



UNIWERSYTET
MIKOŁAJA KOPERNIKA
W TORUNIU

Wydział Fizyki, Astronomii
i Informatyki Stosowanej

Programowanie Robotów Mobilnych

Budowa mapy

Rafał Szczepański

e-mail: [szczepi \(at\) umk.pl](mailto:szczepi(at)umk.pl)

www.umk.pl/~szczepi

22.03.2023



Plan prezentacji

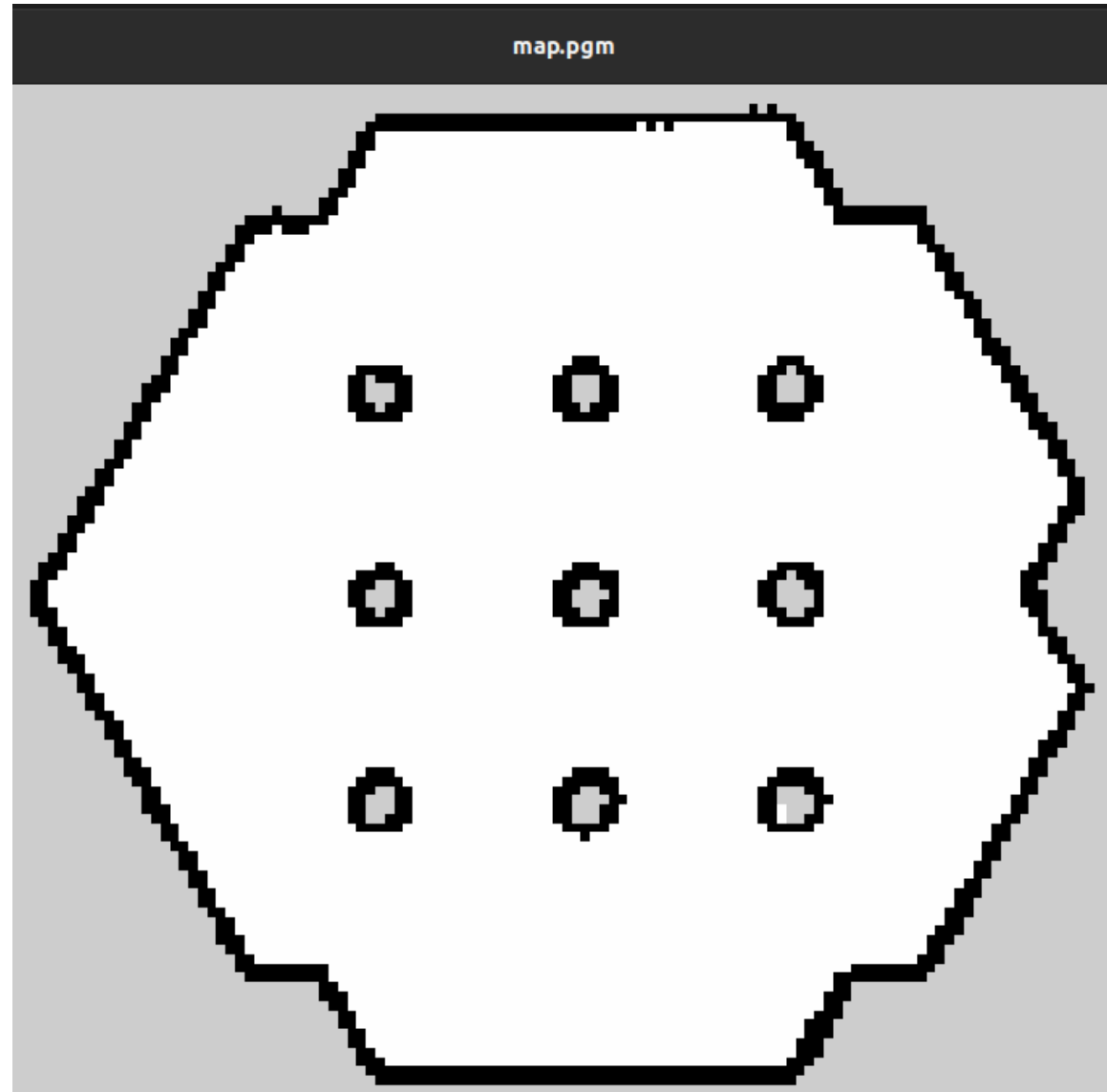
- Siatka zajętości
- Algorytm mapowania siatki zajętości
- Przykład

Siatka zajętości (ang. *occupancy grid*)

- Zdyskretyzowana siatka terenu 2D lub 3D,
- Zajętość (ang. *occupancy*) określamy obiektami statycznymi, obiekty dynamiczne nie powinny być uwzględniane na mapie,
- Każda komórka, tj. *pixel*, przyjmuje wartość $m^i \in \{0,1\}$

$m^i = 0$ (teraz wolny)

