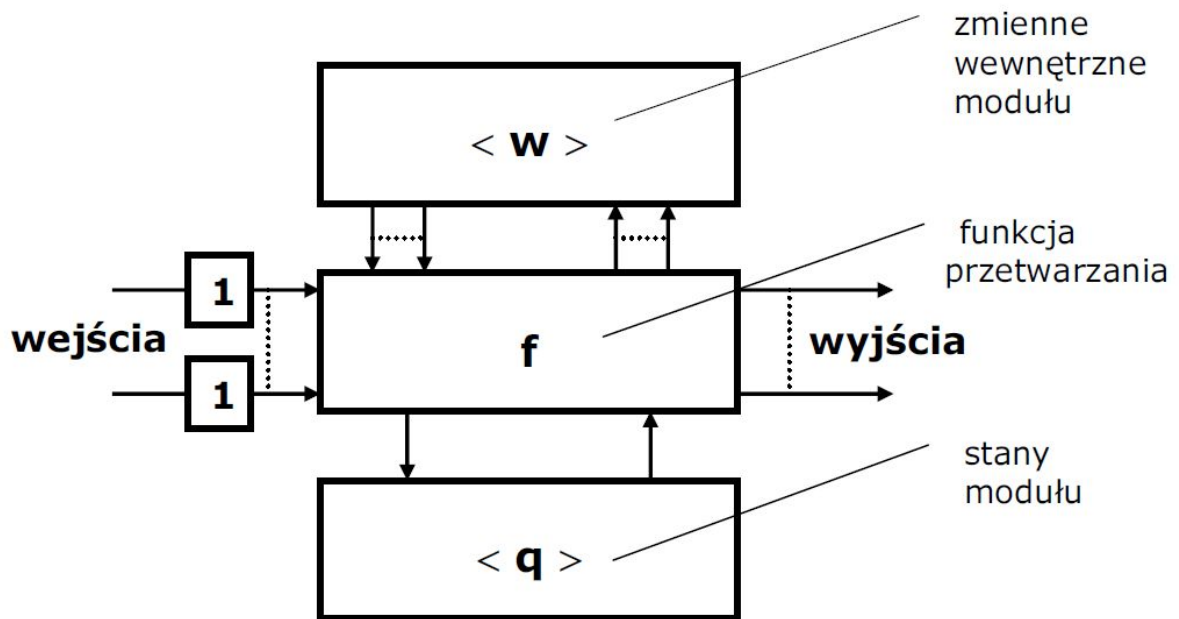


1. Budowa modułu w architekturze subsumption.

Każdy moduł jest automatem o skończonej liczbie stanów, rozszerzonym dodatkowo o zmienne wewnętrzne które mogą obsługiwać struktury danych Lispowych.

Moduł ma pewną liczbę wejść i wyjść. Linie wejściowe wyposażone są w jednoelementowe bufory, tak że ostatnia wiadomość jest zawsze dostępna, jednak może ona zostać zgubiona, gdy nie zostanie wykorzystana przed pojawieniem się kolejnej.



Wyróżnia się 4 stany modułu:

- Output – stan związany z generowaniem wyjścia (wypluwamy wyjście z modułu)
- Side effect - stan pośredni (obliczanie zmiennych na podstawie wejść)
- Conditional dispatch - stan z przejściem warunkowym (sprawdzamy jakiegoś if-a i decydujemy co dalej)
- Event dispatch - stan z przejściem zdarzeniowym (coś przyszło z wejść i w związku z tym decydujemy, co dalej)

2. Pojęcie (def.) nawigacji reaktywnej + problemy związane z określaniem funkcji odwzorowujących.

• Nawigacja reaktywna

Def. Przestrzeń percepcyjna S , jest to przestrzeń, której każda współrzędna reprezentuje sygnał z sensora (wejście systemu nawigacji).

Def. Przestrzeń komend Y , jest to przestrzeń, której każda współrzędna

reprezentuje współrzędną (parametr) komendy sterującej (wyjście systemu nawigacji)

Def. Odwzorowanie: $M : S \rightarrow Y$ przestrzeni percepcyjnej S na przestrzeń komend Y , określamy mianem **nawigacji reaktywnej**.

