3. Теоретске основе проблема локализације мобилних робота

Локализација се односи на одређивање позиције и оријентације мобилног робота у односу на познату мапу окружења, која може бити представљена роботу у разним формама и облицима. Опште посматрано постоје два основна начина формирања мапе окружења:

- *Дефинисање карактеристичних објеката окружења* и
- *Дефинисање карактеристичних локација*

Дефинисање карактеристичних објеката је већ представљено увођењем модела перцепције који је управо развијен за ову сврху, док се други приступ изградњи мапе окружења базира на другачијем приступу у коме се мапа формира од мреже која се састоји од n поља квадратног облика са следећим стањима сваког поља: *заузето, слободно, недефинисано*. У циљу комплетирања Линеаризованог Калмановог филтера задржаћемо се на првом приступу.

3.1 Општа класификација

Основна класификација проблема локализације не постоји и може се направити само *неколико основних подела* у циљу потпуног и ваљаног разумевања почетног положаја:

- *Глобална локализација* – почетни положај робота (почетна позиција и оријентација) није познат. Мобилни робот се налази у окружењу и потребно је да се одреди вектор стања x_t који је, сходно апсолутном незнању о његовим почетим вредностима, моделиран као униформна расподела.

- *Локална локализација* – почетни положај робота (почетна позиција и оријентација) је познат. Потребно је одредити у сваком интервалу семпловања непознате величине. Вектор стања моделира се Гаусовом расподелом.

Као што се види, постоје решења и за потпуно познавање али и потпуно непознавање почетног положаја мобилног робота. Наравно, глобална локализација је на неки начин општији проблем од локалне која представља појам нижег реда, па самим тим и приступ таквом проблему мора да буде општији. То је основни разлог моделирања иницијалне вредности $bel(x_t)$ униформном расподелом.

Глобална локализација пружа једну велику предност приликом пројектовања алгоритама намењених локализацији мобилних робота која не постоји у случају локалне локализације. Наиме, управо то потпуно непознавање почетне позиције мобилног робота омогућава да се пројектовани алгоритми тестирају и да се одреди степен поверења и њихова употребна вредност. Мобилни робот се постави у окружење (за потребе овог теста није од важности да ли се ради о глобалној или локалној локализацији у самом почетку) и након неког временског интервала, након кога је робот утврдио свој положај у окружењу, робот се нагло премести на неку другу локацију. Овај тест од робота, конкретније од алгоритма локализације, захтева потпуну *флексибилност* и *робусност*, што је од апсолутне важности за експлоатацију у стварном окружењу у реалном времену.

Окружење у коме се робот креће може бити класификовано као:

- *Статичко окружење* – то је оно окружење у коме се једино мобилни робот помера, односно, статичко окружење се може дефинисати и као окружење у коме је једина променљива мобилни робот.

- *Динамичко окружење* – је оно окружење у коме поред мобилног робота и остали објекти могу мењати своју позицију и оријентацију у простору, током времена.

Потпуно је јасно да су кретање и локализација у статичком окружењу далеко једноставнији него у динамичком окружењу. Основни примери покретних објеката из окружења су: људи, аутомобили, промена интезитета осветљења (изузетно битно са становишта машинског гледања), други мобилни роботи итд. Два основна начина решавања проблема покретних објеката су: динамичке карактеристике објеката се могу подвести под вектор стања система, и сензорске информације могу бити "очишћене" од утицаја покретних објеката (потпуно игнорисање).

Иако је сам проблем локализације мобилног робота у окружењу од изузетне важности, алгоритам управљања роботом има *потпуну контролу* над акцијама робота. Међутим, у неким специјалним случајевима могуће је допустити да алгоритам (подсистем) задужен за локализацију мобилног робота буде задужен за комплетно управљање и доношење одлука мобилног робота. Другим речима, посматрано са становишта управљања мобилним роботом, локализацију можемо поделити на:

- *Активну* – алгоритми локализације имају потпуну контролу над основним алгоритмом управљања мобилним роботом.

- *Пасивна* –подсистем за локализацију мобилног робота представља само још један подсистем система управљања и има само једну намену: одређивање позиције и оријентације мобилног робота, док главни систем надгледа функционисање и извршење постављеног задатка.