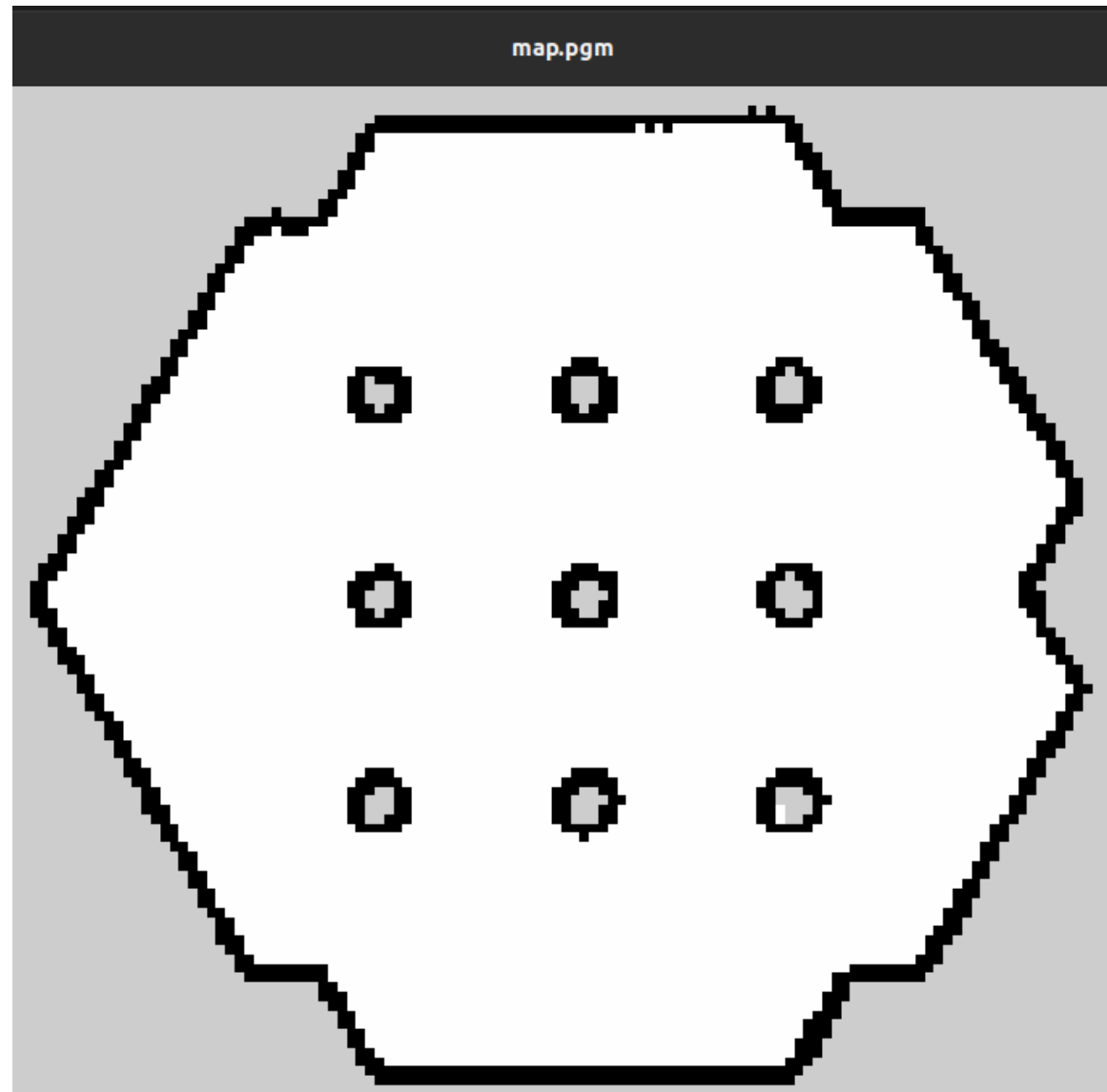
$m^i = 1$ (teraz zajęty)



Siatka zajętości

Założenia siatki zajętości:

- Statyczne środowisko,
- Niezależność pixeli,
- Znajomość dokładnego stanu robota mobilnego podczas poruszania się.





Siatka zajętości

- Siatka zajętości mapy jest przechowywana dyskretnie jako:

$$m^i \in \{0,1\}$$

- Mapa przekonania (ang. *belief map*) jest budowana na podstawie prawdopodobieństwa:

$$bel_t(m^i) = p(m^i|y_t)$$

gdzie: m^i jest zajętością i -tej komórki mapy, y_t jest pomiarem w chwili t .

- Następnie przyjmujemy wartość progową dla naszego przekonania uznając, że komórka jest zajęta ($m^i = 1$) lub wolna ($m^i = 0$).
- Informację o przekonaniu budujemy iteracyjnie w celu zwiększenia odporności na błędy i szumy pomiarowe:

$$bel_t(m^i) = p(m^i|y_{1:t})$$

- Wykorzystując twierdzenie Bayes'a dla każdej iteracji w czasie t otrzymujemy:

$$bel_t(m^i) = \frac{p(y_t|m^i)bel_{t-1}(m^i)}{p(y_t)} = \eta p(y_t|m^i)bel_{t-1}(m^i)$$

Siatka zajętości

Stała normalizująca prawdopodobieństwo do wartości $< 0,1 >$

$$bel_t(m^i) = \eta \cdot p(y_t | m^i) \cdot bel_{t-1}(m^i)$$

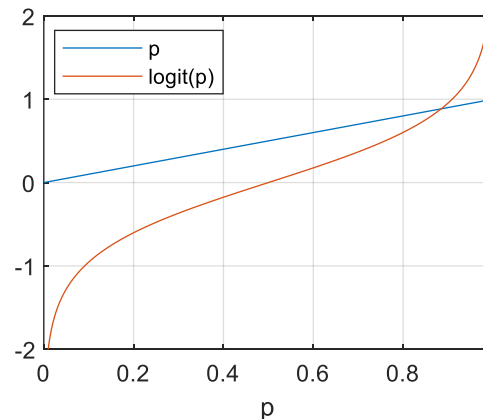
Aktualny pomiar

Poprzednia mapa przekonania

Problem z dokładnością liczb zmiennoprzecinkowych podczas mnożenia liczb blisko zera zmusza do przeskalowania prawdopodobieństwa

$$< 0,1 > \rightarrow < -\infty, \infty >$$

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right)$$



$$< -\infty, \infty > \rightarrow < 0,1 >$$

$$p = \frac{e^{\text{logit}(p)}}{1 + e^{\text{logit}(p)}}$$

Siatka zajętości

Wyznaczenie aktualnego pomiaru
z poprzedniej



$$p(m^i | y_{1:t}) = \frac{p(y_t | y_{1:t-1}, m^i) \cdot p(m^i | y_{1:t-1})}{p(y_t | y_{1:t-1})}$$

Aktualna komórka mapy

Pomiar dla danej komórki

Wyznaczenie aktualnego pomiaru
z poprzedniej

Stosując założenia Markov'a otrzymamy:

$$p(m^i | y_{1:t}) = \frac{p(y_t | m^i) \cdot p(m^i | y_{1:t-1})}{p(y_t | y_{1:t-1})}$$



Siatka zajętości

$$p(m^i | y_{1:t}) = \frac{p(y_t | m^i) \cdot p(m^i | y_{1:t-1})}{p(y_t | y_{1:t-1})}$$

Stosując twierdzenie Bayes'a do modelu pomiaru $p(y_t | m^i) = \frac{p(m^i | y_t) \cdot p(y_t)}{p(m^i)}$ otrzymujemy:

$$p(m^i | y_{1:t}) = \frac{p(m^i | y_t) \cdot p(y_t) \cdot p(m^i | y_{1:t-1})}{p(m^i) \cdot p(y_t | y_{1:t-1})}$$

