 [mm] \quad (5.8)$$

Vzniklá nejistota je tak zanedbatelná. Do celkového výpočtu navíc nemůže být ani zahrnuta vzhledem k tomu, že dále probíhá zaokrouhlení rychlosti, které je zdrojem větší nejistoty.

### 5.3.7 Zaokrouhlování rychlosti

Měření délky dále také ovlivňuje zaokrouhlování rychlosti. Jak bylo zjištěno při výpočtu nejistoty měření rychlosti, je rychlost určena dostatečně přesně, ale při zápisu do databáze jsou hodnoty zaokrouhleny na celá čísla. Takto může vzniknout odchylka od reálné hodnoty až o 0,5 km/h. Při frekvenci snímání 100 Hz je maximální odchylka:

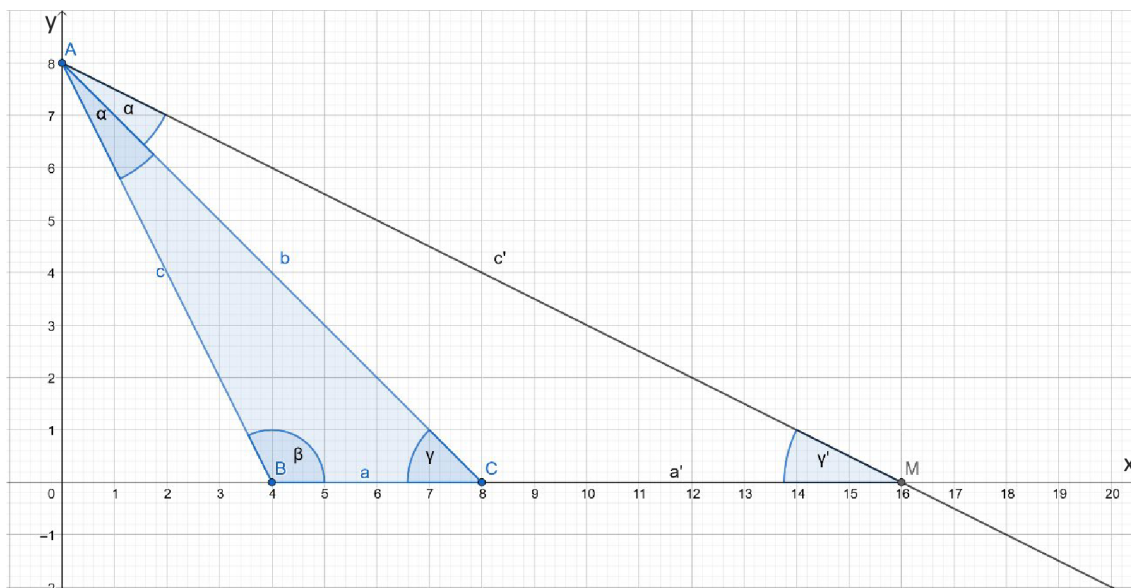
$$D_{max} = \frac{\frac{v_{kph}}{3,6}}{f} = \frac{\frac{v_{kph}}{3,6}}{f} = \frac{0,5}{\frac{3,6}{100}} = 0,0013\bar{8} [m] \quad (5.9)$$

Při rovnoměrném rozdělení bude vzniklá nejistota:

$$u_{B5}(l) = \frac{D_{max}}{\kappa} = \frac{0,0013\bar{8}}{\sqrt{3}} = 0,000802 [m] = 0,8 [mm] \quad (5.10)$$

### 5.3.8 Šířka vozidla

Maximální odchylku měření šířky vozidla je možné určit ze názorného schématu na obrázku 5.5.



Obrázek 5.5: Schéma pro výpočet odchylky měření šířky vozidla

Schéma zobrazuje část měření okrajových bodů vozidla LiDARem. Byla u něj upravena perspektiva, aby bylo lépe poznat, které vzdálenosti se budou v postupu měřit. Osa  $x$  reprezentuje šířku a osa  $y$  výšku (v reálné situaci udávané v metrech). Bod  $A$  reprezentuje LiDAR, bod  $B$  je předposlední bod naměřený na vozidle, bod  $C$  je ten poslední. Polopřímka  $c'$  je paprsek, který už nezachytil vozidlo a bude se tak nejspíš odrážet od vozovky. Průsečík této polopřímky (paprsku LiDARu) s přímkou tvořenou body detekovanými na střeše vozidla tvoří bod  $M$ .

Maximální odchylka při měření šířky může tedy vzniknout tak, že by paprsek  $c'$  těsně nezachytil vozidlo. Odchylka tak bude mít hodnotu délky úsečky  $a'$ . Vzhledem k tomu, že je znám pouze úhel  $\alpha$  (úhlové rozlišení LiDARu,  $0,667^\circ$ ) a změřené výšky vozidla  $c$  a  $d$ , bude nutné použít *sinovou větu* (vztah 5.11 [32]) a *kosinovou větu* pro stranu  $a$  (vztah 5.12 [32]).

$$\frac{a}{\sin \alpha} = \frac{b}{\sin \beta} = \frac{c}{\sin \gamma} \quad (5.11)$$

$$a^2 = b^2 + c^2 - 2bc \cos \alpha \quad (5.12)$$

Ze sinové věty je možné vyjádřit úhel  $\beta$ . Pak už pomocí dopočítáním do  $180^\circ$  jde vyjádřit úhly  $\gamma$  a  $\gamma'$ . Znovu dosazení do sinové věty (jak ukazuje vztah 5.11) povede k vyjádření délky úseček  $a$  a  $a'$ . Kombinace všech těchto vztahů tak vytvoří výsledný vztah 5.13 pro výpočet maximální odchylky na jedné straně:

$$D_{\max\_i} = a' = \frac{c \sin 2\alpha}{\sin \left( 180^\circ - 2\alpha - \arcsin \frac{b \sin \alpha}{\sqrt{b^2 + c^2 - 2bc \cos \alpha}} \right) - \sqrt{b^2 + c^2 - 2bc \cos \alpha}} [m] \quad (5.13)$$

K odchylce dochází na obou stranách vozidla, takže pro určení odchylky pro obě strany je potřeba vypočítat odchylku pro každou stranu zvlášť a pak z nich určit průměr. Je to z důvodu, že šířka vozidla bude určena jako střed mezi paprsky, které se odrazily od vozidla a paprsků, které se už od vozidla neodrazily. Výsledná odchylka je tedy:

$$D_{\max} = \frac{D_{\max\_1} + D_{\max\_2}}{2} [m] \quad (5.14)$$

Pro naměřené délky  $c = 6127$  mm a  $b = 6155$  mm bude maximální odchylka  $D_{\max 1} = 75$  mm. Pro rovnoměrné rozdělení s  $D_{\max 1} = D_{\max 2}$  bude pak nejistota:

$$u_{B6}(w) = \frac{D_{\max}}{\kappa} = \frac{75}{\sqrt{3}} = 0,04330 [m] = 43,3 [mm] \quad (5.15)$$

### 5.3.9 Výška vozidla

Při měření výšky vozidla se vychází ze stejného intervalu vyfiltrovaných hodnot reprezentujících výškově dominantní část vozidla, jako u určování výšky. Z těchto hodnot (výšky  $h_j$  a průměrná výška  $h_{avg}$ ) je možné určit  $D_{\max}$  pro daný LiDAR z maxima směrodatných odchylek podle vztahu:

$$D_{\max\_i} = \max \left( \sqrt{\frac{1}{n} \sum_j^n (h_j - h_{avg})^2} \right) [m] \quad (5.16)$$

Z hodnot  $D_{\max\_1} = 14,1183$  mm a  $D_{\max\_2} = 15,6747$  mm (pro ID vozidla 1582094) je vybrána větší ( $D_{\max} = 15,6747$  mm) a ta je dosazena do vztahu 5.17. Při normálním rozdělení je hodnota nejistoty:

$$u_{B7}(h) = \frac{D_{\max}}{\kappa} = \frac{15,6747}{2} = 7,84 [mm] \quad (5.17)$$

## 5.4 Výsledek měření

Výsledek měření je nutno uvést pro všechny rozměry. Naměřené rozměry jsou pro tento příklad určeny ze záznamu o vozidle s ID 1582094. Rozměry jsou myšleny délkou  $l$ , šířkou  $w$  a výškou  $h$ .



$$l = 3422 \text{ mm}; w = 1738 \text{ mm}; h = 1428 \text{ mm} \quad (5.18)$$

Jak již bylo řečeno, nejistotu A není možné určit, proto bude přeskočen její výpočet, a vypočtena bude rovnou nejistota typu B. Ta je dána geometrickým součtem dílčích nejistot, které daný rozměr ovlivňují. Pro jednotlivé rozměry je to tedy:

$$\begin{aligned} \mathbf{u_B(l)} &= \sqrt{u_{B1}^2(l) + u_{B2}^2(l) + u_{B3}^2(l) + u_{B5}^2(l)} \\ &= \sqrt{12,5^2 + 3^2 + 141^2 + 0,8^2} = \mathbf{141,59 [mm]} \end{aligned} \quad (5.19)$$

$$\begin{aligned} \mathbf{u_B(w)} &= \sqrt{u_{B1}^2(w) + u_{B2}^2(w) + u_{B6}^2(w)} = \sqrt{12,5^2 + 3^2 + 43,3^2} \\ &= \mathbf{45,17 [mm]} \end{aligned} \quad (5.20)$$

$$\begin{aligned} \mathbf{u_B(h)} &= \sqrt{u_{B1}^2(h) + u_{B2}^2(h) + u_{B7}^2(h)} = \sqrt{12,5^2 + 3^2 + 7,84^2} \\ &= \mathbf{15,06 [mm]} \end{aligned} \quad (5.21)$$

Bez jakékoliv nejistoty typu A je kombinovaná nejistota  $u_c$  přímo rovna vypočtené nejistotě typu B:

$$\mathbf{u_c(l)} = u_B(l) = \mathbf{141,59 \text{ mm}} \quad (5.22)$$

$$\mathbf{u_c(w)} = u_B(w) = \mathbf{45,17 \text{ mm}} \quad (5.23)$$

$$\mathbf{u_c(h)} = u_B(h) = \mathbf{15,06 \text{ mm}} \quad (5.24)$$

Vzhledem k tomu, že rozměry vozidla by se měli v ideálním případě blížit středu intervalu, který nejistota definuje, je možné počítat s normálním rozdělením pravděpodobnosti. U kombinované nejistoty se tedy bude daný rozměr v intervalu vyskytovat s pravděpodobností 68 %. Aby byla tato pravděpodobnost vyšší, konkrétně 95 %, bude použit koeficient rozšíření  $k_r = 2$ . Rozšířená nejistota  $U$  pak bude:

$$\mathbf{U(l)} = k_r u_c(l) = 2 \cdot 141,59 \text{ mm} = \mathbf{283,18 \text{ mm}} \quad (5.25)$$

$$\mathbf{U(w)} = k_r u_c(w) = 2 \cdot 45,17 \text{ mm} = \mathbf{90,34 \text{ mm}} \quad (5.26)$$

$$\mathbf{U(h)} = k_r u_c(h) = 2 \cdot 15,06 \text{ mm} = \mathbf{30,12 \text{ mm}} \quad (5.27)$$

Rozšířené nejistoty se pro výsledek zaokrouhlují na 2 platná místa nahoru, až na výjimky, kde by zaokrouhlení dolů způsobilo snížení nejistoty o méně než 5 % [21]. To

je případ nejistot všech rozměrů. Pro vozidlo s ID 1582094 jsou tedy rozměry změřeny následovně:

$$l = (3422 \pm 283) \text{ mm} \quad (5.28)$$

$$w = (1738 \pm 90) \text{ mm} \quad (5.29)$$

$$h = (1428 \pm 30) \text{ mm} \quad (5.30)$$

Pro porovnání, reálné rozměry vozidla jsou  $l = 4689$  mm,  $w = 1829$  mm a  $h = 1477$  mm. V tomto případě ani jeden reálný rozměr neleží ve vypočítaném intervalu. V případě šířky a výšky by tyto rozměry v intervalu byly, pokud by se zvýšila konstanta rozšíření. Délka je ale změřena s absolutní chybou 1267 mm, a to i když vozidlo jelo podle naměřených dat rychlostí 88 km/h. Je zřejmé, že zde nejspíš vznikla chyba při měření. Není ale úplně jasné, jaká chyba to byla. Na základě tohoto měření tedy nejde rozhodnout o správnosti určení nejistot. Pro vytvořený algoritmus popsany v kapitole 6 bude použito větší množství dat, na kterých o této správnosti rozhodnout půjde.

## 6. VYTVOŘENÝ ALGORITMUS

Algoritmus pro stanovení rozměrů vozidel a nejistot měření (dále program) byl vytvořen v LabVIEW verze 2021, což je grafické programovací prostředí vhodné zejména pro zpracování naměřených dat [33].

Program byl vytvořen jako projekt LabVIEW založený na šabloně *Simple State Machine* [34]. Šablona obsahuje ve struktuře projektu hlavní VI (ekvivalent skriptu v běžných programovacích jazycích) s názvem *Main*, ve kterém je základní stavový automat. Ten je v blokovém diagramu tvořen *Case* strukturou uvnitř *While* smyčky. Přes posuvné registry si automat mezi stavy předává informace o dalším stavu a vzniklých chybách.