Поређењем ова два приступа може се закључити да је грешка локализације значајно мања када се примени активни приступ у односу на пасивни. Претходна чињеница не би требало да представља изненађење с обзиром да се алгоритам на основу кога мобилни робот одређује позицију и оријентацију базира на минимизацији грешке локализације, па ће у ту сврху и управљање роботом, које је под контролом овог алгоритма, бити активно укључено у читав процес управо у циљу смањења поменуте грешке. С друге стране, управо та чињеница представља основни проблем приступа активне локализације. Наиме, мобилни робот *мора да* буде оспособљен да *успешно одреди свој положај* и у случају када је робот пројектован и намењен за неки други проблемски задатак. Другим речима, *локализација* представља *само један од низа проблема* које мобилни робот мора да активно и симултано решава.

Веома занимљив аспект постављеног проблема локализације представља *број мобилних робота* који симултано решавају постављени проблем. До сада је претпостављен један и само један робот у окружењу, али се у општем случају локализација може класификовати на следећи начин:

- *Један и само један мобилни робот врши локализацију* – све информације из окружења су примљене употребом сензора који се налазе на самом роботу, чији ће управљачки подсистем извршити њихову обраду и интерпретацију.

- *Два и више робота симултано решавају проблем локализације* – у току локализације оваквог типа могуће је да мобилни робот прима информације са сензора који припадају неком другом роботу.

Решавање проблема *локализације тима мобилних робота* може бити решен применом алгоритама локализације на појединачне роботе без могућности комуникације и размене прикупљених информација. Међутим, проблем постаје много занимљивији уколико се "дозволи" комуникација између чланова тима и активна размена информација о објектима у окружењу.

3.2 Линеаризовани Калманов филтер

Увођењем модела кретања и модела перцепције комплетни алгоритам Линеаризованог Калмановог филтера може се у потпуности математички приказати и доказати. У том смислу модел кретања изведен на основу транслаторне и угаоне брзине кретања ће бити употребљен, али са једном мањом изменом. Наиме, иако су у наведеном моделу кретања брзина v и угаона брзина ω мобилног робота посматране као управљања, у овом конкретном примеру за управљања ће бити усвојене *угаоне брзине точкова* ω_L и ω_D , односно, угаона брзина левог и десног точка респективно. Аналитичка веза између изабраних управљања са транслаторном брзином и угаоном брзином мобилног робота је:

$$\begin{aligned} v &= \frac{R}{2}(\omega_L + \omega_D) \\ \omega &= \frac{R}{d}(\omega_L - \omega_D) \end{aligned} \tag{3.1}$$

где R у претходним релацијама представља полупречник точка мобилног робота.

Модел кретања је дат у следећем облику:

$$\begin{Bmatrix} x(k+1) \\ y(k+1) \\ \theta(k+1) \end{Bmatrix} = \begin{bmatrix} x(k) + \frac{R}{2}(\omega_L + \omega_D)\Delta t \cos(\theta(k) + \frac{R}{d}(\omega_L - \omega_D)\Delta t) \\ y(k) + \frac{R}{2}(\omega_L + \omega_D)\Delta t \sin(\theta(k) + \frac{R}{d}(\omega_L - \omega_D)\Delta t) \\ \theta(k) + \frac{R}{d}(\omega_L - \omega_D)\Delta t \end{bmatrix} \tag{3.2}$$

Један од разлога промене компоненти вектора управљања u је везан за знатно једноставније и елегантније извођење ЛКФ-а, док се с друге стране угаоне брзине точкова могу директно контролисати командама *left_wheel_angular_velocity (numeric_value, time_duration)* и *right_wheel_angular_velocity(numeric_value, time_duration)* у управљачком софтверу. Наравно, команде *forward_velocity(numeric_value,time_duration)* и *angular_velocity(numeric_value, time_duration)* и даље важе.

Новоуведеном трансформацијом није нарушена основна идеја и поставка Линеаризованог Калмановог филтера. Алгоритам Линеаризованог Калмановог филтера[1] ће бити приказан у наставку, а математичко дефинисање ће уследити након тога.