Z powyższego równania będziemy liczyć funkcję logit:

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right)$$



Siatka zajętości

$$p(\neg m^i | y_{1:t}) = 1 - p(m^i | y_{1:t}) = \frac{p(\neg m^i | y_t) \cdot p(y_t) \cdot p(\neg m^i | y_{1:t-1})}{p(\neg m^i) p(y_t | y_{1:t-1})}$$

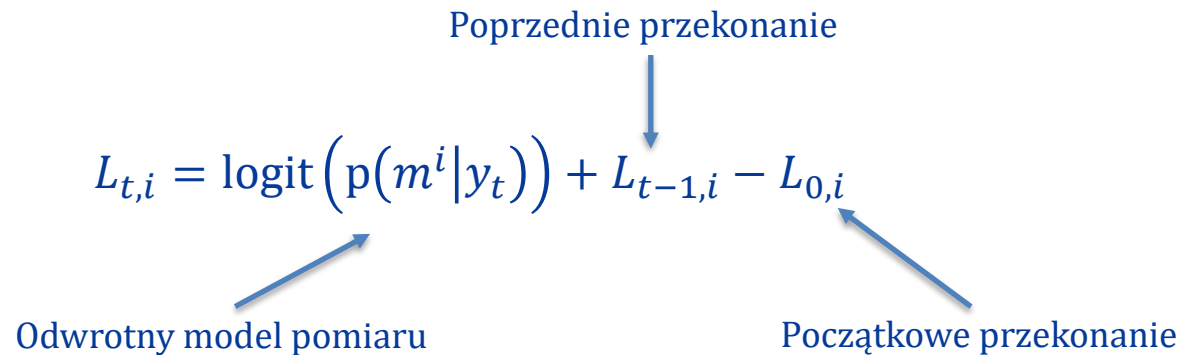
$$\frac{p(m^i | y_{1:t})}{1 - p(m^i | y_{1:t})} = \frac{\frac{p(m^i | y_t) \cdot p(y_t) \cdot p(m^i | y_{1:t-1})}{p(m^i) p(y_t | y_{1:t-1})}}{\frac{p(\neg m^i | y_t) \cdot p(y_t) \cdot p(\neg m^i | y_{1:t-1})}{p(\neg m^i) p(y_t | y_{1:t-1})}}$$

$$\frac{p(m^i | y_{1:t})}{1 - p(m^i | y_{1:t})} = \frac{p(m^i | y_t) \cdot p(\neg m^i) \cdot p(m^i | y_{1:t-1})}{p(\neg m^i | y_t) \cdot p(m^i) \cdot p(\neg m^i | y_{1:t-1})}$$

$$\text{logit}(p(m^i | y_{1:t})) = \text{logit}(p(m^i | y_t)) + \text{logit}(p(m^i | y_{1:t-1})) - \text{logit}(p(m^i))$$

Siatka zajętości

$$\text{logit}\left(p(m^i|y_{1:t})\right) = \text{logit}\left(p(m^i|y_t)\right) + \text{logit}\left(p(m^i|y_{1:t-1})\right) - \text{logit}\left(p(m^i)\right)$$

$$L_{t,i} = \text{logit}\left(p(m^i|y_t)\right) + L_{t-1,i} - L_{0,i}$$


Poprzednie przekonanie

Odwrotny model pomiaru

Początkowe przekonanie

Otrzymane rozwiązanie jest stabilne numeryczne oraz efektywne obliczeniowo.



Odwrotny model pomiaru dla skanera laserowego LiDAR

$$L_{t,i} = \text{logit}\left(p(m^i|y_t)\right) + L_{t-1,i} - L_{0,i}$$

Stan siatki zajętości na podstawie pomiaru

Model pomiaru $(y_t|m)$ możemy interpretować jako:

Pod warunkiem mapy m jakie są pomiary odległości y_t

Odwrotny model pomiaru $(m|y_t)$ możemy interpretować jako:

Pod warunkiem pomiarów odległości y_t jaka jest mapa m

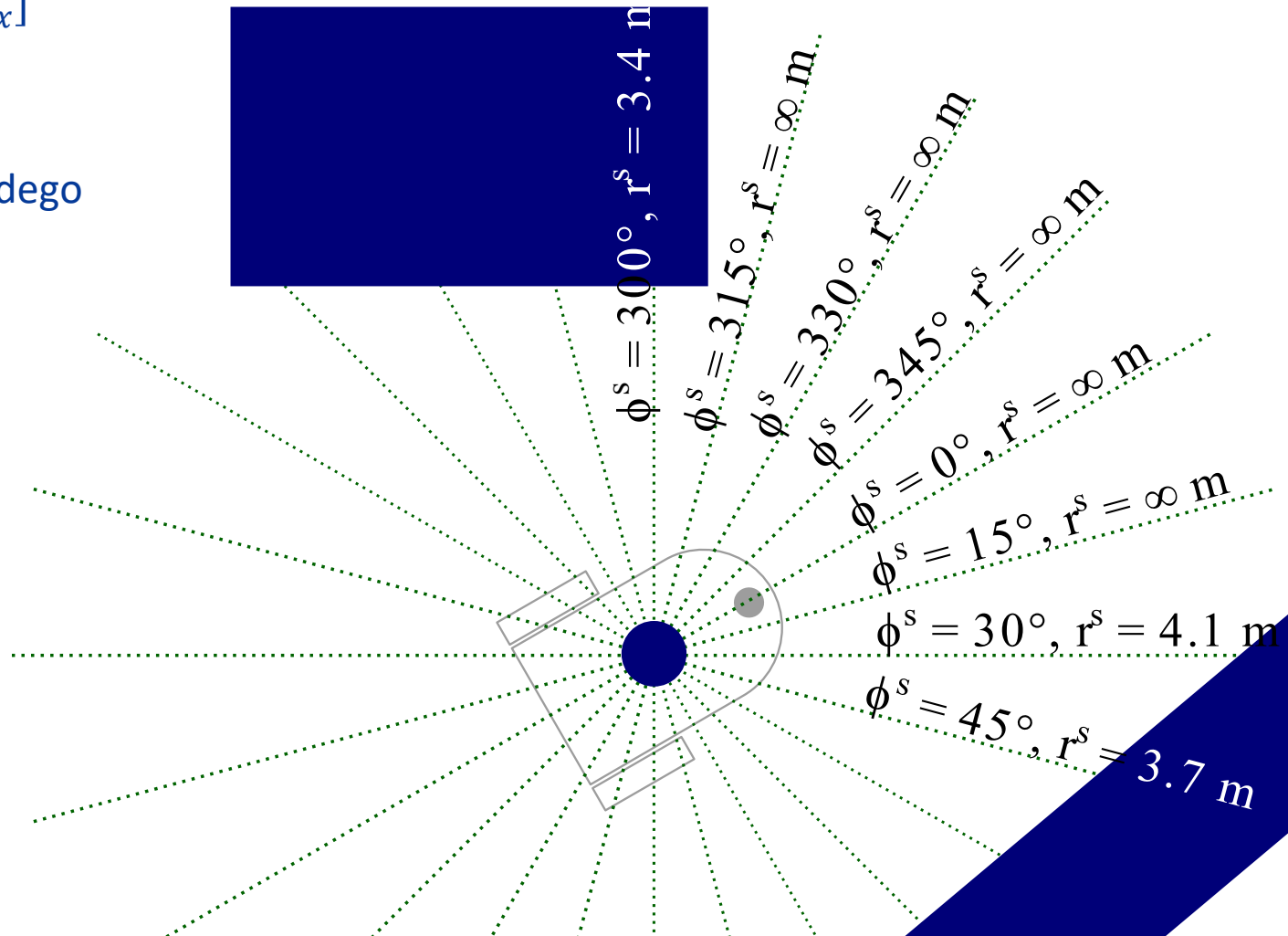
Odwrotny model pomiaru dla skanera laserowego LiDAR