Výchozí stavový automat také obsahuje základní stavy jako inicializace, čekání, ukončení a dva prázdné uživatelské stavy, ale žádný z těchto stavů (až na ukončení) není ve výsledném automatu použit.

### 6.1 Stavový diagram

Od počátku návrhu programu bylo jasné, že program bude založen jako stavový automat. Hlavním důvodem byl samotný postup zpracování dat, který lze snadno převést na cyklus nutných kroků – první je potřeba přečíst data, pak je správně upravit, vypočítat zadané neznámé a nejlépe i data zobrazit. V případě této práce je to přečtení dat z databáze, přepočítání těchto hodnot na správný formát, výpočet rozměrů a nejistot a také zobrazení dat do grafu.

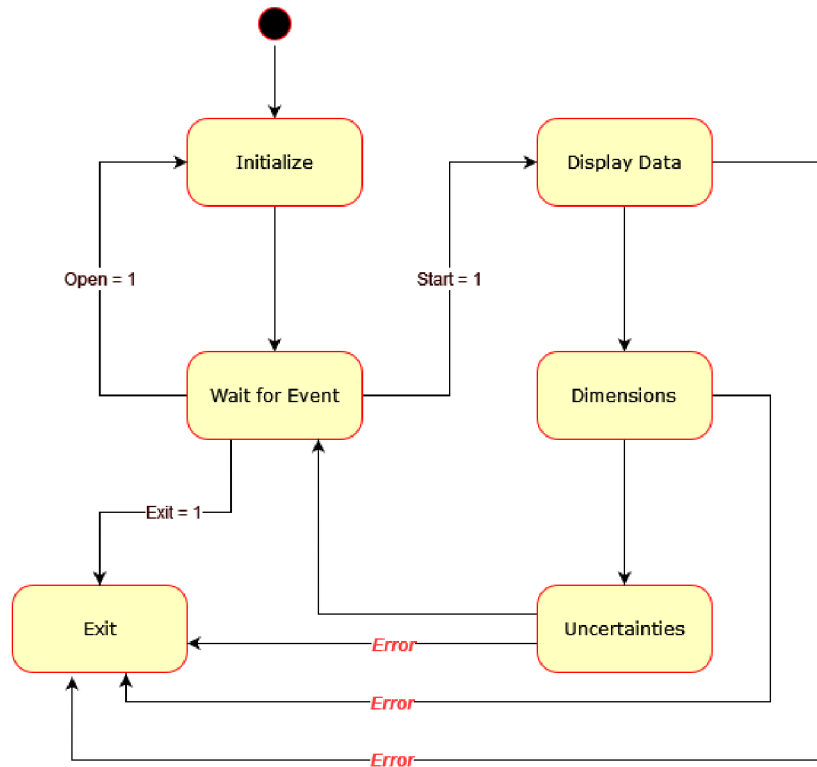
Dalším důvodem pro použití stavového automatu bylo i využití LabVIEW, ve kterém lze snadno vytvořit nepřehledný kód. Stavový automat poskytne určitou přehlednost a z nepřehledných částí mohou být vytvořeny podprogramy.

Výsledný program pak obsahuje tyto stavy:

- *Initialize*
- *Display Data*
- *Dimensions*
- *Uncertainties*
- *Wait for Event*
- *Exit*

Přechody mezi stavy popisuje diagram na obrázku 6.1. První stav, který nastane po startu programu je stav *Initialize*. Pak program přejde do stavu čekání (*Wait for Event*), kde program čeká na uživatele, až vykoná určitou akci. Zde může nastat několik scénářů. Pokud uživatel stiskne tlačítko *Exit*, program přejde do stavu *Exit* a bude bezpečně ukončen. Pokud přepne přepínač pro volbu dat, přejde program zpět do stavu *Initialize*. A pokud uživatel vybere ID vozidla a stiskne tlačítko *Start*, přejde program do stavu

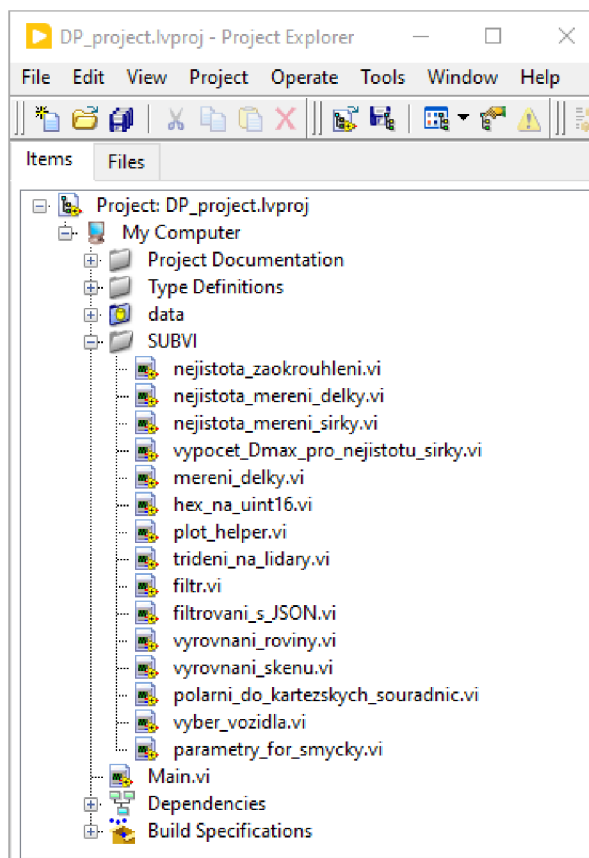
*Display Data*, jinak je ukončen chybou (v případě nezvolení ID vozidla). Ve stavu *Display Data* program zůstane do doby, než dokončí všechny operace (filtrování vstupních dat, přepočty hodnot, zobrazení dat do grafů) a pak automaticky přejde do stavu *Dimensions*. Zde se provedou výpočty rozměrů vybraného vozidla a program pak přejde do stavu *Uncertainties*, kde provede výpočty nejistot. Po provedení výpočtů nejistot program automaticky přejde do stavu *Wait for Event*.



Obrázek 6.1: Stavový diagram programu

## 6.2 Strom projektu

Na obrázku 6.2 je zobrazen strom vytvořeného programu. Hlavní soubor má název *Main.vi* a obsahuje stavový automat. Soubory s příponou VI ve složce *SUBVI* jsou podprogramy (nebo také SUBVI), které budou jednotlivě popsány později. Ve stromu projektu je také složka *data*, ve které se nachází soubory CSV se vstupními daty a textové soubory, které uchovávají data ve formátu JSON pro jednotlivé ID vozidel. Tato data nemohou kvůli své délce a formátu jako jeden řetězec být uloženy v souborech CSV.



Obrázek 6.2: Strom projektu LabVIEW

### 6.3 Blokový diagram v LabVIEW

Jak již bylo zmíněno, program je vytvořen na základě šablony LabVIEW *Simple State Machine*. V blokovém diagramu je tedy hlavní *While* smyčka, ve které se nachází *Case* struktura s 6 stavy. Náhled na stavový automat ve stavu *Display Data* je k dispozici v příloze B. *While* smyčka pak předává hodnoty přes posuvné registry. Těmi jsou:

- Parametry měření
  - Struktura (*Cluster*) 5 hodnot (Systematická chyba [mm], Statistická chyba [mm], Frekvence snímání [Hz], Úhlové rozlišení [°], První měřený úhel [°]), všechny ve formátu *Double*. Systematická chyba, statistická chyba a frekvence snímání je určena uživatelem, úhlové rozlišení a první měřený úhel vyčten z databáze
- Rozměry a nejistoty
  - Struktura (*Cluster*) 6 hodnot (Nejistota měření délky [mm], Nejistota měření šířky [mm], Nejistota měření výšky [mm], Délka [mm], Šířka [mm], Výška [mm]), všechny ve formátu *Double*.

- Data obou LiDARů
  - Pole struktur tvořených dvěma 1D poli hodnot typu *Uint16* pro oba LiDARy
- Nezpracovaná filtrovaná data
  - 2D pole hodnot typu *String* získaných ze SUBVI *vyber\_vozidla.vi*, obsahuje data pouze vybraného ID vozidla (bez hlavičky)
- Cesta k datům
  - Adresa aktuálně používaného souboru dat (formát *File path*)
- Indexy vozidla
  - 1D pole struktur tvořených 4 prvky reprezentujícími první, druhý, předposlední a poslední index skenu zachycující vozidlo (formát *Int32*)
- Nejistota výšky
  - Hodnota typu *Double* reprezentující vypočítanou nejistotu typu B měřené výšky
- Stav programu
  - Typová definice hodnot *Uint16* reprezentujícími jednotlivé stavy
- JSON data
  - Struktura 9 hodnot a polí vytvořených čtením dat ve formátu *JSON*
- ID vozidla
  - Uživatelem vybrané ID vozidla předávané v typu *String*
- Data bez hlavičky
  - 2D pole dat vyčtených z vybraného souboru CSV s odebranou hlavičkou (všechny data ve formátu *String*)
- Chyba
  - Slouží pro předávání vzniklých chyb v programu, a tak i bezpečnému ukončení v případě vzniklé chyby, data se předávají jako speciální chybová struktura

Z těchto posuvných registrů je 6 inicializováno ještě před spuštěním stavového automatu. Pro posuvné registry *Parametry měření* a *Rozměry a nejistoty* je potřeba definovat strukturu (*Cluster*) pořadím, názvy a formáty prvků tak, aby se nemusela definovat pokaždé při změně vstupních dat ve stavu inicializace. Posuvný registr *Parametry měření* je inicializován výchozími hodnotami pro měření s použitým LiDAREm Sick LMS511 - 25 (systematická chyba [mm]), 6 (statistická chyba [mm]), 100 (frekvence snímání [Hz]), 0,667 (úhlové rozlišení [°]) a 10,333 (první měřený úhel [°]). Další posuvný registr (*Rozměry a nejistoty*) je inicializován nulovými hodnotami. Další posuvné registry jsou inicializovány také nulovými hodnotami, aby mohly být použity přímo v prvním stavu (*Initialize*).

Mimo stavový automat se nachází ještě blok pro řízení záložek, ve kterých jsou na ovládacím panelu grafy a vstupní data, a také bloky na zpracování chyby, které jsou součástí šablony *Simple State Machine*.

### 6.3.1 Stav inicializace (*Initialize*)

Stav inicializace zahájí předání dat z posuvných registrů a volba souboru uživatelem, viz kapitola 6.5. To povede k předání cesty k souboru LabVIEW funkci *Read Delimited Spreadsheet.vi*, která přečte vybraný soubor CSV a uloží data do 2D pole dat ve formátu *String*. Pomocí základních funkcí pro práci s poli je odebrán první řádek s hlavičkou a vyčteny parametry *První měřený úhel* a *Úhlové rozlišení*.

Pro výběr vozidla, který bude následovat v dalším kroku, je potřeba vytvořit seznam vhodných ID vozidel. Pomocí funkce *Index array* a *Remove Duplicates From ID Array.vim* je vytvořen seznam všech možných ID, který je upřesněn *For* smyčkou. Zde jsou data vyfiltrována pro každé ID pomocí SUBVI *vyber\_vozidla.vi* a následně je zkontrolováno, jestli data obsahují hodnoty pro oba LiDARy. Výstupem ze smyčky je pole ID vozidel, které obsahují všechna data potřebná ke správnému zobrazení.

Dále se v tomto stavu načtou uživatelem zvolené hodnoty systematické chyby, statistické chyby a frekvence snímání.

### 6.3.2 Stav zobrazování dat (*Display Data*)

Většina operací celého programu se děje právě v tomto stavu. Je to z důvodu, aby se nemusely provádět další *For* smyčky jen pro zpracování dat pro zobrazení jednotlivých 2D rovin skenů. Proto jsou všechny výpočty vzdáleností, převody dat a srovnání roviny grafu prováděny zde. Náhled na stavový automat v tomto stavu je v příloze B.

Jako první je v tomto stavu spuštěno SUBVI *vyber\_vozidla.vi*, ve kterém se vstupní 2D pole dat vyfiltruje na data jen pro vybrané ID vozidla. Výstupní filtrovaná data pak vstupují do SUBVI *trideni\_na\_lidary.vi*, kde se vstupní naměřené hodnoty z pole roztrídí na data prvního a druhého LiDARu a vyčte se ID pruhu silnice. Z filtrovaných dat je také vyčtena hodnota rychlostí, ze které podle rovnice 5.5 je vypočítána vzdálenost mezi rovinami jednotlivých skenů. Pro tuto rovnici je nutné znát i frekvenci snímání. Ta je společně s hodnotami prvního měřeného úhlu a úhlového rozlišení vyčtena z *Clusteru* parametrů měření.

Zároveň se bude provádět SUBVI *filtrvani\_s\_JSON.vi*, ze kterého po vstupu vybraného ID vozidla a samotném výpočtu popsaného v podkapitole 6.3.7.10, je získán *Cluster* dat získaného z JSON dat pro vybrané ID vozidla.