Табела 3.1
Алгоритам Линеаризованог Калмановог филтера

1. $ALGORITAM\ Linearizovani\_Kalmanov\_filter(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t, c_t, m)$

2. $\theta = \mu_{t-1,\theta}$

3. $$G_t = \begin{pmatrix} 1 & 0 & -\frac{R\Delta t(\omega_L + \omega_D)}{2} \sin(\theta + (\omega_L - \omega_D)\frac{R\Delta t}{d}) \\ 0 & 1 & \frac{R\Delta t(\omega_L + \omega_D)}{2} \cos(\theta + (\omega_L - \omega_D)\frac{R\Delta t}{d}) \\ 0 & 0 & 1 \end{pmatrix}$$

4. $$V_t = \begin{pmatrix} \frac{R\Delta T}{2}\cos(\theta + (\omega_L - \omega_D)\frac{R\Delta t}{d}) - \frac{(R\Delta T)^2}{2d}(\omega_L + \omega_D)\sin(\theta + (\omega_L - \omega_D)\frac{R\Delta t}{d}) & \ldots.. \\ \frac{R\Delta T}{2}\sin(\theta + (\omega_L - \omega_D)\frac{R\Delta t}{d}) + \frac{(R\Delta T)^2}{2d}(\omega_L + \omega_D)\cos(\theta + (\omega_L - \omega_D)\frac{R\Delta t}{d}) & \ldots\ldots \\ \frac{R\Delta T}{d} & \end{pmatrix}$$

$$\begin{pmatrix} \frac{R\Delta T}{2}\cos(\theta + (\omega_L - \omega_D)\frac{R\Delta t}{d}) + \frac{(R\Delta T)^2}{2d}(\omega_L + \omega_D)\sin(\theta + (\omega_L - \omega_D)\frac{R\Delta t}{d}) \\ \frac{R\Delta T}{2}\sin(\theta + (\omega_L - \omega_D)\frac{R\Delta t}{d}) - \frac{(R\Delta T)^2}{2d}(\omega_L + \omega_D)\cos(\theta + (\omega_L - \omega_D)\frac{R\Delta t}{d}) \\ -\frac{R\Delta T}{d} \end{pmatrix}_{3x2}$$

[1] Основне величине дефинисане у алгоритму могу се веома лако извести применом методологије ЛКФ-а која је приказана у претходном делу рада. Приказани алгоритам изведен је за угаоне брзине точкова као управљања.

LKF ALGORITAM (nastavak)

$$M_t = \begin{pmatrix} \alpha_1 \omega_L^2 & 0 \\ 0 & \alpha_1 \omega_D^2 \end{pmatrix}$$

$$\bar{\mu}_{t-1} = \mu_{t-1} + \begin{pmatrix} \frac{R\Delta t(\omega_L + \omega_D)}{2} \cos(\theta + (\omega_L - \omega_D) \frac{R\Delta t}{d}) \\ \frac{R\Delta t(\omega_L + \omega_D)}{2} \sin(\theta + (\omega_L - \omega_D) \frac{R\Delta t}{d}) \\ \omega_t \Delta t \end{pmatrix}$$

$$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + V_t M_t V_t^T$$

$$Q_t = \begin{pmatrix} \sigma_r^2 & 0 & 0 \\ 0 & \sigma_\phi^2 & 0 \\ 0 & 0 & \sigma_s^2 \end{pmatrix}$$

for all observed features $z_t^i = (r_t^i \ \phi_t^i \ s_t^i)$ *do*

$$j = c_t^i$$

$$q = (m_{j,x} - \bar{\mu}_{t,x})^2 + (m_{j,y} - \bar{\mu}_{t,y})^2$$

$$\hat{z}_t^i = \begin{pmatrix} \sqrt{q} \\ a\tan 2(m_{j,y} - \bar{\mu}_{t,y}, m_{j,x} - \bar{\mu}_{t,x}) - \mu_{t-1,\theta} \\ m_{j,s} \end{pmatrix}$$

$$H_t^i = \begin{pmatrix} \frac{m_{j,x} - \bar{\mu}_{t,x}}{\sqrt{q}} & \frac{m_{j,y} - \bar{\mu}_{t,y}}{\sqrt{q}} & 0 \\ \frac{m_{j,y} - \bar{\mu}_{t,y}}{\sqrt{q}} & -\frac{m_{j,x} - \bar{\mu}_{t,x}}{\sqrt{q}} & -1 \\ 0 & 0 & 0 \end{pmatrix}$$