Kąty pod jakimi otrzymujemy pomiar:

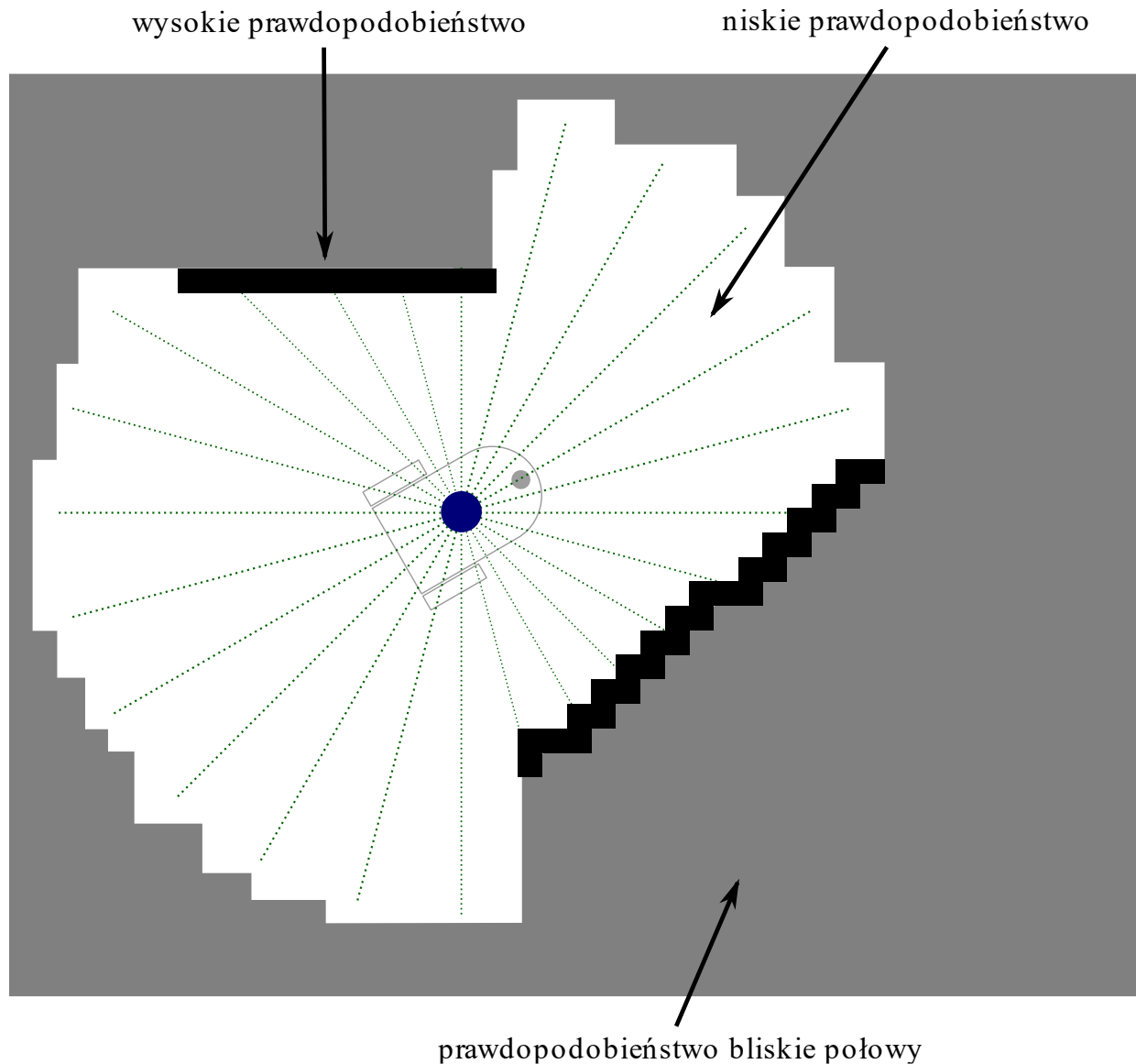
$$\phi^s = [\phi_{min}^s \quad \dots \quad \phi_{max}^s]$$
$$\phi_j^s \in \phi^s$$