Hlavní částí tohoto stavu je hlavní *For* smyčka (pro srozumitelnější popis tohoto stavu bude nazývána vnější smyčkou), do které vstupují naměřené hodnoty obou LiDARů, ID pruhu, ve kterém vybrané vozidlo jede, hodnota prvního měřeného úhlu, úhlové rozlišení a vzdálenost mezi rovinami skenů. V této smyčce se volá SUBVI *parametry\_for\_smycky.vi*, ve kterém se spočítá počet iterací a délka řetězce dat LiDARů. Zároveň se ve vnější smyčce vynásobí vzdálenost mezi rovinami měření tisícem (převod na milimetry) a aktuální iterací smyčky. Tím se bude vytvářet pole hodnot na časové ose.

Vnější smyčka obsahuje vnitřní *For* smyčku, která slouží pro výpočty naměřených hodnot vozidla pro aktuální hodnotu času. Počet iterací je dán výstupní hodnotou SUBVI

*parametry\_for\_smycky.vi*. Vykonává se v ní převod hexadecimálních znaků z databáze na hodnoty typu *Uint16* pomocí SUBVI *hex\_na\_uint16.vi*. Toto SUBVI je ve vnitřní smyčce dvakrát (pro výpočet dat obou LiDARů) a vstupuje do něj kromě řetězce hexadecimálních znaků jednotlivých LiDARů také délka tohoto řetězce a hodnota *i* aktuální iterace. Výstupem je hodnota vzdálenosti *Uint16* pro aktuální čas. Tyto hodnoty vystupují z vnější smyčky a indexují se na pole.

Ve vnitřní smyčce také probíhá výpočet úhlů pro srovnání zobrazení pomocí SUBVI *vyrovnavani\_roviny.vi*. Vstupuje do něj ID pruhu, ve kterém se vozidlo nachází a vystupují z něj korekční úhly pro oba LiDARy, které se přičítají k úhlům vstupujícím do dvou SUBVI *polarni\_do\_kartezskych\_souradnic.vi* podle rovnice:

$$\varphi_{Lx} = 360^\circ - \varphi_1 - (\delta \cdot i_{in}) + \varphi_{kLx} \quad (6.1)$$

$\varphi_{Lx}$ : úhel pro přepočítání z polárních souřadnic do kartézských pro LiDAR x

$\varphi_1$ : první měřený úhel LiDAREm

$\delta$ : úhlové rozlišení LiDARu

$i_{in}$ : iterace vnitřní smyčky

$\varphi_{kLx}$ : korekční úhel pro LiDAR x

Dalším krokem je přepočítání hodnot z polárních souřadnic na kartézské v SUBVI *polarni\_do\_kartezskych\_souradnic.vi*, které je voláno pro každý LiDAR. Vstupem vzdálenosti je výstupní hodnota ze SUBVI *hex\_na\_uint16.vi* a vstupem úhlu natočení je vypočítaný úhel pro daný LiDAR  $\varphi_{Lx}$ . Po přepočítání jsou hodnoty navýšeny o korekční hodnoty vystupující ze SUBVI *vyrovnavani\_skemu.vi*. Všechny hodnoty pak vystupují z vnitřní smyčky a indexují se na pole.

Ve vnější smyčce jsou vytvořeny 3 *Cluster* – první z polí neupravených nepřevedených hodnot ve formátu *Uint16*, které vystupovali ze SUBVI *hex\_na\_uint16.vi* pro oba LiDARy. Tento *Cluster* vystupuje z vnější smyčky a indexuje se na pole *Clusterů* a je předán posuvnému registru dat obou LiDARů. Další dva *Cluster* jsou vytvořeny ze souřadnic *x* a *y* pro každý LiDAR zvlášť. Pomocí funkce *Build Array* je z nich vytvořeno pole, které se pak zobrazuje každých 300 ms ve 2D grafu pro měřené roviny. Každý *Cluster* zvlášť vystupuje z vnější smyčky a indexuje se na pole *Clusterů* a obě pole pak vstupují do SUBVI *filtr.vi*. Do tohoto SUBVI vstupuje ještě *Cluster* parametrů měření a *Cluster* hodnot vyčtených z dat JSON ze SUBVI *filtrrovani\_s\_JSON.vi*. Výstup ze SUBVI je předán odpovídajícím posuvným registrům (*indexy vozidla, nejistota výšky, rozměry a nejistoty*).

Z vnější smyčky vystupují také jednotlivé pole souřadnic obou LiDARů. Společně s hodnotami ujeté vzdálenosti pro aktuální čas se při výstupu z pole indexují na 2D pole a společně vstupují do SUBVI *plot\_helper.vi*. Výstup z tohoto SUBVI se zobrazuje na 3D grafu.



### 6.3.3 Stav výpočtu rozměrů (*Dimensions*)

Předem byl tento stav určen k výpočtu jednotlivých rozměrů vozidla. Později byly výpočty šířky a výšky přesunuty, aby nebylo nutné opakovat stavy. V tomto stavu tedy probíhá výpočet délky vozidla z filtrovaných dat a dat *JSON*. Dále v tomto stavu se pomocí funkce *Unbundle by name* rozbali *Cluster Rozměry a nejistoty*, zapíše se do něj délka a pak se všechny rozměry zobrazí na ovládací panel. Stejně tak se zobrazí zjištěné rozměry vozidla podle firmy Cross.

### 6.3.4 Stav výpočtu nejistot (*Uncertainties*)

Pro výpočet všech nejistot typu B, kombinovaných i rozšířených nejistot je určen tento stav. Kromě výpočtů a zobrazování hodnot nejistot také předává data mezi posuvnými registry, aby byly dostupné i v ostatních stavech.

Jako první nejistota je vypočítána nejistota dána systematickou chybou  $UB_1(x)$  podle rovnice 5.2. Stejným způsobem je vypočítána nejistota statistické chyby  $UB_2(x)$  dle rovnice 5.4. Maximální odchylky jsou brány z *Clusteru* parametrů měření. Další parametr, který je z tohoto *Clusteru* brán je *Frekvence snímání*, která vstupuje do SUBVI nejistoty měření délky  $UB_3(x)$  a nejistoty zaokrouhlení  $UB_5(x)$ . Z *Clusteru* parametrů měření je také brána hodnota *Úhlové rozlišení* ve stupních, kterou potřebuje jako vstup SUBVI nejistoty měření šířky.

Výsledné hodnoty nejistot typu B jsou vypočítány podle rovnic 5.19 (pro délku  $l$ ), 5.20 (pro šířku  $w$ ) a 5.21 (pro výšku  $h$ ). Vzhledem k tomu, že program nepočítá s žádnými nejistotami typu A, jsou výsledné nejistoty typu B zároveň kombinovanými nejistotami. Vynásobením konstantou rozšíření  $k = 2$  jsou získány rozšířené nejistoty daných rozměrů. Ty jsou pak zobrazeny na Ovládacím panelu. Společně s nimi jsou zobrazeny i nejistoty zjištěné firmou Cross. Před zobrazením je nutné je opět vynásobit konstantou rozšíření  $k$ . Programem vypočítané nejistoty jsou uloženy do *Clusteru*, která je předán posuvnému registru. Pak už je jen zvolen další stav (*Wait for Event*).

### 6.3.5 Klidový stav (*Wait for Event*)

Jak je tomu i v ostatních stavech, i klidový stav předává data mezi posuvnými registry. Dalším úkolem klidového stavu je kontrolovat akce uživatele, a to s co nejmenším dopadem na výkon. Toho se docílí použitím *Event* struktury. Ta, pokud je aktivní, nechává počítač vykonávat jinou užitečnou činnost a reaguje až když uživatel vykoná definovanou činnost.

V případě tohoto programu *Event* struktura obsluhuje stisk tlačítka *Start*, *Exit* nebo *Open*. *Exit* tlačítko změní stav na stav ukončení. Tlačítko *Open* vyvolá nabídku pro výběr nového souboru a následně zapne stav inicializace, pokud je přepínač v poloze vlastní data. Jinak vybere výchozí data a opět zapne stav inicializace. Tlačítko *Start* uvede program do stavu zobrazení dat v případě, že bylo vybráno ID z nabízených možností. Pokud ne, bude program ukončen přechodem do stavu *Exit* s vypsáním chybové hlášky.

### 6.3.6 Stav ukončení (*Exit*)

V tomto stavu se vykonává bezpečné ukončení. Všechna data jsou předána mezi posuvnými registry pro možnou kontrolu i po vypnutí programu. Samotné ukončení je pak realizováno splnění podmínky pro ukončení *While* smyčky vpravo dole na blokovém diagramu.

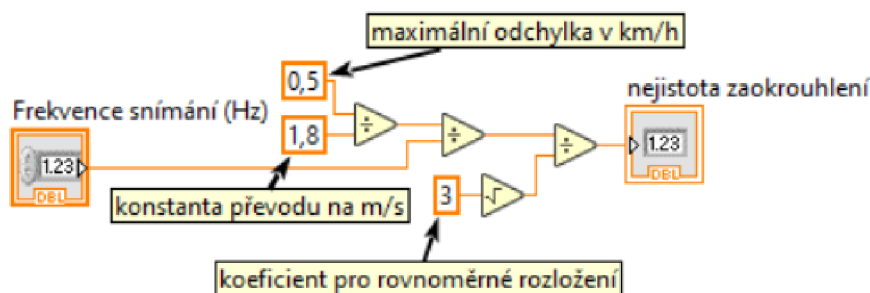
### 6.3.7 Podprogramy (SUBVI) stavového automatu

Vzhledem k tomu, že v LabVIEW nelze využívat funkci zvětšování a zmenšování zobrazení v blokovém diagramu, je důležité pro zachování přehlednosti často přesunovat části kódu do podprogramů, kterým se v LabVIEW říká *SUBVI*. Výsledný hlavní program by pak neměl zabírat víc částí, než je schopna zobrazit obrazovka bez použití posuvníků. Z tohoto důvodu bylo vytvořeno pro program 15 *SUBVI*, které budou dále vysvětleny:

- Nejistota zaokrouhlení
- Nejistota měření délky
- Nejistota měření šířky
- Výpočet  $D_{max}$  pro nejistotu šířky
- Měření délky
- Hex na uint16
- Plot helper
- Třídění na LiDARy
- Filtr
- Filtrování s JSON
- Vyrovnání roviny
- Vyrovnání skenů
- Polární do kartézských souřadnic
- Výběr vozidla
- Parametry For smyčky

#### 6.3.7.1 Nejistota zaokrouhlení

SUBVI obsahuje rovnici pro výpočet nejistoty typu B dané zaokrouhlením. Rovnice je k nalezení v podkapitole 5.3.7 Zaokrouhlování rychlosti. Vstupem je hodnota frekvence snímání a výstupem je hodnota nejistoty zaokrouhlení. Vstup i výstup je typu *Double*. Blokový diagram tohoto SUBVI je na obrázku 6.3.



Obrázek 6.3: Blokový diagram SUBVI *Nejistota zaokrouhlení*

### 6.3.7.2 Nejistota měření délky

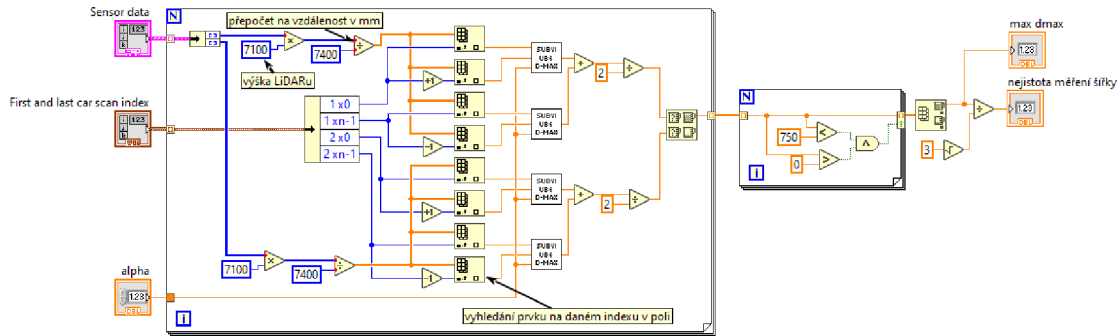
Vstupem do tohoto SUBVI je vstupní 2D pole *String* hodnot a *Double* hodnota frekvence snímání. Ze vstupního pole je vybrána pomocí funkce *Index Array* rychlost vozidla a funkcí *Decimal String To Number* převedena na *Int32*. Rovnice tohoto skriptu je uvedena v podkapitole 5.3.5 Délka vozidla.

### 6.3.7.3 Nejistota měření šířky

Jak název napovídá, v tomto SUBVI se počítá nejistota měření šířky z výsledku SUBVI *Výpočet  $D_{max}$  pro nejistotu šířky*. Vstupem jsou data senzorů (pole struktur dvou polí *UInt16* hodnot), indexy vozidla (pole struktur 4 *Int32* prvků) a hodnota úhlového rozlišení. Hlavní částí je *For* smyčka, ve které se zpracovávají postupně data prvních dvou zmíněných vstupů. Podle velikosti těchto polí se řídí i počet iterací smyčky. Data senzorů jsou rozděleny na 2 pole jednotlivých senzorů funkcí *Unbundle*. Dále jsou přepočteny na reálnou délku v milimetrech (výška LiDARu je 7,1 m). Funkcí *Unbundle By Name* je rozdělen *Cluster* pole indexů na 4 indexy (první a poslední prvek, který detekoval vozidlo pro oba LiDARy). Z dat senzorů je pomocí funkce *Index Array* na indexech z *Clusteru* vyčtena vzdálenost těchto bodů od LiDARů. Tyto hodnoty pak jsou vstupy do SUBVI *Výpočet  $D_{max}$  pro nejistotu šířky* společně s hodnotou úhlového rozlišení.