$$S_t^i = H_t^i \bar{\Sigma}_t [H_t^i]^T + Q_t$$

$$K_t^i = \bar{\Sigma}_t [H_t^i]^T [S_t^i]^{-1}$$

$$\bar{\mu}_t = \mu_t + K_t^i (z_t^i - \hat{z}_t^i)$$

$$\bar{\Sigma}_t = (I - K_t^i H_t^i) \bar{\Sigma}_t$$

endfor

$$\mu_t = \bar{\mu}_t$$

$$\Sigma_t = \bar{\Sigma}_t$$

return μ_t, Σ_t

3.3. Опис, математичко дефинисање и анализа алгоритма

3.3.1 Опис алгоритма

Основа на којој се базира алгоритам је чињеница, а у неку руку и претпоставка, да у сваком тренутку, приликом сваког мерења, робот *зна* коју препреку је путем сензора приметио (ред 10 у алгоритму – табела 3.1). Другим речима, процес идентификације карактеристичних објеката у окружењу се извршава апсолутно тачно без грешака и без конфузије. Улазне величине у алгоритам су дате преко:

- μ_{t-1} - средња вредност вероватноће расподеле положаја мобилног робота, која има облик Гаусове расподеле, у тренутку *t-1*,
- Σ_{t-1} - матрица коваријанси поменуте расподеле,
- u_t – управљање,
- z_t – мерење у тренутку t,
- c_t – вектор идентификације карактеристичног објекта (препреке),
- m – мапа окружења.

Излазне величине алгоритма су средња вредност вероватноће расподеле μ_t и матрица коваријанси Σ_t положаја мобилног робота у тренутку *t*.

Од реда бр. 2 алгоритма може се видети основна идеја линеаризације нелинеарне функције развојем исте у Тејлоров ред. У реду бр.3 основног алгоритма ЛКФ-а (Табела 3.1) приказана је матрица трансформације (Јакобијан трансформације) $G_t = \dfrac{\partial g(x_{t-1}, u_t)}{\partial x_{t-1}}$ помоћу које се и врши поменута линеаризација. Корак предикције је приказан у реду бр.6 и реду бр.7 у Табели 3.1 где се одређују предвиђене вредности тражених величина након извршеног кретања пре него што је мерење извршено. Поређењем израза за матрицу коваријанси основног ЛКФ-а, у коме модел кретања и модел перцепције нису уведени у потпуном облику, и комплетног алгоритма ЛКФ-а, може се приметити основна разлика. Наиме, грешку модела кретања је потребно *из простора управљања* превести *у простор стања система*, што се у самом изразу може и видети:

$$\overline{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + \underbrace{V_t M_t V_t^T}_{\Re(u_t) :\to \Re(x_t)} \tag{3.3}$$

Основа алгоритма је *петља* која "*пролази*" *кроз све карактеристичне објекте* у тренутку *t*. Као што је већ наглашено, ред бр.10 је од изузетне важности за успешно функционисање алгоритма, с обзиром да се управо овде претпоставља *апсолутна идентификација* карактеристичних објеката. Сваки од вектора идентификације *i*-тог објекта $c_t = \{c_t^1, c_t^2, ..., c_t^n\}$ може бити представљен компонентама које дефинишу боју, висину, ширину објекта, итд. Наравно, све наведене основне карактеристике карактеристичних објеката у окружењу су потпуно опционе, па је сходно постављеном проблему потребно и извршити дефинисање вектора идентификације објекта/објеката.

Након идентификације објекта следи наставак алгоритма представљен у виду предикције мерења. У реду бр.11 одређује се аналитичка зависност између онога што је сензорима потврђено (m_j) и уверења мобилног робота о положају у коме се налази у тренутку пријема поменуте информације из окружења ($\overline{\mu}_t$). Наравно, сама зависност дефинише дужину у Еуклидском простору. Поменута предикција мерења која укључује позицију карактеристичног објекта у односу на предвиђену позицију робота, као и угао (оријентацију) објекта може се видети у реду бр.12. Линеаризација модела перцепције која је дефинисана следећом релацијом општог типа:

$$\begin{aligned} h(x_t) &= h(\mu_{t-1}, u_t) + \frac{\partial h(x_t)}{\partial x_t}(x_t - \bar{\mu}_t) \\ &= h(\bar{\mu}_t) + H_t(x_t - \bar{\mu}_t) \end{aligned} \tag{3.4}$$