Zasięg pomiaru dla każdego
z kątów:

$$r^s = [r_1^s \quad \dots \quad r_j^s]$$
$$r_j^s \in [0, r_{max}^s]$$

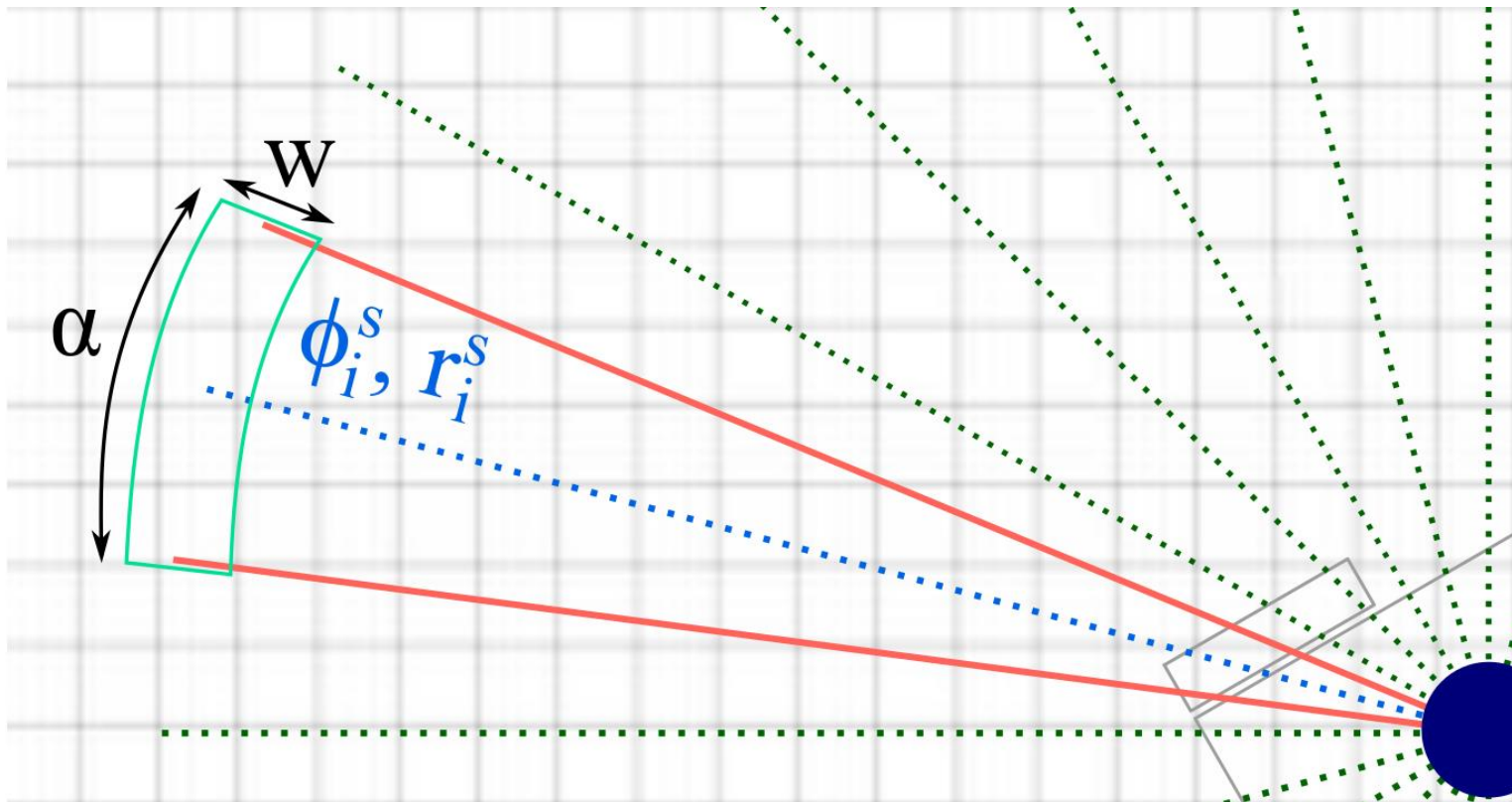


Odwrotny model pomiaru dla skanera laserowego LiDAR



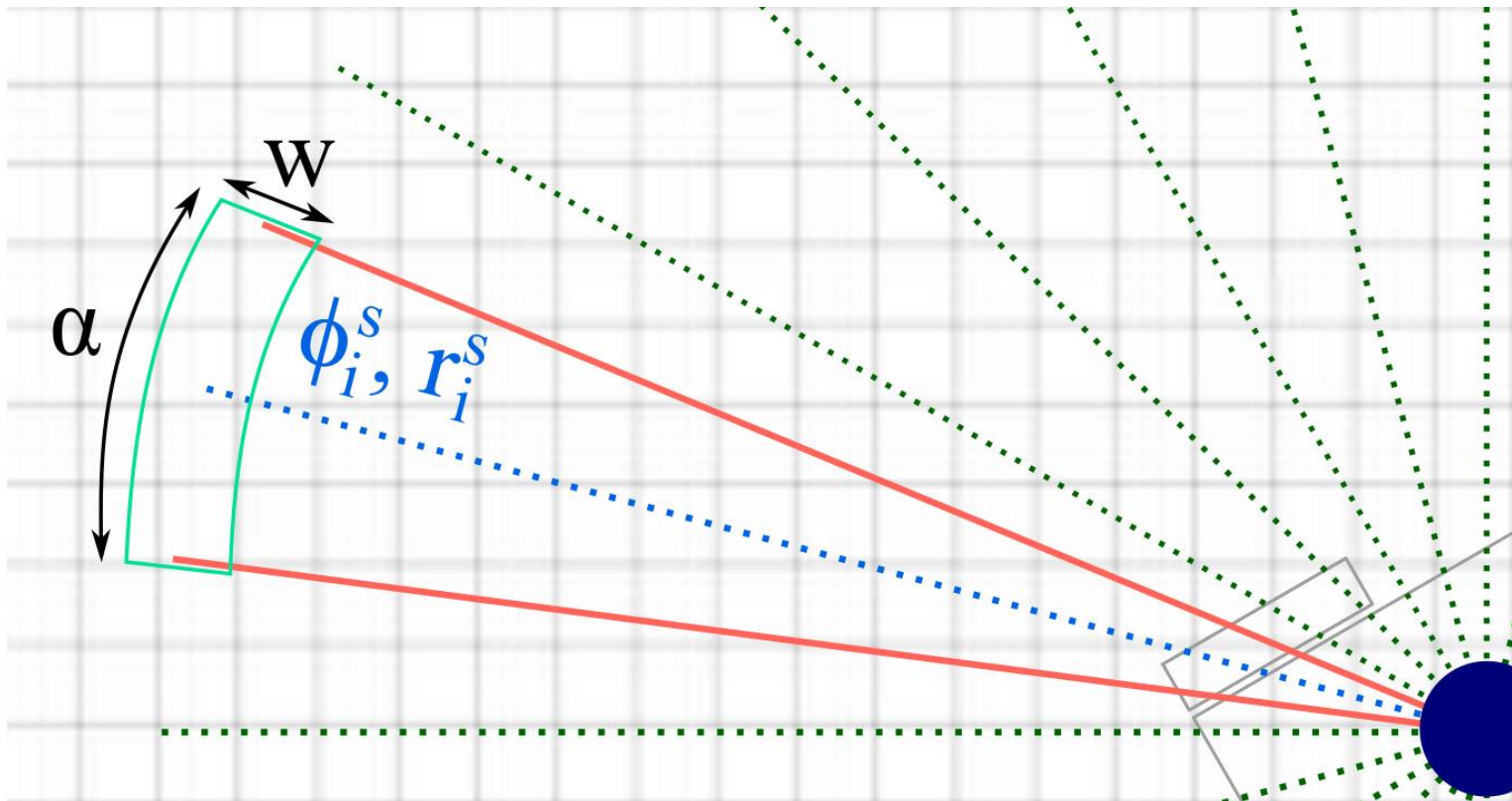
Odwrotny model pomiaru dla skanera laserowego LiDAR