Po Výpočtech  $D_{max}$  jsou výsledné hodnoty pro jednotlivé LiDARy pro obě strany vozidla sečteny a vyděleny dvěma a vybrána větší hodnota z nich. Tato hodnota vystupuje z *For* smyčky a indexuje se do pole. To vstupuje do další *For* smyčky s podmínkou, která propouští hodnoty pouze v intervalu (0; 750), což vyfiltruje vysoké hodnoty způsobené nedetekováním druhého bodu na vozidle. Hodnota 750 byla vybrána jako přibližná hodnota této hranice filtrace vzhledem k datům, které filtruje, a je možné ji beze změny výstupu mírně upravit. Ty jsou způsobeny tím, že se paprsek nevrátí do LiDARu z důvodu odražení od oken vozidla. Z výsledného pole, které smyčka propustila, je vybráno maximum (výsledné  $D_{max}$ ) a vyděleno odmocninou ze 3 (výsledná nejistota typu

B pro měření šířky). Jediným výstupem je výsledná nejistota typu B. Náhled na blokový diagram tohoto SUBVI je na obrázku 6.4.



Obrázek 6.4: Blokový diagram SUBVI *Nejistota měření šířky*

### 6.3.7.4 Výpočet $D_{max}$ pro nejistotu šířky

Obsahem SUBVI *Výpočet  $D_{max}$  pro nejistotu šířky* je kód pro rovnici 5.13 z podkapitoly 5.3.8 Šířka vozidla. Vstupem jsou 3 hodnoty *Double* –  $b$  (poslední paprsek odražený od vozidla),  $c$  (předposlední paprsek odražený od vozidla) a úhel  $\alpha$  (úhlové rozlišení) ve stupních. Výstupem je *Double* hodnota  $D_{max}$  pro danou stranu vozidla.

### 6.3.7.5 Měření délky

Do tohoto SUBVI vstupují 3 parametry: *Cluster* dat vyčtených z formátu *JSON*, filtrovaná data a parametry LiDARu. Ze zjištěných dat *JSON* se vyčte pole hodnot *Center mean*. Tyto hodnoty obsahují nalezené souřadnice středu vozidla, ale pro účel této části kódu se využívají jako kontrola, zda sken našel vozidlo nebo ne. Hodnota *Center mean* ve formátu *String* se tedy porovná s řetězcem *null*, pokud se tyto řetězce nerovnájí, správná hodnota je propuštěna z *For* cyklu. Pomocí bloku *Array size* je zjištěna velikost výsledného pole a tím pádem i počtu skenů, které zachytily vozidlo. Při inkrementaci o 1 je přidán jeden další sken, který vozidlo „posune“ doprostřed mezi skeny, které ho nezachytili. Vozidlo se tedy nachází uprostřed intervalu všech možností jeho délky.

Počet skenů je potřeba ještě vynásobit vzdáleností mezi skeny (vypočítanou podle rovnice 6.2) a nakonec tisícem, aby byla uvedena hodnota v milimetrech. Blokový diagram SUBVI je na obrázku 6.5.

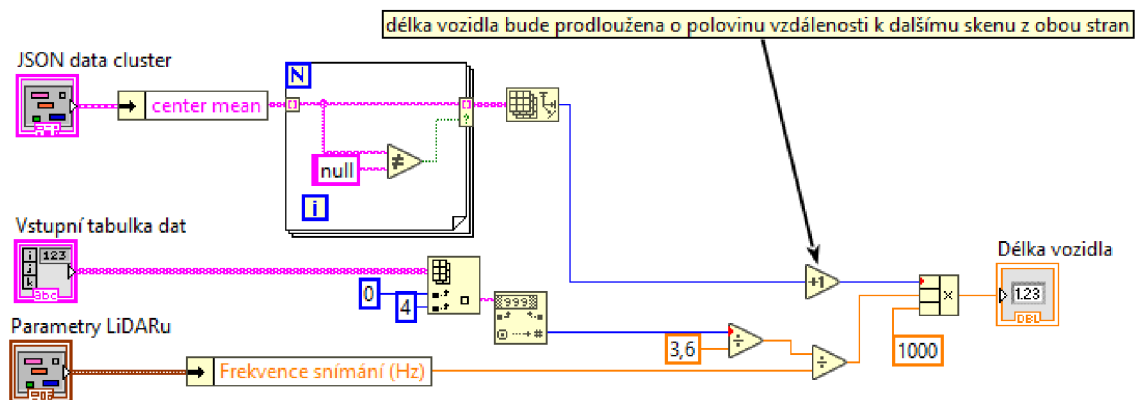
$$s = \frac{v_{mps}}{f} = \frac{v_{kph}}{3,6 f} \quad (6.2)$$

$s$ : délka vozidla [m]

$v_{mps}$ : rychlost vozidla [m/s]

$v_{kph}$ : rychlost vozidla [km/h]

$f$ : frekvence snímání LiDARu [Hz]

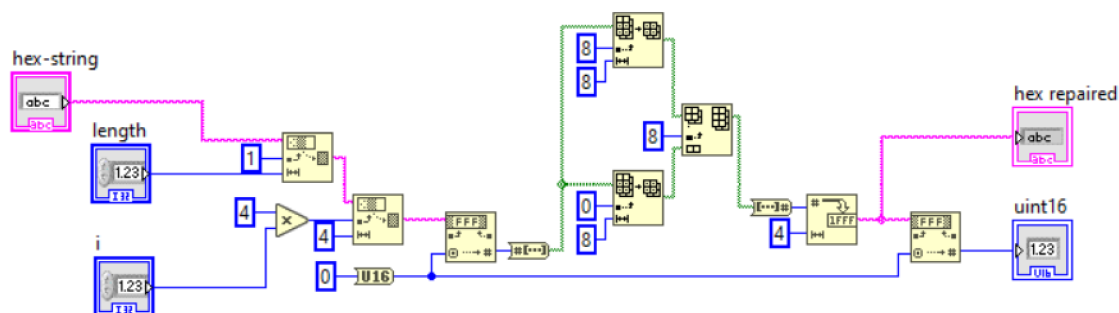


Obrázek 6.5: Blokový diagram SUBVI *Měření délky*

### 6.3.7.6 Hex na uint16

V tomto SUBVI dochází k převodu hexadecimálních znaků exportovaných z databáze na vzdálenosti mezi LiDAREm a naměřeným bodem ve formátu *Uint16*. Vstupem je řetězec hexadecimálních znaků, jeho délka (bez uvozovek) a iterace *For* smyčky, ve které SUBVI běží. Pomocí funkce *String Subset* je vybrána část řetězce bez uvozovek a následně vybrána část 4 hexadecimálních znaků, které dohromady tvoří jednu hodnotu vzdálenosti ve formátu *Uint16*. Jako počáteční index je brán čtyřnásobek aktuální iterace smyčky.

Samotný převod je pak realizován funkcí *Hexadecimal String To Number* s výstupním formátem zvoleným *Uint16* konstantou. Vzhledem k tomu, že databázová funkce *ENCODE* používá opačnou endianitu než LabVIEW, je nutné přehodit jednotlivé byty hodnoty *Uint16*. toho lze docílit více způsoby. Ve funkci je například zvolen způsob převodu na binární pole, následně je 16bitové pole rozděleno na dvě 8bitové pomocí funkce *Array Subset*, je přehozeno jejich pořadí a pomocí funkce *Insert Into Array* jsou pole opět spojeny a převedeny zpět přes hexadecimální znaky na hodnotu *Uint16*. Výstupem z funkce je hodnota *Uint16* a opravená část hexadecimálních znaků. Výsledný blokový diagram je pak na obrázku 6.6.



Obrázek 6.6: Blokový diagram SUBVI *Hex na uint16*

### 6.3.7.7 Plot helper

*Plot helper* SUBVI byl vytvořen z důvodu zjednodušení hlavního VI. Slouží k vytvoření 3D grafů pro oba LiDARy. Vstupem je pět 2D polí hodnot *Double*, první je 2D pole hodnot  $x$  pro první LiDAR, pole  $y$  pro první LiDAR, další dvě pole jsou hodnoty  $x$  a  $y$  pro druhý LiDAR a pak společné pole časové osy  $z$  pro oba LiDARy. Posledním vstupem je *Error cluster*. Pole prvního LiDARu vstupují do LabVIEW bloku *Surface plot helper.vi*, ale vstupují do jiných vstupů, než by bylo očekáváno. Je to z důvodu, že 3D graf nejde vhodně otočit. Pokud se tedy prohodí osy ( $x \rightarrow y$ ,  $y \rightarrow z$ ,  $z \rightarrow x$ ), bude graf otočen správně.

Propojením prvního bloku *Surface plot helper.vi* se stejným blokem, do kterého vstupují hodnoty druhého LiDARu, budou vytvořeny data pro oba LiDARy, které se dají zobrazit do jednoho grafu. Výstupem tohoto SUBVI je pak pole typu *3D plot lvclass* a *Error cluster*.

### 6.3.7.8 Třídění na LiDARy

Aby bylo možné později v programu počítat s daty jednotlivých LiDARů, je nutné je roztrždit. V SUBVI třídění na LiDARy je třídění realizováno *For* smyčkou s podmínkou. Vstup SUBVI je pouze 2D pole filtrovaných dat, ale do smyčky vstupuje pouze 10. (naměřená data) a 13. sloupec (ID LiDARu). Pokud je ID rovno jedné, pak mohou data vystupovat a indexovat první pole *sensor 1 data*. Pokud je ID větší než jedna, budou data indexovat druhé pole *sensor 2 data*. Pak je ještě vyčtena hodnota ID pruhu silnice z 4. sloupce a prvního řádku. Tato hodnota je převedena číslo typu *Int32* do výstupu *lane*. Pole *sensor 1 data* a *sensor 2 data* jsou také výstupy ze SUBVI, ale jsou to pole hodnot ve formátu *String*.

### 6.3.7.9 Filtr

SUBVI *filtr.vi* se nachází ve stavu pro zobrazování dat a je určeno k výpočtu šířky, výšky, nejistoty měření výšky a hledání krajních indexů pole skenu vozidla. Vstupuje zde *Cluster* dat JSON, naměřená data obou senzorů a *Cluster* hodnot rozměrů a nejistot. Také je zde vytvořen *Cluster* krajních indexů pole skenu vozidla, který po výpočtech výstupem SUBVI.

Hlavní částí tohoto SUBVI jsou 2 identické, po sobě se vykonávající *For* smyčky. Vstupem do první smyčky jsou naměřená data prvního LiDARu a vytvořený *Cluster* pro indexy. Ten po výpočtech vystupuje z první smyčky a vstupuje do druhé, což udává jejich pořadí vykonávání. Do druhé smyčky pak vstupují ještě naměřená data druhého LiDARu.

První akcí ve smyčce je rozdělení vstupního pole *Clusterů* polí hodnot  $x$  a  $y$  po daný LiDAR. Blokem *Unbundle* je každý *Cluster* rozdělen a pole  $x$  a  $y$  vstupují do 1. vnitřní *For* smyčky. Zde je kontrolováno, zda je hodnota  $x$  v intervalu  $\langle 500; 3500 \rangle$  (v tomto intervalu hodnot  $x$  se bude na skenu nacházet vozidlo). Pokud ano, je hodnota  $x$ ,  $y$  a aktuální iterace *i<sub>loop1</sub>* propuštěna z pole. Pole těchto vybraných hodnot  $x$  a  $y$  vstupuje

do 2. vnitřní smyčky. Uvnitř je kontrolováno, zda je hodnota  $y$  větší než 500 (jakákoliv hodnota vyšší než 50 cm na daném intervalu hodnot  $x$  bude znamenat nalezený bod na vozidle). Pokud ano, opět vystupují vybrané hodnoty  $x$ ,  $y$  a  $i_{loop2}$ .

Zmíněné pole indexů z 1. a 2. vnitřní *For* smyčky  $i_{loop1}$  a  $i_{loop2}$  jsou použity pro výpočet krajních indexů polí skenu vozidla. Pokud je sečtena první hodnota pole indexů  $i_{loop1}$  s první hodnotou pole indexů  $i_{loop2}$ , vznikne index původního pole naměřených hodnot, na kterém se v poli nachází první hodnota nalezená na vozidle (v *Clusteru* označená jako  $a \times 0$ , kde  $a$  je ID LiDARu). Pokud se pak vypočítá délka pole indexů  $i_{loop2}$  a odečte se 1 (LabVIEW indexuje od nuly) a zjistí se tak hodnota na posledním indexu pole a ta se sečte s první hodnotou pole indexů  $i_{loop1}$ , bude zjištěn index posledního bodu nalezeného na vozidle (označeného  $a \times n - 1$ , kde  $a$  je opět ID LiDARu).