представљена је Јакобијаном трансформације H_t. Као и код модела кретања, и код модела перцепције потребно је извршити трансформацију, али овог пута је потребно предикцију (матрицу коваријанси уверења робота о положају) превести у простор мерења, односно:

$$S_t^i = \underbrace{H_t^i \bar{\Sigma}_t [H_t^i]^T}_{\Re(x_t) :\to \Re(z_t)} + Q_t \tag{3.5}$$

Сада је могуће одредити матрицу Калмановог појачања, која је дата у реду бр.15, након чега је једноставно срачунати тражене величине μ_t и Σ_t. Потребно је нагласити да се петља извршава за сваки појединачни примећени објекат (ред бр.9), чиме се обезбеђује да актуелне информације буду укључене у комплетан процес што је могуће пре.

3.3.2 Математичко дефинисање алгоритма

Све неопходне величине одређују се применом већ описаног поступка дефинисања основних величина Линеаризованог Калмановог филтера. Применом основних теорема математичке анализе у матричном облику долази се до основних израза који су наведени у алгоритму. То је и основни разлог што, иако се зове *Математичко дефинисање алгоритма*, неће бити приказан комплетан поступак одређивања основних величина.

Модел кретања је дат једначином (3.2), где су за управљања система усвојене угаоне брзине обртања точкова робота, $u_t = \{\omega_L \ \omega_D\}^T$. Модел кретања се може написати и у следећем облику:

$$\begin{Bmatrix} x' \\ y' \\ \theta' \end{Bmatrix} = \underbrace{\begin{Bmatrix} x \\ y \\ \theta \end{Bmatrix} + \begin{bmatrix} \frac{R}{2}(\omega_L + \omega_D)\Delta t \cos(\theta + \frac{R}{d}(\omega_L - \omega_D)\Delta t) \\ \frac{R}{2}(\omega_L + \omega_D)\Delta t \sin(\theta + \frac{R}{d}(\omega_L - \omega_D)\Delta t) \\ \frac{R}{d}(\omega_L - \omega_D)\Delta t \end{bmatrix}}_{g(u_t, x_{t-1})} \tag{3.6}$$

Даље, развијањем у Тејлоров ред добијамо:

$$\begin{aligned} g(x') &= g(\mu_{t-1}, u_t) + \frac{\partial g(x_{t-1}, u_t)}{\partial x_{t-1}}(x_{t-1} - \mu_{t-1}) \\ &= g(\mu_{t-1}, u_t) + G_t(x_{t-1} - \mu_{t-1}) \end{aligned} \tag{3.7}$$

где је G_t Јакобијан трансформације, потпуно одређен следећом матрицом

$$G_t = \frac{\partial g(x_{t-1}, u_t)}{\partial x_{t-1}} = \begin{bmatrix} \frac{\partial x'}{\partial \mu_{t-1,x}} & \frac{\partial x'}{\partial \mu_{t-1,y}} & \frac{\partial x'}{\partial \mu_{t-1,\theta}} \\ \frac{\partial y'}{\partial \mu_{t-1,x}} & \frac{\partial y'}{\partial \mu_{t-1,y}} & \frac{\partial y'}{\partial \mu_{t-1,\theta}} \\ \frac{\partial \theta'}{\partial \mu_{t-1,x}} & \frac{\partial \theta'}{\partial \mu_{t-1,y}} & \frac{\partial \theta'}{\partial \mu_{t-1,\theta}} \end{bmatrix} \tag{3.8}$$

Ред бр.3 у алгоритму комплетног Линеаризованог Калмановог филтера се сада веома једноставно одређује.