- α - definiuje kąt wpływu, dla którego są uwzględniane komórki mapy
- w - definiuje szerokość wpływu przy wykryciu przeszkody



Odwrotny model pomiaru dla skanera laserowego LiDAR

- Odległość względna komórki mapy od skanera: $r^i = \sqrt{(m_x^i - x_{1,t})^2 + (m_y^i - x_{2,t})^2}$
- Orientacja względna komórki mapy od skanera: $\phi^i = \operatorname{atan2}\left(\frac{m_y^i - x_{2,t}}{m_x^i - x_{1,t}}\right) - x_{3,t}$



Odwrotny model pomiaru dla skanera laserowego LiDAR

- Indeks najbliższego pomiaru:

$$k = \arg \min_j (|\phi^i - \phi_j^s|)$$

- Wysokie prawdopodobieństwo zajętości:

Jeżeli $r_k^s < r_{max}^s$ i $|\phi^i - \phi_k^s| \leq \frac{\beta}{2}$ i $|r^i - r_k^s| < \frac{w}{2}$

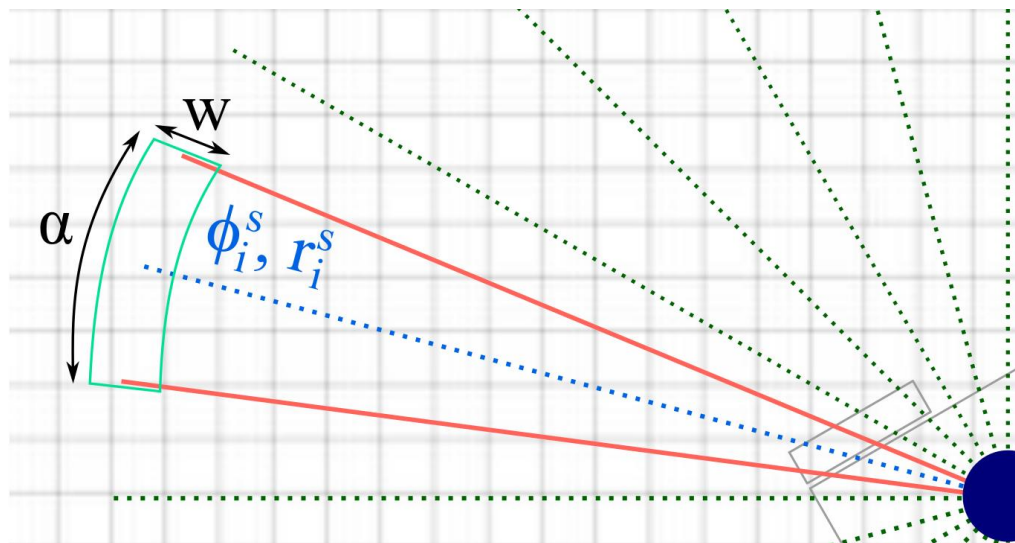
- Niskie prawdopodobieństwo zajętości:

Jeżeli $r^i < r_k^s$ i $|\phi^i - \phi_k^s| \leq \frac{\beta}{2}$

- Brak informacji na temat danej komórki:

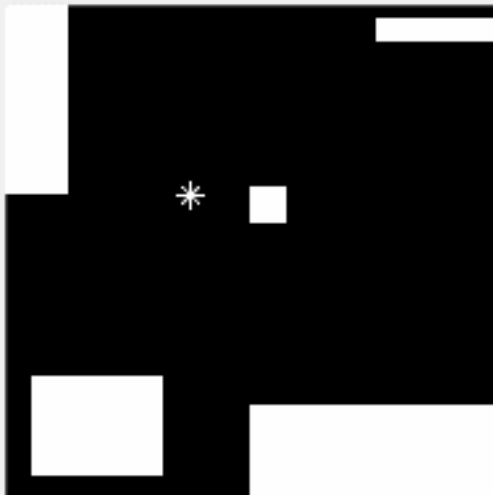
W pozostałych przypadkach

**Duża złożoność obliczeniowa
powyższej metody!**



Odwrotny model pomiaru dla skanera laserowego LiDAR

Environment map



Inverse Laser scan measurement



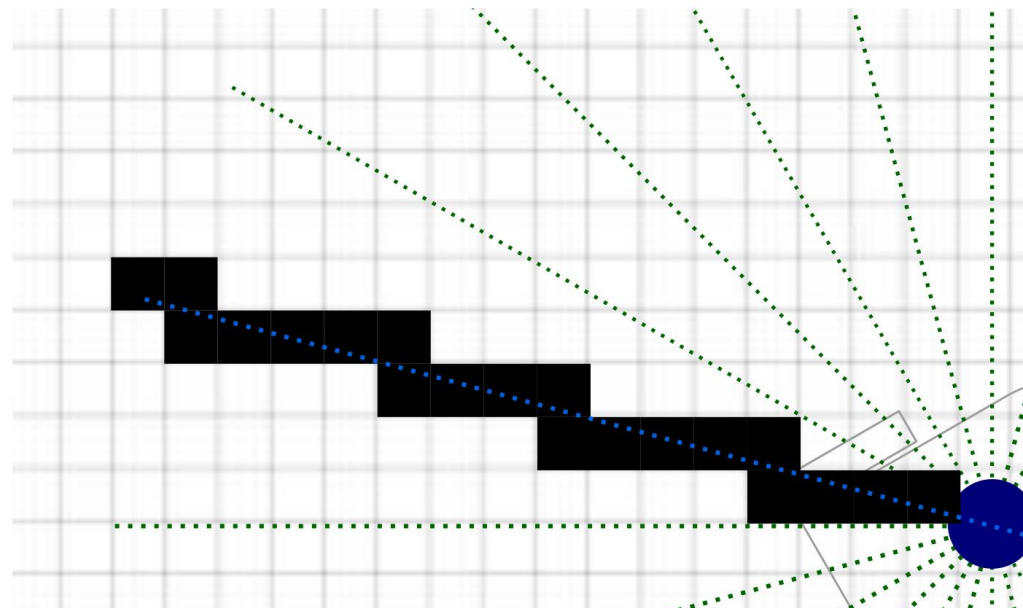
Builted map



Odwrotny model pomiaru dla skanera laserowego LiDAR

Algorytm śledzenie promieni

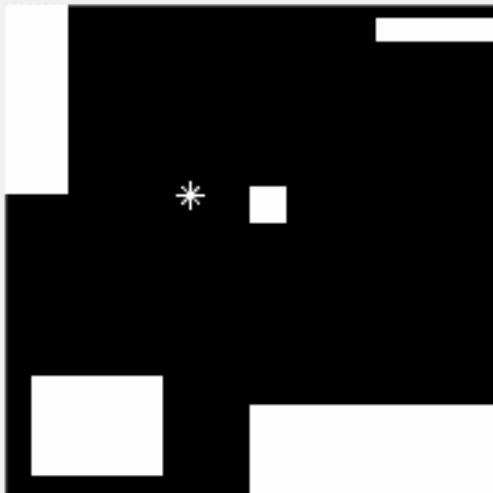
- Zamiast analizować każdą komórkę mapy i sprawdzać przedstawione wcześniej warunki można wykorzystać podejście śledzenia promieni. Znając pozycję robota i odległość odczytu (lub brak przeszkody i uwzględnić maksymalny zasięg skanera laserowego) możemy wygenerować rasteryzację linii skanera laserowego.
- Algorytm śledzenia promieni aktualizuje komórki mapy, przez którą przechodzi pomiar skanera zamiast aktualizować każdą komórkę na mapie.
- Dzięki temu wykonuje znacznie mniej aktualizacji i ignoruje przestrzeń mapy, której pomiar nie dotyczy.



Odwrotny model pomiaru dla skanera laserowego LiDAR

Algorytm śledzenie promieni

Environment map



Inverse Laser scan measurement



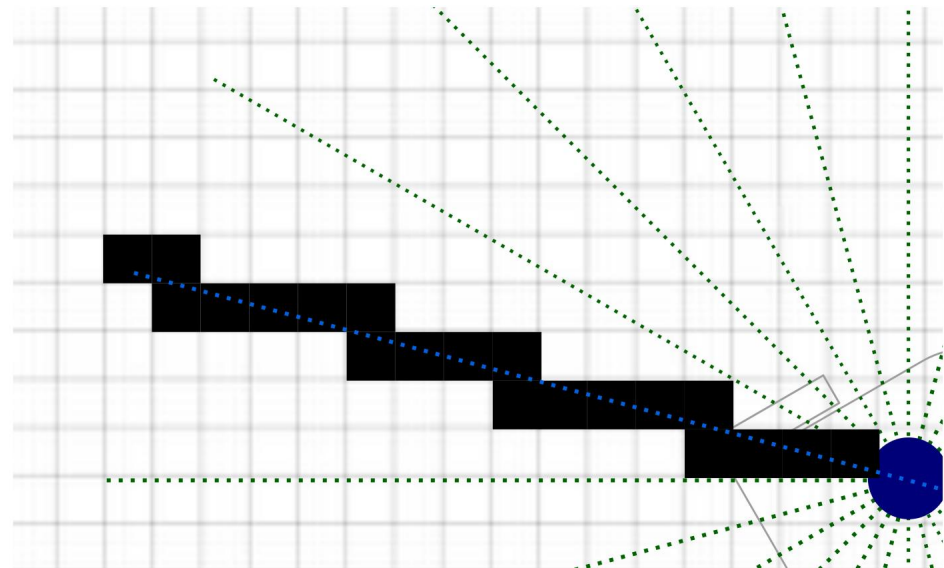
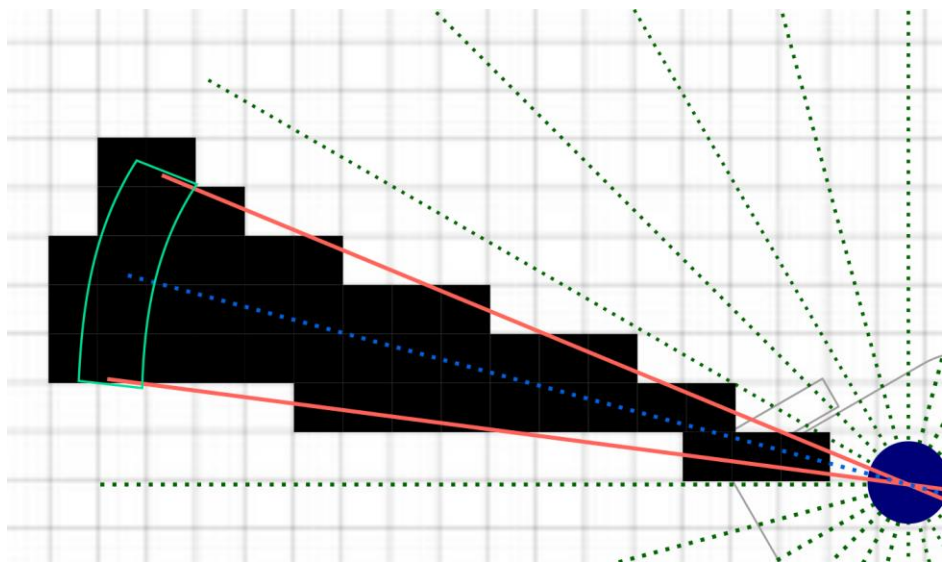
Builted map