Výstupní pole hodnot  $x$  z druhé vnitřní *For* smyčky je použito pro výpočet šířky. Z pole je zjištěna první a poslední hodnota  $x$  a po jejich vzájemném odečtení v absolutní hodnotě je zjištěna šířka vozidla pro daný sken. Tato hodnota vystupuje z vnější *For* smyčky a indexuje se do pole.

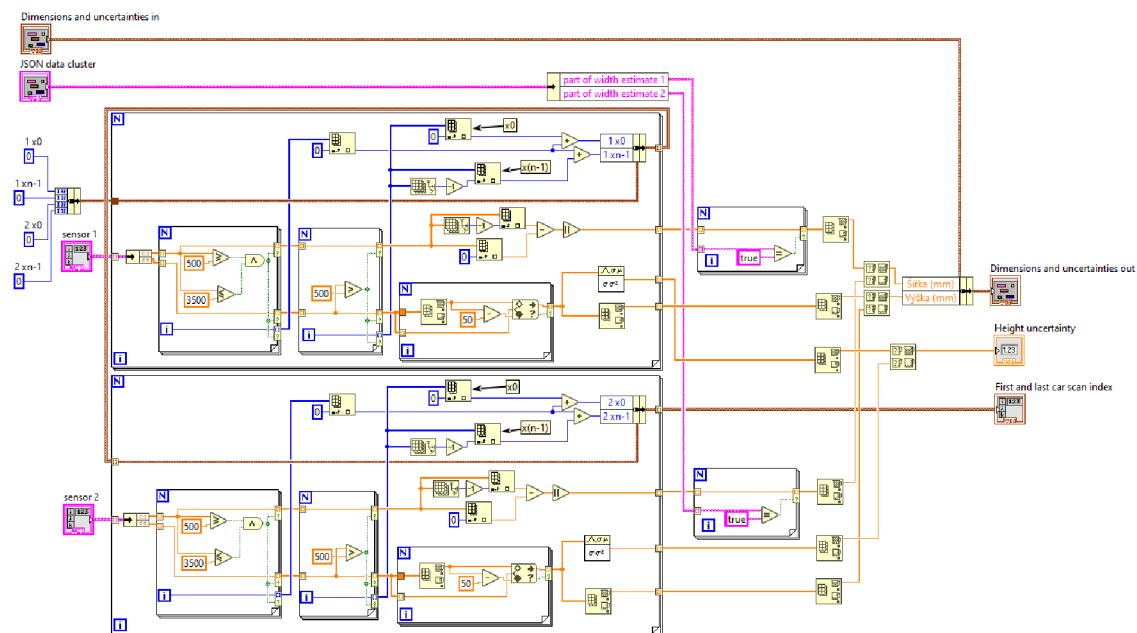
Z druhé vnitřní *For* smyčky vystupuje také pole hodnot  $y$ , které vstupuje do třetí vnitřní *For* smyčky. Zde je zjištěna jeho maximální hodnota a pomocí funkce *In Range and Coerce* je vytvořen interval  $(\max(y) - 50; \max(y))$ , který je výstupní podmínkou pro hodnoty pole této smyčky. Toto výstupní pole je vstupem do funkce *Array Max & Min*, kde je zjištěno opět jeho maximum (hodnot daného skenu) a vystupuje z hlavní smyčky, kde se indexuje na pole maximálních hodnot výšek jednotlivých skenů.

Výstupní pole z třetí vnitřní smyčky je použito i jako vstup do funkce *Std Deviation and Variance*. Touto funkcí je spočítána směrodatná odchylka hodnot  $y$  v intervalu  $(\max(y) - 50; \max(y))$ . Směrodatná odchylka pak vystupuje z hlavní smyčky a indexuje se na pole směrodatných odchylek jednotlivých skenů.

Pole hodnot vypočítaných šířek je bráno jako vstup do další *For* smyčky zároveň s hodnotami z JSON dat *part of width estimate a* ( $a$  je opět ID LiDARu). Ve smyčce je kontrolováno, že sken byl použit pro výpočet šířky (*part of width estimate a = true*). Pokud ano, bude vypočítaná hodnota propuštěna ze smyčky a vytvoří pole. Maximální hodnota tohoto pole je pak vypočítaná šířka vozidla daným LiDAREm. Stejná smyčka proběhne i pro druhý LiDAR a na výsledná šířka je dána maximem z těchto dvou hodnot. Vstupuje do *Clusteru* rozměrů a nejistot, který je výstupem ze SUBVI.

Z výstupních polí výšek a směrodatných odchylek je pro každý LiDAR spočítáno maximum a pak je vybrána větší hodnota z maxim pro oba LiDARy. Tímto způsobem je vypočítána výška vozidla, která je na závěr zapsána do výstupního *Clusteru Dimensions and uncertainties out*. Maximální směrodatná odchylka je pak ještě vydělena dvěma (normální rozdělení) a zapsána na výstup ze SUBVI jako *Height uncertainty*.

Celý tento proces je zobrazen na obrázku 6.7.



Obrázek 6.7: Blokový diagram SUBVI *Filtr*

### 6.3.7.10 Filtrování s JSON

Toto SUBVI vyžaduje mít nainstalovaný doplněk *JSONtext* pro LabVIEW. Odkaz na instalaci je přímo v LabVIEW v nabídce bloků pro blokový diagram v kategorii *Programming / String / Flatten/Unflatten String*. Vstupem je řetězec znaků *vehicle ID* a *Error Cluster*.

Pomocí funkce *Application Directory* je vybrána cesta ke složce s projektem a za pomoci funkcí *Build Path* je upřesněna cesta na složku *data* a je vybrán textový soubor s vybraným ID vozidla. Tato data kvůli své délce není možné číst z exportovaného souboru z databáze, protože ten náhodně zalomuje příliš dlouhé řádky. Dat v tomto řetězci znaků je uloženo tak velké množství, že pokud program nekomunikuje přímo s databází, je nutné tento řetězec znaků uložit zvlášť do textového souboru.

Po vybrání přesné cesty k souboru je pomocí funkce *Open/Create/Replace File* otevřen tento soubor a přečten funkcí *Read from Text File*. Vzhledem k nevhodnému uložení dat ve formátu JSON je nutné řetězec upravit. Funkcí *String Subset* je vybrán celý řetězec mezi uvozovkami (prvním a posledním znakem jsou uvozovky, ale formát JSON uvozovkami ohraničen nemá být) a následně funkcí *Search and Replace String* je každá dvojice uvozovek (tj. “”) nahrazena pouze jedním znakem uvozovek (tj. “”).

V tomto stavu funkce doplňku *JSONtext* můžou s řetězcem pracovat. Je tedy možné použít jeho funkce – v tomto případě budou potřeba funkce *Find Multiple Items* a *Find Item*.

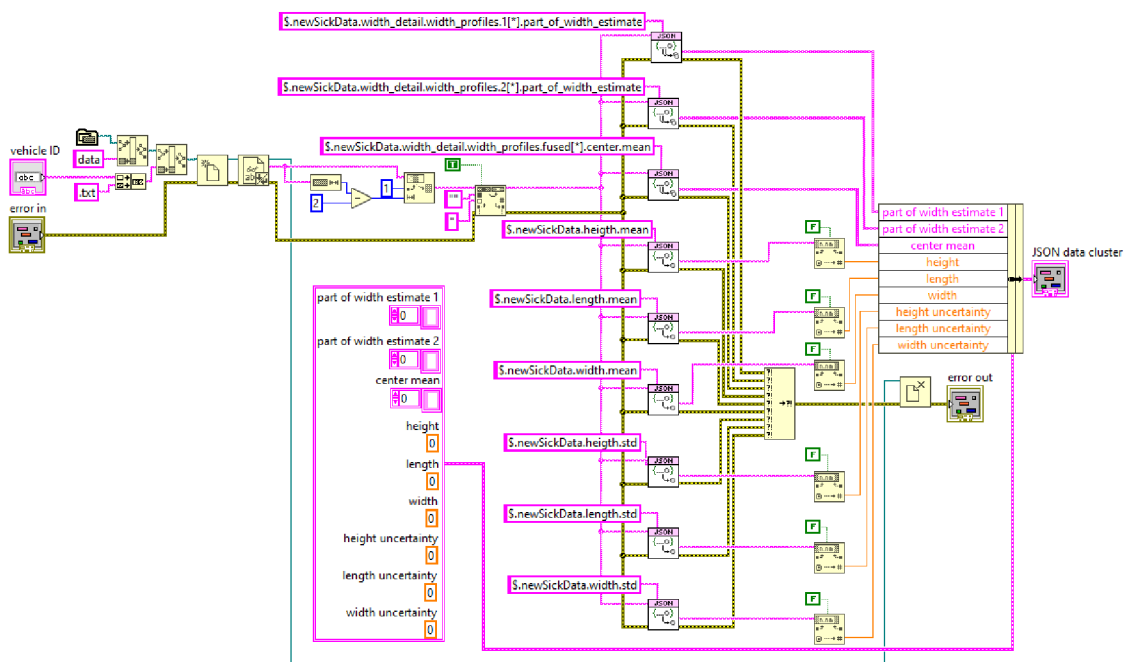
Data ve formátu JSON obsahují mimo jiné informace o každém skenu vozidla daného ID. Tyto data se tedy v řetězci opakují (s různými hodnotami v závislosti na skenu). Pro



vyčtení těchto dat je nutné použít funkci *Find Multiple Items*. Daty jsou hodnoty *part\_of\_width\_estimate* pro oba LiDARy (binární hodnota rozhodující o tom, zda byl sken použit k výpočtu šířky) a hodnoty *center.mean*, které jsou v programu použity pro zjištění, zda se paprsky odrážely od vozidla nebo jen od vozovky. Pro neopakující se data v řetězci je použita funkce *Find Item*. Těmito daty jsou hodnoty rozměrů vozidel (*height.mean*, *length.mean*, *width.mean*) a hodnoty kombinovaných nejistot (*height.std*, *length.std*, *width.std*) zjištěných firmou Cross. Tyto hodnoty jsou následně převedeny z formátu *String* na formát *Double* a budou zobrazovány na ovládacím panelu pro porovnání s hodnotami zjištěnými výpočty tímto programem.

Všechny hodnoty jsou nakonec uloženy do *Clusteru* pomocí funkce *Bundle By Name* inicializovaným konstantním *Clusterem* s konstantami stejného formátu, jako budou dané hodnoty, a se stejnými názvy.

Všemi funkcemi tohoto SUBVI prochází *Error Cluster*, takže jakákoliv chyba vyvolaná jednou z těchto funkcí povede k bezpečnému ukončení programu a vyvolání chybové hlášky. Na obrázku 6.8 je pak zobrazeno blokové schéma tohoto SUBVI.



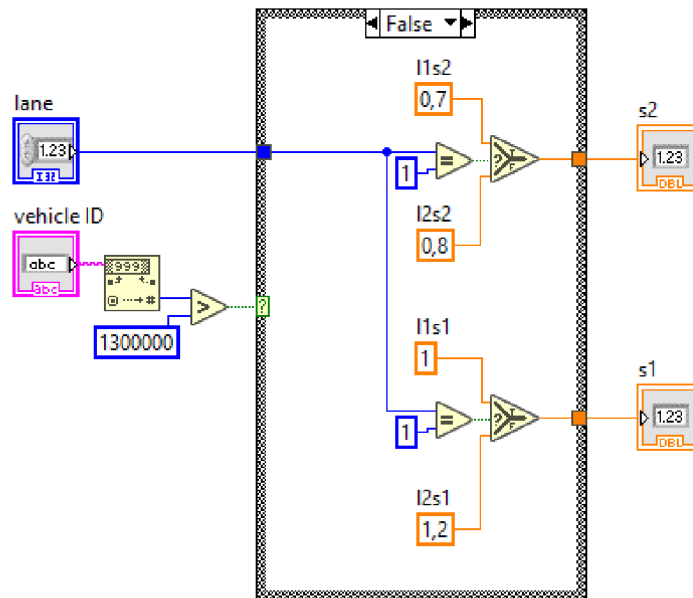
Obrázek 6.8: Blokový diagram SUBVI *Filtrování s JSON*

### 6.3.7.11 Vyrovnání roviny

SUBVI *vyrovnani\_roviny.vi* slouží ke korekci natočení roviny vozovky. Vstupem je ID pruhu silnice a ID vozidla. ID vozidla je z formátu *String* převedeno funkcí *Decimal String To Number* na *Int32* a je porovnáno s hodnotou 1300000, protože pokud ID vozidla přesáhne tuto hodnotu na tomto měřicím stanovišti v daný časový interval, znamená to,

že vozidlo jede v opačném směru oproti vozidlům s ID menší než 1300000. Výsledek porovnání rozhoduje o stavu *Case* struktury.

V obou stavech je porovnáváno ID pruhu s číslem 1 a blokem *Select* je vybírána vhodná konstanta natočení ve stupních pro oba LiDARy. V obou stavech probíhá tento výběr, rozdílné jsou pouze konstanty natočení. Hodnoty natočení vystupují z *Case* struktury a tvoří výstupy ze SUBVI. Náhled na blokové schéma je na obrázku 6.9.