Матрица грешака модела кретања је дата као:

$$M_t = \begin{pmatrix} \alpha_1 \omega_L^2 & 0 \\ 0 & \alpha_1 \omega_D^2 \end{pmatrix} \tag{3.9}$$

у простору управљања одређеног вектором $u_t = \{\omega_L \ \omega_D\}^T$ па је потребно извршити трансформацију и превести ову информацију у простор стања. У ту сврху потребно је одредити и срачунати Јакобијан ове трансформације који је дат следећом матрицом:

$$G_t = \frac{\partial g(x_{t-1}, u_t)}{\partial x_{t-1}} = \begin{bmatrix} \frac{\partial x'}{\partial v_t} & \frac{\partial x'}{\partial \omega_t} \\ \frac{\partial y'}{\partial v_t} & \frac{\partial y'}{\partial \omega_t} \\ \frac{\partial \theta'}{\partial v_t} & \frac{\partial \theta'}{\partial \omega_t} \end{bmatrix} \tag{3.10}$$

Након дефинисања матрице (3.10), могуће је формирати матрични израз $V_t M_t V_t^T$ чиме се у потпуности извршава пресликавање усвојене грешке кретања, која је одређена у простору управљања, у простор стања (ред бр.7).

Сходно основној поставци Линеаризованог Калмановог филтера, али и Калмановог филтера у општем облику, да би комплетан поступак био завршен неопходно је дефинисање корака корекције филтера. У ту сврху усвојен је модел перцепције у следећем облику:

$$z_t^i = \begin{Bmatrix} r_t^i \\ \phi_t^i \\ s_t^i \end{Bmatrix} = \underbrace{\begin{pmatrix} \sqrt{(m_{j,y} - y)^2 + (m_{j,x} - x)^2} \\ a\tan 2(m_{j,y} - \bar{\mu}_{t,y}, m_{j,x} - \bar{\mu}_{t,x}) - \mu_{t-1,\theta} \\ m_{j,s} \end{pmatrix}}_{h(x_t, j, m)} + \aleph(0, Q_t) \tag{3.11}$$

где су са $(m_{j,x} \ \ m_{j,y})$ обележене координате објекта који је идентификован сензорима уз, наравно, апсолутну идентификацију. Линеаризацијом, развијањем у Тејлоров полином добијамо:

$$h(x_t, j, m) \approx h(\bar{\mu}_t, j, m) + H_t^i (x_t - \bar{\mu}_t) \tag{3.12}$$

Јакобијан ове трансформације је:

$$H_t^i = \frac{\partial h(\overline{\mu}_t, j, m)}{\partial x_t} = \begin{bmatrix} \frac{\partial r_t^i}{\partial \overline{\mu}_{t,x}} & \frac{\partial r_t^i}{\partial \overline{\mu}_{t,y}} & \frac{\partial r_t^i}{\partial \overline{\mu}_{t,\theta}} \\ \frac{\partial \phi_t^i}{\partial \overline{\mu}_{t,x}} & \frac{\partial \phi_t^i}{\partial \overline{\mu}_{t,y}} & \frac{\partial \phi_t^i}{\partial \overline{\mu}_{t,\theta}} \\ \frac{\partial \theta_t^i}{\partial \overline{\mu}_{t,x}} & \frac{\partial \theta_t^i}{\partial \overline{\mu}_{t,y}} & \frac{\partial \theta_t^i}{\partial \overline{\mu}_{t,\theta}} \end{bmatrix} \quad (3.13)$$

чиме је у потпуности одређен ред бр.13 у алгоритму.

3.3.3 Анализа и дискусија

Као што се из изведеног алгоритма може видети, алгоритам одређује средњу вредност μ_t и матрицу коваријанси Σ_t расподеле вероватноће позиције и оријентације мобилног робота у окружењу. У самом процесу одређивања непознатих величина μ_t и Σ_t извршена је подела на два подпроцеса: *предикцију (предвиђање)* и *корекцију (ажурирање)*.

У првом циклусу, приликом предикције вредности величина на основу података (мерења и управљања) који су доступни до тог тренутка, врши се предикција непознатих величина представљених матрицама $\overline{\mu}_t$ и $\overline{\Sigma}_t$. Предикција величине $\overline{\mu}_t$ врши се простим додавањем матрице, добијене на основу модела кретања, матрици $\overline{\mu}_{t-1}$ која представља позицију и оријентацију робота у тренутку *t-1*. Овај ред алгоритма (ред бр.6) представља апсолутно логичну чињеницу о моделу кретања као моделу промене стања, који врши транзицију система из стања одређеног вектором $\overline{\mu}_{t-1}$ у стање одређено вектором $\overline{\mu}_t$. Том приликом потребно је одредити и вредност матрице коваријанси, која нам управо дефинише колико је робот сигуран у то где се налази. Сама матрица $\overline{\Sigma}_t$ представља збир две компоненте, једне која *није под утицајем кретања система* $G_t \Sigma_{t-1} G_t^T$ и друге која *представља утицај кретања* робота на предикцију $V_t M_t V_t^T$. Као што се види, прва компонента представља утицај матрице коваријанси претходног стања μ_{t-1} на будући положај робота. У циљу потпуног комплетирања процеса предикције, неопходно је овој матрици додати и усвојену грешку модела кретања преведене у простор управљања из простора стања.