Odwrotny model pomiaru dla skanera laserowego LiDAR

Algorytm śledzenie promieni

Alg. \ Wymiary mapy	80x80	160x160	800x800
Pełny odwrotny model	2.965804 s	11.159494 s	311.292021 s
Śledzenie promieni	0.144957 s	0.251876 s	4.624526 s



Siatka zajętości w Robot Operating System

- Składa się z dwóch plików: obrazu (np. *.pgm*, *portable grey map*) oraz pliku konfiguracyjnego (*.yaml*).
- Przykładowy plik konfiguracyjny:

image: map.pgm

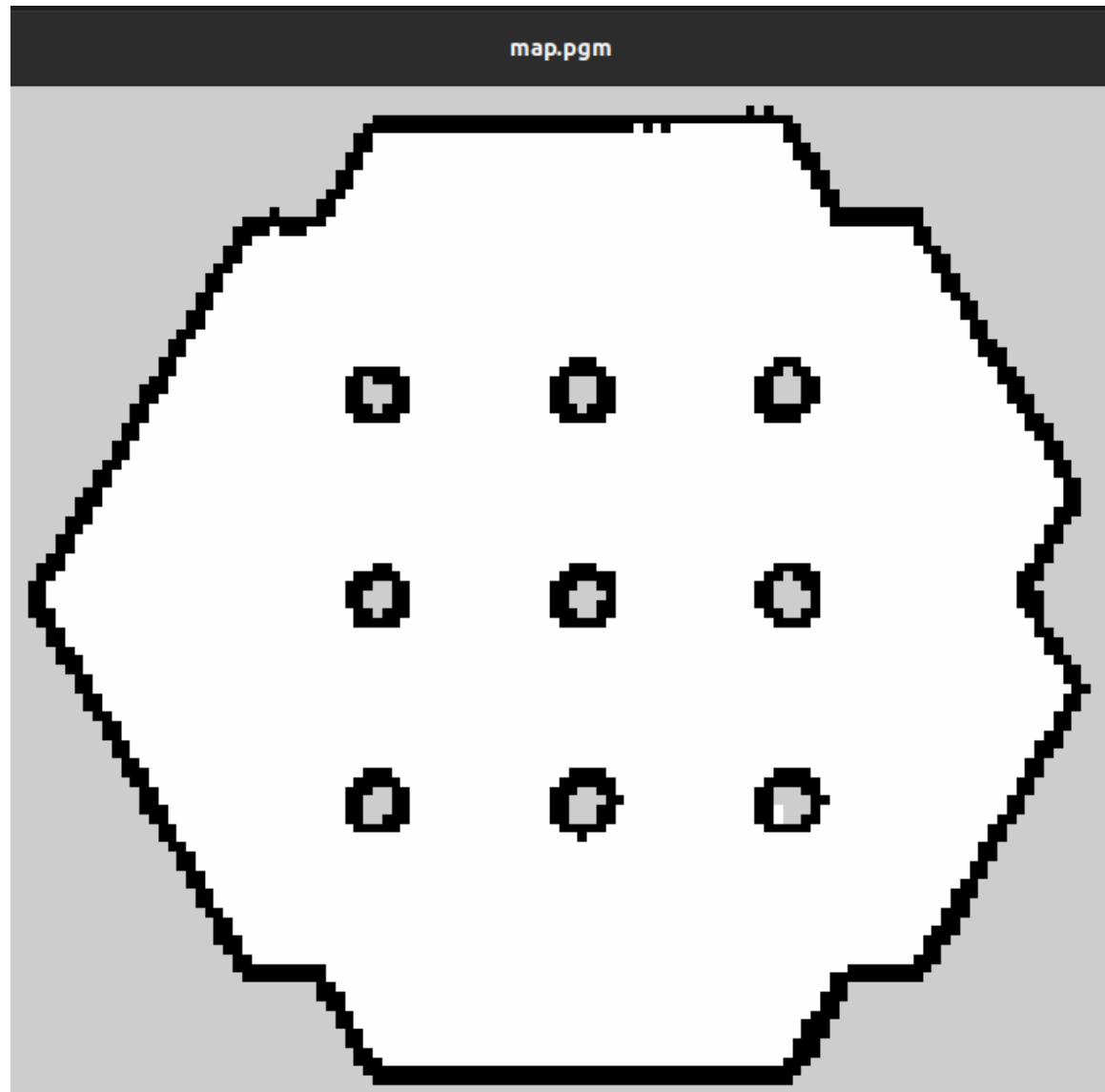
resolution: 0.1

origin: [0.0, 0.0, 0.0]

occupied_thresh: 0.65

free_thresh: 0.196

negate: 0





Siatka zajętości w Robot Operating System

- **image**: wskazuje na ścieżkę do mapy zajętości,
- **resolution**: rozdzielczość wyrażoną w metrach na piksel,
- **origin**: wskazuje pozycję dolnego lewego rogu mapy i ewentualny obrót mapy $[x, y, yaw]$,
- **occupied_thresh**: wartość progowa mapy, która oznacza, że pixel jest uważany za zajęty (ang. occupied),
- **free_thresh**: wartość progowa mapy, która oznacza, że pixel jest uważany za wolny (ang. free),
- **negate**: domyślnie kolor biały oznacza pixel wolny, a kolor czarny zajęty, jeżeli to oznaczenie na mapie jest odwrotne (tj. czarne = wolny, białe = zajęty) to należy ustawić tą opcję na wartość 1,
- **mode**: ta opcja nie występuje w powyższym pliku, ponieważ jest opcjonalna. Definiuje ona interpretację danych z mapy na dane przekazywane przez *map_server*. Domyślnie jest to wartość *trinary* (oznaczająca, że pixel może być zajęty, wolny lub niezdefiniowany). Pozostałe opcje to *scale* (skala liniowa od 0 do 100) i *raw* (surowe dane z pliku o wartościach nienormalizowanych – wartości w zakresie $[0, 255]$).