Obrázek 6.9: Blokový diagram SUBVI *Vyrovnaní roviny*

### 6.3.7.12 Vyrovnaní skenů

Vyrovnaní skenů probíhá stejně jako vyrovnaní roviny. Vstupem do SUBVI je opět ID pruhu silnice a ID vozidla, které je převedeno z formátu *String* na *Int32* a porovnáváno s hodnotou 1300000. Binární hodnota výstupu porovnání rozhoduje o stavu *Case* struktury. Pokud je hodnota větší než 1300000, nastává stav *True* v *Case* struktuře a vybírají se hodnoty jeden směr, v opačném případě to bude opačný směr.

V *Case* struktuře se vybírají blokem *Select* hodnoty posunutí po ose *x* a *y* pro oba pruhy silnice. Každý stav tedy obsahuje 8 konstant uvedených v milimetrech. Výstupem z *Case* struktury jsou 4 hodnoty, které jsou zároveň výstupy ze SUBVI – posunutí *x* pro pruh 1 a pruh 2, posunutí *y* pro pruh 1 a pruh 2.

### 6.3.7.13 Polární do kartézských souřadnic

Naměřené hodnoty LiDARem jsou po vyčtení z databáze a úpravě na formát *Uint16* uloženy v polárních souřadnicích. Aby je bylo možné zobrazit v grafu, je třeba převést tyto souřadnice na kartézské. Od toho slouží toto SUBVI, které realizuje rovnice pro tento

převod. Vstupem je délka  $r$  ve formátu *UInt16* mezi LiDARem a naměřeným bodem a úhel  $\varphi$  formátu *Double*. Rovnice výpočtu pak jsou následující:

$$x = r \cdot \cos \varphi \quad (6.3)$$

$$y = r \cdot \sin \varphi \quad (6.4)$$

#### 6.3.7.14 Výběr vozidla

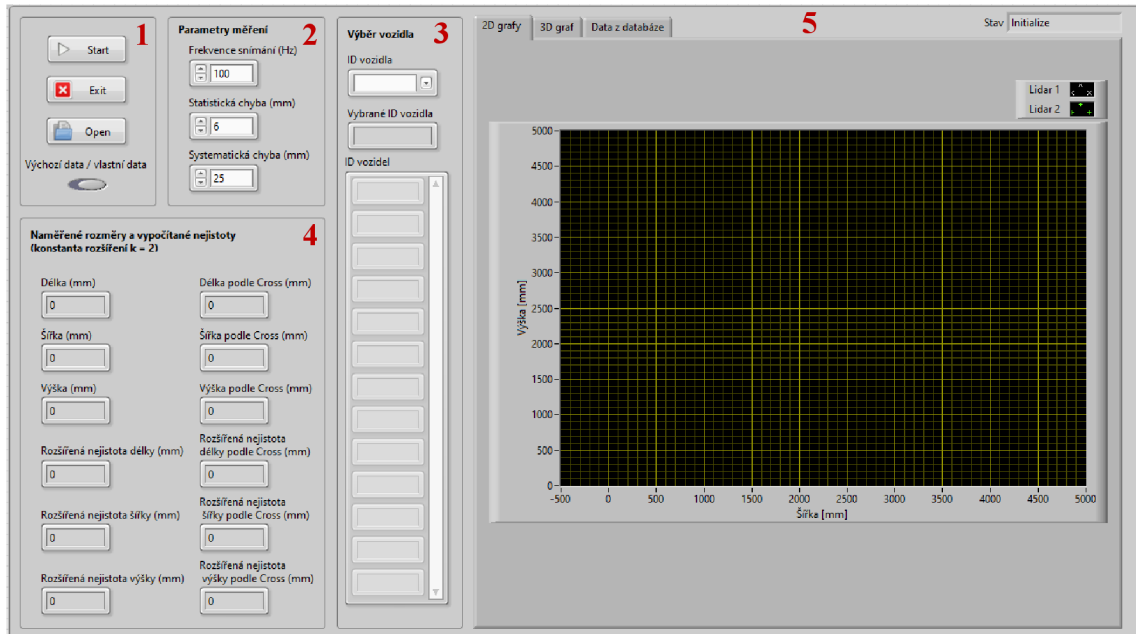
Výběr vozidla probíhá během stavu inicializace stavového automatu a má dva vstupy – 2D pole hodnot *String* nefiltrovaná data a *String* hodnota ID vozidla, kterou vybírá uživatel. Data vstupují do první *For* smyčky (počet cyklů určen indexováním nefiltrovaných dat) ve které pomocí funkce *Index Array* porovnáváme ID v 2D poli s vybraným ID. Pokud řádek podmínku splní, je propuštěn z cyklu. Výsledné pole indexuje další *For* smyčku, ve které se kontroluje správnost dat porovnáváním parametru *Scancount* a hodnoty 0 ve formátu *String*. Pokud je hodnota 0, je sken určen ke zjištění výšky vozovky před projetím vozidla. Pokud tedy řádek splňuje podmínku, že není *Scancount* roven nule, jsou data propuštěna z cyklu. Výsledné 2D pole hodnot typu *String*, které obsahují pouze záznamy vozidla vybraného ID, je výstupem SUBVI.

#### 6.3.7.15 Parametry For smyčky

Zde vstupují naměřené hodnoty obou LiDARů jako řetězce hexadecimálních znaků. Funkcí *String Length* je zjištěna délka těchto řetězců a z těchto délek jsou odečteny pro oba řetězce dva znaky pro uvozovky. Dále jsou hodnoty vyděleny čtyřmi a zaokrouhleny dolů (směrem k záporným hodnotám). Dále jsou porovnány a funkcí *Select* je vybrána větší hodnota, která byla dělena čtyřmi reprezentující počet iterací smyčky. Další funkcí *Select* je stejnou podmínkou vybrán delší řetězec (délka řetězce bez uvozovek). Výstupem ze SUBVI je tedy celé číslo typu *Double* reprezentující počet iterací a druhá hodnota typu *Int32* reprezentující délku řetězce dat.

## 6.4 Ovládací panel v LabVIEW

Na obrázku 6.10 je možné vidět ovládací panel vytvořeného programu. Ten je rozdělen na několik částí, které budou nyní popsány.



Obrázek 6.10: Náhled na ovládací panel vytvořeného programu

V části 1 (vlevo nahoře) jsou hlavní ovládací prvky programu – tlačítka *Start*, *Exit*, *Open* a přepínač výchozích a vlastních dat.

Ve vedlejší části (2) *Parametry měření* může uživatel změnit hodnoty frekvence snímání LiDARu a chyb měření, které jsou dostupné v datovém listu pro daný LiDAR. Výchozí hodnoty jsou nastaveny pro LiDAR Sick LMS511.

Další částí (3) je *Výběr vozidla*. Blok *ID vozidla* je rozbalovací nabídka jednotlivých ID vozidel, které jsou v daném vstupním souboru. Po vybrání se dané ID zobrazí v dalším bloku *Vybrané ID vozidla*. Pod tímto blokem je zobrazeno pole se všemi záznamy vozidel daného souboru.

Vlevo dole je část 4, která zobrazuje hodnoty rozměrů vozidla a nejistoty tohoto měření, a to jak hodnoty vytvořeného programu, tak hodnoty spočítané algoritmem firmy Cross. Všechny hodnoty jsou uvedeny v milimetrech a rozšířené nejistoty jsou spočítány s konstantou rozšíření  $k = 2$ .

Poslední část ovládacího panelu (5) je rozdělena na 3 záložky. První záložka zobrazuje jednotlivé naměřené body v daných rovinách, další záložka zobrazuje 3D graf celého vozidla a poslední záložka obsahuje 2D pole s daty vybraného souboru. V pravém horním rohu pak ovládací panel zobrazuje ještě aktuální stav programu.

## 6.5 Ovládání programu za běhu

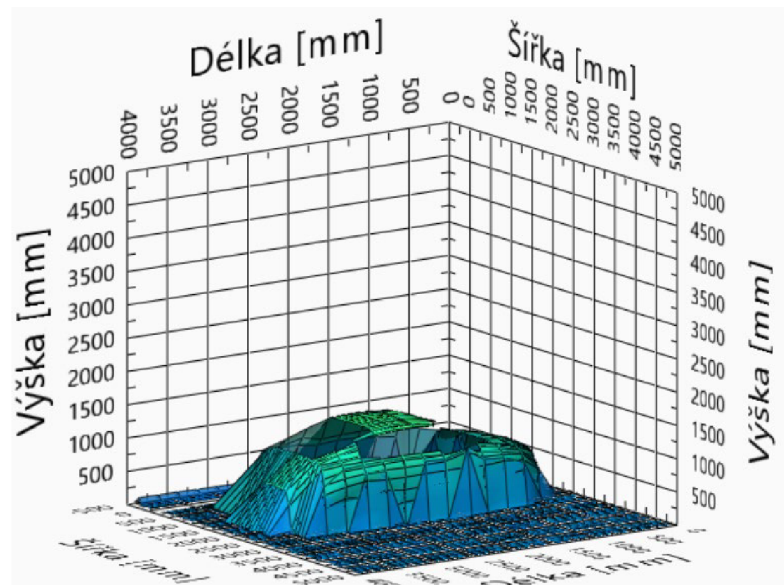
Při spuštění programu se zobrazí okno s výběrem souboru. Data určená pro zpracování programem jsou uložena ve složce projektu v podsložce *Data*. Název je pak „*data-(vehicle)*“, kde (*vehicle*) je model vozidla. Po vybrání vhodného souboru zmizí okno výběru souboru a aktualizuje se pole hodnot *ID vozidel*, které bude obsahovat všechny ID vozidel vybraného modelu. Stav bude v této chvíli zobrazovat *Wait for Event*, což znamená, že program čeká na akci uživatele.

Původní soubor dat je stále možné změnit. Pokud je to nutné, stačí, aby byl přepínač v poloze *vlastní data* a uživatel stiskne tlačítko *Open*. To vyvolá okno pro výběr souboru. Pokud by byl přepínač v poloze *výchozí data* a bylo by stisknuto tlačítko *Open*, byl by automaticky vybrán soubor „*data-Octavia\_IV\_sedan.csv*“. Po vybrání nového souboru proběhne krátká inicializace a program pak zůstane opět ve stavu *Wait for Event*.

Aby bylo možné zobrazit data a vypočítat hodnoty rozměrů a nejistot, musí uživatel vybrat ID v rozbalovací nabídce *ID vozidla*. Možnosti jsou zobrazeny také v poli *ID vozidel*. Zobrazování a výpočty se spustí po stisku tlačítka *Start*.

Po spuštění zobrazování a výpočtů bude program postupně po 300 ms zobrazovat jednotlivé měřicí roviny LiDARů na grafu v záložce *2D grafy*. Po automatickém zobrazení všech rovin zůstane na 2D grafu zobrazena poslední měřicí rovina. Na další záložce je možné zobrazit 3D graf naměřeného vozidla, jak ukazuje obrázek 6.11 pro vozidlo Škoda Octavia IV sedan s ID 1582094. V další záložce jsou k nahlédnutí i vyfiltrovaná data použitá pro výpočty a zobrazení. Po zobrazení všech naměřených rovin dat se zobrazí i vypočtené hodnoty rozměrů vozidla a nejistoty tohoto měření.

Po zobrazení dat a výpočtu všech hodnot je program uveden opět do stavu *Wait for Event*. Zde se může uživatel opět rozhodnout, co má program dále dělat. Pro bezpečné ukončení pak stačí stisknout tlačítko *Exit*.



Obrázek 6.11: 3D graf naskenovaného vozidla s ID 1582094

## 7. ZHODNOCENÍ VÝSLEDKŮ

Pro první srovnání výpočtů budou použity vypočtené hodnoty teoretických výpočtů z podkapitoly 5.4, hodnoty vytvořeného programu a vypočtené hodnoty firmy Cross pro vozidlo Škoda Octavia IV sedan s ID 1582094. Reálné rozměry vozidla jsou  $l = 4689$  mm,  $w = 1829$  mm a  $h = 1477$  mm. Vypočítané hodnoty budou rozděleny do třech tabulek – tabulka 7.1 pro délku vozidla 7.2 pro šířku a 7.3 pro výšku. Pro každý rozměr bude uvedena jeho hodnota z teoretického výpočtu, vytvořeného programu a hodnota vypočítaná firmou Cross. Uvedena bude i absolutní chyba od reálného rozměru, vypočítaná rozšířená nejistota pro  $k = 2$  a také, zda se reálná hodnota nachází v intervalu daném rozšířenou nejistotou nebo ne.

Tabulka 7.1: Výsledné hodnoty pro délku vozidla Škoda Octavia IV sedan

	Teoretické výpočty	Výpočty vytvořeného programu	Výpočty firmy Cross
Délka [mm]	3420	3420	3420
Absolutní chyba délky [mm]	1269	1269	1269
Rozšířená nejistota délky [mm]	280	280	210
Výskyt v intervalu daném rozšířenou nejistotou [-]	NE	NE	NE

Tabulka 7.2: Výsledné hodnoty pro šířku vozidla Škoda Octavia IV sedan

	Teoretické výpočty	Výpočty vytvořeného programu	Výpočty firmy Cross
Šířka [mm]	1738	1740	1721
Absolutní chyba šířky [mm]	91	89	108
Rozšířená nejistota šířky [mm]	90	230	32
Výskyt v intervalu daném rozšířenou nejistotou [-]	NE	ANO	NE

Tabulka 7.3: Výsledné hodnoty pro výšku vozidla Škoda Octavia IV sedan

	Teoretické výpočty	Výpočty vytvořeného programu	Výpočty firmy Cross
Výška [mm]	1428	1428	1447
Absolutní chyba výšky [mm]	49	49	30
Rozšířená nejistota výšky [mm]	30	30	18
Výskyt v intervalu daném rozšířenou nejistotou [-]	NE	NE	NE

Jak již bylo zmíněno v kapitole 5.4, u vozidla s ID 1582094 vznikla pravděpodobně chyba při měření rychlosti, což vedlo ke špatnému určení délky vozidla všemi třemi metodami. Ani v jednom případě tedy reálná délka vozidla neleží v intervalu daném rozšířenou nejistotou délky, jak ukazuje tabulka č. 7.1. Tímto rozměrem se tedy nemá smysl zabývat.