Након завршетка процеса предикције потребно је извршити мерење, односно прикупљање информација из окружења, и "увести" те информације у процес закључивања. Матрица коваријанси која представља незнање робота о позицији и оријентацији које заузима обележена је са S_t. У реду бр.12 дефинисано је предвиђено мерење $\hat{z}_t^i$, које је срачунато помоћу сензорске информације $(m_{j,x} \quad m_{j,y})$ и вектора $\overline{\mu}_t$ који физички представља уверење робота о позицији и оријентацији, другим речима предикцију. Матрица Q_t је грешка мерења, односно грешка модела перцепције која мора бити укључена у одређивање непознатих величина. Незнање о тренутном положају робота је представљено матрицом коваријанси $\overline{\Sigma}_t$ која је одређена у кораку предикције. Међутим, као и приликом анализе модела промене стања, неопходно је извршити одговарајућу трансформацију. Наиме, матрица коваријанси $\overline{\Sigma}_t$ мора бити "преведена" из простора стања у простор мерења, с обзиром да је матрица Q_t дефинисана у том простору.

Увођењем вектора иновације мерења $z_t^i - \hat{z}_t^i$, који дефинише колико има нових информација у односу на предвиђену вредност $\hat{z}_t^i$, могуће је одредити све остале параметре у циљу завршетка комплетне петље. Физичко тумачење вектора иновације је следеће: што је вектор краћи, односно, што је разлика између стварног и предвиђеног мерења мања $z_t^i - \hat{z}_t^i$, то је и вероватноћа управо тог мерења у том тренутку већа.

Матрица Калмановог појачања се одређује сходно реду бр. 15 у приказаном алгоритму. Као што је већ наглашено, што матрица Калмановог појачања има већу вредност то је и поверење у остварено мерење све израженије. Основно физичко тумачење матрице Калмановог појачања је следеће: Калманово појачање скалира вектор иновације мерења и преводи га у простор стања, где управо та разлика између предвиђеног и оствареног мерења служи за померање предикције стања мобилног робота у оном правцу у коме ће се поменути вектор иновације смањити. Основни циљ управо описаног процеса је да се, што је могуће тачније, одреди позиција и оријентација робота (вектор стања) на основу изведених управљања и мерења.

Основна идеја, на којој целокупан алгоритам почива, заснива се на томе да робот *у сваком тренутку зна* који је објекат примећен у непосредној близини (мерењем). Другим речима, алгоритам претпоставља *апсолутну идентификацију и препознавање* објеката откривених приликом мерења, употребом сензора. Да би на основу ове претпоставке алгоритам, а тиме и мобилни робот, успешно функционисао у реалним условима, мапа окружења мора да буде *апсолутно тачна* и *поуздана*. С друге стране, сваки објекат посредством вектора идентификације мора да буде адекватно представљен, без обзира да ли се ради о боји, висини или некој другој карактеристичној особини објекта.

Претходна претпоставка, која је са успехом имплементирана у алгоритам, може да представља основу на којој би сам алгоритам био уопштен и више прилагођен реалним ситуацијама. Наиме, у општем случају апсолутно препознавање објекта не мора да буде тачно (и није), с обзиром да се и ова целокупна научна област развијала и развија управо због проблема са поверењем у сензорске информације и акције актуатора. Једно од уопштења основног алгоритма Линеаризованог Калмановог филтера базира се на претпоставци да је вектор идентификације објекта једнак вектору идентификације оног објекта који има највише шанси да буде примећен у датом делу окружења. Резултат ове претпоставке је имплементација додатне петље у већ постојећу петљу, а све у циљу адекватне идентификације објеката. Као што се из анализе и дискусије може видети, процес *корекције* самог Калмановог филтера и свих његових верзија, укључујући и линеаризовани облик филтера - ЛКФ, и даље траје.