Chyba, která vznikla špatně naměřenou rychlostí naštěstí neovlivňuje měření šířky a výšky. Šířka je pro 3 různé metody určena v tabulce č. 7.2. Vypočítané šířky se pohybují v intervalu dvou centimetrů. To samé ale nelze říct o rozšířených nejistotách. Nejistota z teoretických výpočtů je téměř 3 tak velká jako ta, která byla vypočítána firmou Cross, a hodnota nejistoty vypočítaná programem je téměř osmkrát větší než hodnota firmy Cross. Je tedy zřejmé, že postup firmy Cross pro výpočet dané nejistoty se od výpočtů této práce zásadně liší. Postup firmy Cross ale nebyl zjišťován, aby byl postup výpočtu práce vytvářen od základu jako nový, a ne jako vylepšení toho stávajícího. Vysoké hodnoty rozšířené nejistoty šířky tohoto programu nejsou tedy špatné, jen využívají jiný postup. A v případě vozidla s ID 1582094 dokonce vytvořila nejistota interval, který obsahuje i reálnou hodnotu šířky.

Určení výšky opět probíhalo podobně v případě všech tří metod, jak ukazuje tabulka 7.3. Liší se navzájem od sebe o necelé 2 centimetry. Bohužel ani v jenom případě ale nebyla určena rozšířená nejistota tak, aby zahrnula do intervalu i reálnou hodnotu výšky.

Pro lepší představu o funkčnosti programu bylo měření provedeno pro všechny záznamy o projetí vozidel Škoda Octavia III combi. Vypočítané hodnoty rozměrů, absolutních chyb od reálných rozměrů a rozšířených nejistot rozměrů byly zaznamenány do souboru programu Microsoft Office Excel a byly vyjádřeny průměrné hodnoty všech těchto hodnot. Bylo také zjištěno, jak často se reálná hodnota rozměru vozidla nacházela v intervalu určeném rozšířenou nejistotou pro  $k = 2$ . Výsledky byly zapsány do tabulek 7.4, 7.5 a 7.6. Uvažované rozměry vozidla jsou  $l = 4658$  mm,  $w = 1814$  mm

a  $h = 1466$  mm. Porovnání bude provedeno pouze mezi hodnotami vytvořeného programu a hodnotami vypočítanými firmou Cross.

Tabulka 7.4: Výsledné hodnoty pro délku vozidla Škoda Octavia III combi

	Výpočty vytvořeného programu	Výpočty firmy Cross
Průměrná délka [mm]	5200	5090
Průměrná absolutní chyba délky [mm]	550	440
Průměrná rozšířená nejistota délky [mm]	440	280
Výskyt v intervalu daném rozšířenou nejistotou [%]	25	18

Tabulka 7.5: Výsledné hodnoty pro šířku vozidla Škoda Octavia III combi

	Výpočty vytvořeného programu	Výpočty firmy Cross
Průměrná šířka [mm]	1680	1755
Průměrná absolutní chyba šířky [mm]	140	104
Průměrná rozšířená nejistota šířky [mm]	190	45
Výskyt v intervalu daném rozšířenou nejistotou [%]	79	4

Tabulka 7.6: Výsledné hodnoty pro výšku vozidla Škoda Octavia III combi

	Výpočty vytvořeného programu	Výpočty firmy Cross
Průměrná výška [mm]	1396	1419
Průměrná absolutní chyba výšky [mm]	70	47
Průměrná rozšířená nejistota výšky [mm]	33	18
Výskyt v intervalu daném rozšířenou nejistotou [%]	0	7

Jak je možné vidět v tabulce 7.4, vypočítané hodnoty délky jak v tomto programu, tak firmou Cross, mají přibližně 0,5 m absolutní chybu. Je to dáno zejména tím, že LiDAR



skenuje s frekvencí snímání pouze 100 Hz, což se viditelně projevuje i na vypočítané nejistotě jak programem, tak firmou Cross. Vytvořeným programem vypočítaná nejistota je znatelně větší než nejistota vypočítaná firmou Cross, ale zato tvoří interval, ve kterém se reálná hodnota délky vyskytuje častěji.

Vzhledem k aktuálnímu otočení LiDARu (měřicí rovina je kolmo ke směru jízdy vozidla i vozovce) není možné docílit lepších výsledků, protože 2D LiDARy s vyšší frekvencí snímání, než je 100 Hz se zatím nevyrábí. Možností zlepšení by mohlo být zakoupení 3D LiDARu nebo přidání další 2D LiDAR, který bude jinak natočený, takže zachytí délku vozidla více paprsky. Tyto návrhy ale můžou vést k vlastním problémům a rozhodně ke komplikovanějším výpočtům.

V tabulce 7.5 jsou vypočítané hodnoty pro šířku vozidla. Zde je průměrná absolutní chyba vypočítané délky v programu 140 mm. Vzhledem k vypočítané průměrné rozšířené nejistotě 190 mm je reálná hodnota šířky v intervalu daném nejistotou v 79 % případů.

Dominantní nejistotou je opět nejistota daná metodou měření šířky, ostatní jsou v porovnání s ní téměř zanedbatelné. Tuto nejistotu nejspíš počítá firma Cross dost odlišným způsobem. Jejich průměrná rozšířená nejistota je přibližně čtvrtinová v porovnání s nejistotou z programu.

Zde je možné zlepšení zejména ve zvýšení úhlového rozlišení LiDARu, které je zodpovědné za dominantní nejistotu. Z porovnávaných LiDARů v kapitole 4.1 jsou vhodnější na měření šířky LiDARy Sick LMS111, VanJee WLR-711 a Hokuyo UGM-50LAP. Ty ale zaostávají oproti Sick LMS511 v jiných oblastech, jak popisuje porovnání LiDARů.

Tabulka 7.6 vypovídá o tom, že hodnoty výšky jsou určeny přesně. Program vypočítal rozšířenou nejistotu výšky na 33 mm, firma Cross nejistotu stanovila ještě přesněji (18 mm). O to zajímavější je, že reálná hodnota výšky v těchto intervalech ve většině případů není. Možným řešením tohoto problému by mohlo být volba větší konstanty rozšíření. Zlepšení by mohlo být i například započítání nejistoty dané tlakem v pneumatikách, která způsobuje, že vozidlo je nižší, než udává výrobce.

## 8. ZÁVĚR

Tato diplomová práce se zabývala analýzou nejistot měření rozměrů vozidel LiDARem a byla vypracována pro firmu Cross Zlín, která se mimo jiné tímto měřením zabývá.

Pojem LiDAR byl v 2. kapitole vysvětlen včetně jeho historie, konstrukce, principů, na kterých může pracovat a zmíněny byly i příklady použití včetně ukázky, jak naměřená data vypadají.

V další kapitole byly popsány chyby měření, včetně druhů a vztahů pro jejich výpočet. Kapitola ale byla hlavně věnována vysvětlení pojmu nejistota měření. Byly vysvětleny typy nejistot, jejich význam, vztahy pro výpočet i pravidla pro správný zápis. Uvedeny byly i zdroje nejistot, se kterými je možno se při měření setkat.

Kapitola 4 byla věnována popisu měřicího stanoviště firmy Cross, což bylo zobrazeno i na schématu, které zobrazovalo veškerou instrumentaci typického stanoviště. Část byla věnována i srovnání LiDARů konkurenčních firem, nicméně se ukázalo, že výběr LiDARů značky Sick, které firma Cross používá, je nejvhodnější, nicméně nebyl brán ohled na cenu. V této kapitole byly dále popsány metody měření jednotlivých rozměrů LiDARem včetně vztahů pro výpočet těchto rozměrů z naměřených dat a názorných ukázek tohoto měření.

Pátá kapitola popisuje samotnou analýzu nejistot. Představena byla databáze naměřených a vypočtených dat firmy Cross a data, na kterých budou výpočty předvedeny. Zmíněny byly i výrobcem udávané pracovní podmínky LiDARu. Dále byly popsány jednotlivé nejistoty, které měření ovlivňují a pokud to bylo možné, byly doplněny i výpočty a schémata. Na závěr byly uvedeny výsledky měření pro vybrané vozidlo včetně rozšířené nejistoty pro všechny rozměry.

Další kapitola se věnuje podrobnému popisu vytvořeného algoritmu pro výpočet rozměrů a nejistot těchto rozměrů z dostupných naměřených dat. Vytvořený algoritmus je napsán v prostředí LabVIEW jako stavový automat. Postupně jsou popsány jednotlivé stavy a podprogramy celkového programu. Vysvětleno je i ovládání programu uživatelem přes ovládací panel.

V poslední kapitole jsou shrnuty dosažené výsledky na vhodné sadě dat a jsou srovnány s reálnými rozměry vozidla a výsledky dosaženými firmou Cross. Navrženy jsou i možné vylepšení měření.

V současném stavu práce splňuje zadání i doporučený rozsah pro diplomovou práci. Její zpracovávání mi poskytlo rozšíření obzorů v oblasti LiDARů a utvrdilo znalosti o nejistotách. Přínos pro mě mělo i vytváření programu v LabVIEW, ve kterém jsem si mohl vyzkoušet návrh a vytváření pokročilejších programů pro zpracování dat. Věřím, že určitý přínos bude mít tato práce i pro její čtenáře.

## 9. LITERATURA

- [1] GARGOUM, Suliman a Karim EL-BASYOUNY. Automated extraction of road features using LiDAR data: A review of LiDAR applications in transportation. *2017 4th International Conference on Transportation Information and Safety (ICTIS)* [online]. IEEE, 2017, **4**, 563-574 [cit. 2021-11-08]. ISBN 978-1-5386-0437-3. Dostupné z: doi:10.1109/ICTIS.2017.8047822
- [2] GREGERSEN, Eric. Lidar. In: *Encyclopedia Britannica* [online]. Chicago, IL: Britannica, 2016 [cit. 2021-12-23]. Dostupné z: <https://www.britannica.com/technology/lidar>
- [3] BLAKEMORE, Ein. Lasers are driving a revolution in archaeology. In: *National Geographic* [online]. Washington, D.C.: National Geographic, c2015-2022 [cit. 2022-05-12]. Dostupné z: <https://www.nationalgeographic.com/culture/article/lasers-lidar-driving-revolution-archaeology>
- [4] Historical Maps and Charts audio podcast. In: *National Ocean Service website* [online]. Silver Spring, MD: NOAA, 2021 [cit. 2021-11-08]. Dostupné z: <https://oceanservice.noaa.gov/facts/lidar.html>
- [5] WASSER, Leah A. The Basics of LiDAR: Light Detection and Ranging - Remote Sensing. In: *Neon science* [online]. Boulder, CO: NSF, 2020 [cit. 2021-11-08]. Dostupné z: <https://www.neonscience.org/resources/learning-hub/tutorials/lidar-basics>
- [6] RAJ SINGH, Amit. What are the Top 5 Uses of LiDAR. In: *Geospatial World* [online]. Geospatial World, 2021 [cit. 2021-12-23]. Dostupné z: <https://www.geospatialworld.net/blogs/top-5-uses-lidar-2/>
- [7] CHAKRAVARTY, Sudeep. Top 10 LiDAR Manufacturers in World. In: *Market Research Reports* [online]. Lewes, DE: Market Research Blog, c2010-2021 [cit. 2021-11-08]. Dostupné z: <https://www.marketresearchreports.com/blog/2019/04/22/top-10-lidar-manufacturers-world>
- [8] *Hexagon* [online]. Stockholm: Hexagon AB, 2022 [cit. 2022-05-12]. Dostupné z: <https://hexagon.com/>
- [9] *Luminar* [online]. Orlando, FL: Luminar Technologies, 2022 [cit. 2022-05-12]. Dostupné z: <https://www.luminartech.com/>

- [10] *Hokuyo USA* [online]. Indian Trail, NC: Hokuyo Automatic, c2016-2022 [cit. 2022-05-12]. Dostupné z: <https://hokuyo-usa.com/>
- [11] *Velodyne Lidar* [online]. San Jose, CA: Velodyne Lidar, 2022 [cit. 2022-05-12]. Dostupné z: <https://velodynelidar.com/>
- [12] *VanJee Technology* [online]. Beijing: VanJee Technology [cit. 2022-05-12]. Dostupné z: <http://www.vanjee.net/>
- [13] LIU, Jingyun, Qiao SUN, Zhe FAN a Yudong JIA. TOF Lidar Development in Autonomous Vehicle. *2018 IEEE 3rd Optoelectronics Global Conference (OGC)* [online]. IEEE, 2018, **3**, 185-190 [cit. 2021-11-08]. ISBN 978-1-5386-7399-7. Dostupné z: doi:10.1109/OGC.2018.8529992
- [14] A Guide to Lidar Wavelengths for Autonomous Vehicles and Driver Assistance. In: *Velodyne Lidar* [online]. San Jose, CA: Velodyne Lidar, 2021 [cit. 2021-12-23]. Dostupné z: <https://velodynelidar.com/blog/guide-to-lidar-wavelengths/>
- [15] TERNEUS, David. A Guide To LiDAR Wavelengths. In: *TopoDOT Blog* [online]. Orlando, FL: TopoDOT, 2021 [cit. 2021-12-23]. Dostupné z: <https://blog.topodot.com/a-guide-to-lidar-wavelengths/>
- [16] AKO, Thomas. Is LiDAR Dangerous For Our Eyes ?. In: *Photonics Report* [online]. Photonics Report, 2021 [cit. 2021-12-23]. Dostupné z: <https://photonicsreport.com/blog/is-lidar-dangerous-for-our-eyes/>
- [17] Technology in Focus: Bathymetric Lidar. In: *GIM International* [online]. Lemmer: GIM International, 2021 [cit. 2021-12-23]. Dostupné z: <https://www.gim-international.com/content/article/technology-in-focus-bathymetric-lidar-2>
- [18] A Brief Introduction to Time-of-Flight Sensing Part 2: Indirect ToF Sensors. In: *Terabee* [online]. Saint-Genis-Pouilly: Terabee, 2019 [cit. 2021-11-08]. Dostupné z: <https://www.terabee.com/a-brief-introduction-to-time-of-flight-sensing-part-2-indirect-tof-sensors/>
- [19] ŠEDIVÁ, Soňa. *Laboratorní cvičení č. 1 – Chyby přímých a nepřímých měření. Nejistoty měření: BMVE - Měření v elektrotechnice*. Brno, 2018.
- [20] *Nejistoty v měření I: vyjadřování nejistot* [online]. Děčín: Automa – časopis pro automatizační techniku, s. r. o., 2001, [cit. 2021-11-17]. Dostupné z: [http://automa.cz/Aton/FileRepository/pdf\\_articles/10960.pdf](http://automa.cz/Aton/FileRepository/pdf_articles/10960.pdf)
- [21] ŠEDIVÁ, Soňa. *Měření v elektrotechnice, Přednáška č. 2: Nejistoty měření, kompatibilita měření*. Brno, 2018.
- [22] ŠEDIVÁ, Soňa. *Měření v elektrotechnice, Přednáška č. 1: Základní pojmy, chyby měření*. Brno, 2018.

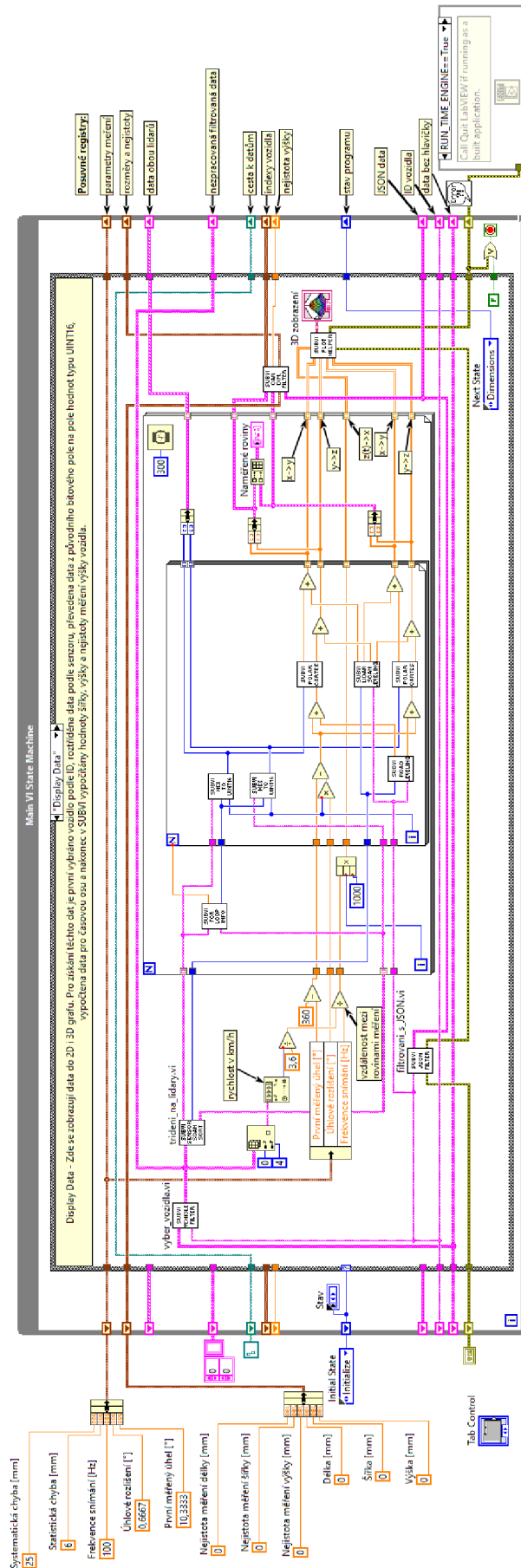
- [23] *Nejistoty v měření III: nejistoty nepřímých měření* [online]. Děčín: Automa – časopis pro automatizační techniku, s. r. o., 2001, [cit. 2021-11-17]. Dostupné z: [http://automa.cz/Aton/FileRepository/pdf\\_articles/10962.pdf](http://automa.cz/Aton/FileRepository/pdf_articles/10962.pdf)
- [24] *Nejistoty v měření II: nejistoty přímých měření* [online]. Děčín: Automa – časopis pro automatizační techniku, s. r. o., 2001, [cit. 2021-11-17]. Dostupné z: [http://automa.cz/Aton/FileRepository/pdf\\_articles/10961.pdf](http://automa.cz/Aton/FileRepository/pdf_articles/10961.pdf)
- [25] Datový list výrobku SICK LMS511-10100 PRO. In: *SICK Sensor Intelligence* [online]. Praha: SICK, 2021 [cit. 2021-11-17]. Dostupné z: [https://cdn.sick.com/media/pdf/1/41/941/dataSheet\\_LMS511-10100-PRO\\_1046135\\_cs.pdf](https://cdn.sick.com/media/pdf/1/41/941/dataSheet_LMS511-10100-PRO_1046135_cs.pdf)
- [26] Datový list výrobku SICK LMS111-10100. In: *SICK Sensor Intelligence* [online]. Praha: SICK, 2021 [cit. 2021-11-17]. Dostupné z: [https://cdn.sick.com/media/pdf/2/42/842/dataSheet\\_LMS111-10100\\_1041114\\_cs.pdf](https://cdn.sick.com/media/pdf/2/42/842/dataSheet_LMS111-10100_1041114_cs.pdf)
- [27] WLR-711 LiDAR Product Manual. In: *VanJee Technology* [online]. Beijing: VanJee Technology Co.,Ltd, 2021 [cit. 2021-11-17]. Dostupné z: <http://www.vanjee.net/index.php?m=content&c=index&a=show&catid=109&id=120>
- [28] Scanning Laser Range Finder UTM-30LX-FEW Specification: Drawing No. C-42-04155. In: *Hokuyo USA* [online]. Indian Trail, NC: Hokuyo Automatic USA Corporation, 2016 [cit. 2021-11-17]. Dostupné z: [https://hokuyo-usa.com/application/files/5815/9111/2480/UTM-30LX-FEW\\_Specifications.pdf](https://hokuyo-usa.com/application/files/5815/9111/2480/UTM-30LX-FEW_Specifications.pdf)
- [29] 2D-LiDAR UGM-50LAP (UUGM002) Specification: Drawing No. C-42-04425. In: *Hokuyo USA* [online]. Indian Trail, NC: Hokuyo Automatic USA Corporation, 2021 [cit. 2021-11-17]. Dostupné z: [https://hokuyo-usa.com/application/files/9916/1124/5909/UGM-50LAP\\_PNP\\_Specifications.pdf](https://hokuyo-usa.com/application/files/9916/1124/5909/UGM-50LAP_PNP_Specifications.pdf)
- [30] OCTAVIA TECHNICKÁ DATA: Rozměry vozu. In: *Škoda auto* [online]. Mladá Boleslav: Škoda auto a.s., 2021 [cit. 2021-12-23]. Dostupné z: <https://www.skoda-auto.cz/modely/octavia/octavia-iv/octavia-iv-technicka-data>
- [31] ŠKODA OCTAVIA iV. In: *Škoda auto* [online]. Mladá Boleslav: Škoda auto a.s., 2021 [cit. 2022-01-01]. Dostupné z: [https://www.skoda-storyboard.com/en/images/?\\_search\\_type%5B%5D=image&filter%5Bmodel%5D%5B%5D=octavia&filter%5Bderivative%5D%5B%5D=iv](https://www.skoda-storyboard.com/en/images/?_search_type%5B%5D=image&filter%5Bmodel%5D%5B%5D=octavia&filter%5Bderivative%5D%5B%5D=iv)

- [32] VOJÁČEK, Jakub. Sinová a kosinová věta: Dvě matematické věty, které popisují závislost úhlů a stran v obecném trojúhelníku. In: *Matematika pro každého* [online]. Matematika pro každého, c2008-2022 [cit. 2022-01-01]. Dostupné z: <https://maths.cz/clanky/86-sinova-a-kosinova-veta>
- [33] What Is LabVIEW?. In: *National Instruments* [online]. Austin, TX: NATIONAL INSTRUMENTS CORP., 2022 [cit. 2022-05-12]. Dostupné z: <https://www.ni.com/cs-cz/shop/labview.html#>
- [34] Modify the Simple State Machine LabVIEW Template. In: *National Instruments* [online]. Austin, TX: NATIONAL INSTRUMENTS CORP., 2022 [cit. 2022-05-12]. Dostupné z: <https://knowledge.ni.com/KnowledgeArticleDetails?id=kA03q000000YICmCAO&l=cs-CZ>
- [35] Animace skenování pomocí metody LIDAR. In: *Wikipedie* [online]. San Francisco: Wikipedie, 2008 [cit. 2021-11-17]. Dostupné z: <https://commons.wikimedia.org/wiki/File:LIDAR-scanned-SICK-LMS-animation.gif>
- [36] Záběr z Google Street View. In: *Mapy Google* [online]. Mountain View, CA: Google, 2021 [cit. 2021-11-17]. Dostupné z: <https://www.google.cz/maps/@49.2544457,17.5144403,3a,63.6y,330.79h,99.56t/data=!3m6!1e1!3m4!1swC7S8A64jw0J92vPtvXlyw!2e0!7i16384!8i8192>
- [37] Škoda konfigurátor: Octavia iV sedan Sportline. In: *Škoda auto* [online]. Mladá Boleslav: Škoda auto a.s., 2021 [cit. 2022-01-01]. Dostupné z: <https://cc.skoda-auto.com/cze/cs-CZ/colors-scenic?activePage=colors&color=2Y2Y&configurationId=&extraEquipments=&id=CZE%3Bskoda%3B2022%3BNX3RN5%3B1%3BGYCA YCA%5CGYO3YO3%3Bmda20211216031629%3Bcs-CZ%3B%3B63003%3B63003&interior=AH&modifiedPages=colors&snapshotVersion=8d081e53-609b-42c4-80b5-9feb637c2358&trimline=NXR%7CSportline6300363003&visitedPages=trimlines>

## Příloha A - SQL dotaz pro data z databáze

```
SELECT    d.vehicle_id,    va.model,    va.color,    v.lane,    v.speed,
v.total_weight, d.type, d.value, ev.stamp, ENCODE(dim.signal, 'hex'),
dim.start_angle, dim.resolution, dim.sensor_id, dim.scan_count, ev.id
FROM data.dimension AS d
INNER JOIN data.vehicle_annotation AS va
ON d.vehicle_id = va.vehicle_id
INNER JOIN data.vehicle AS v
ON va.vehicle_id = v.id
INNER JOIN raw.event AS ev
ON d.vehicle_id = ev.vehicle_id
INNER JOIN raw.dimension_signal_event AS dim
ON ev.id = dim.id
INNER JOIN data.measurement_detail AS md
on va.vehicle_id = md.vehicle_id
WHERE va.model = 'Skoda_Octavia_IV_Sedan' AND d.type = '3' AND va.weather
= 'Clear'
ORDER BY va.vehicle_id, ev.stamp
```

# Příloha B - Náhled na blokový diagram LabVIEW ve stavu Display Data





## **Příloha C - Přiložené paměťové médium**

Obsahem paměťového média je:

- Elektronická verze diplomové práce ve formátu PDF
- Vyfiltrovaná data z databáze ve formátu CSV pro vozidlo Škoda Octavia IV sedan
- Vyfiltrovaná data z databáze ve formátu CSV pro vozidlo Škoda Octavia III combi
- Vytvořený program LabVIEW pro verzi 2021
- Tabulka programu Microsoft Office Excel s porovnáním výsledných hodnot rozměrů a nejistot
- Ukázka funkčnosti programu ve formátu MP4