- **Problemy związane z określaniem funkcji odwzorowującej**

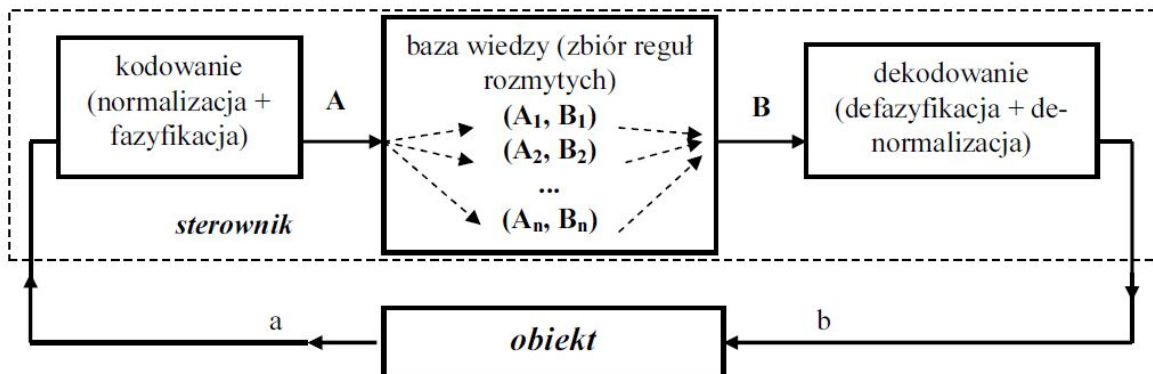
Oznaczmy przez M' odwzorowanie zdolne doprowadzić robota do celu.

Przy określaniu odwzorowania M pojawiają się dwa zasadnicze problemy:

I. M musi być lokalnie dobrą aproksymacją odwzorowania M' . Lokalnie - oznacza tu, że ono obowiązuje w pewnym regionie przestrzeni percepcyjnej (czyli szczególnym przedziale wartości danych sensorycznych) - np.: robot musi zakreślić przed ścianą;

II. M musi przybliżać M' w całej przestrzeni percepcyjnej S (tzn. zapewnić dobrą reakcję w każdej sytuacji percepcyjnej - dla każdej kombinacji danych sensorycznych).

3. Architektura sterownika opartego na logice rozmytej.



4. Pojęcie (def.) „podstawowego” problemu planowania ruchu + pojęcie (def.) przestrzeni konfiguracyjnej + pojęcie (def.) siatki punktów decyzyjnych.

- **Podstawowy problem planowania ruchu**

Mając początkową i końcową pozycję (położenie i orientację) A w W należy, określić τ opisującą ciągłą sekwencję pozycji A unikających kontaktu z przeszkodami B_i ,

prowadzącą od pozycji początkowej do końcowej, lub zasygnalizować jej brak - gdy taka ścieżka nie istnieje.

- **Przestrzeń konfiguracyjna**

Koncepcja przestrzeni konfiguracyjnej polega na reprezentowaniu robota jako punktu w odpowiedniej przestrzeni - właśnie przestrzeni konfiguracyjnej robota - oraz odwzorowaniu przeszkód jako przestrzeni.

To odwzorowanie transformuje problem planowania ruchu obiektu wymiarowego w Problem Planowania ruchu Punktu. To z kolei powoduje, że ograniczenie ruchu staje się bardziej jawne.

- **Siatka punktów decyzyjnych**

Każdą rozpoznawalną pozycję robota w W , reprezentowaną przez $q = (r, \theta)$ w C_{free} można traktować jako stan (reprezentowany przez q). Spośród tych stanów wybieramy takie, dla których istnieje prosty sposób określenia działania prowadzącego do osiągnięcia sąsiedniego stanu $q' \rightarrow q'z \quad q', q'k \in S' (?)$, gdzie: y - działanie, S' - zbiór wyróżnionych stanów spójnego regionu C_{free} .

Wyróżnione stany q' określamy mianem **punktów decyzyjnych**.

Dla potrzeb planowania trasy, ze zbioru wszystkich punktów decyzyjnych wybieramy najmniejszą liczbę punktów, które wyznaczają dowolny sposób poruszania się w otoczeniu.

Przestrzenny układ tak wybranych stanów określamy mianem **siatki punktów decyzyjnych** i oznaczamy przez S^* .

5. Metody planowania trasy (w tym ideologia klasyfikacji).

Metody planowania trasy można sklasyfikować ze względu na sposób reprezentacji wiedzy o otoczeniu.

- **Metody mapy dróg**

Polegają na uchwyceniu i odwzorowaniu spójności przestrzeni swobodnej za pomocą sieci 1-wymiarowych krzywych, zwanej **mapą dróg** $R, R \subseteq C_{free}$.

Planowanie ścieżki sprowadza się do połączenia konfiguracji początkowej q_{init} i końcowej q_{goal} z R i w poszukiwaniu ścieżki w R .

- **Metody dekompozycji komórkowej**

Polegają na podziale przestrzeni swobodnej na regiony wypukłe, zwane komórkami $K, K \subset C_{free}$ wewnątrz których ścieżka łączy dwie dowolne konfiguracje

$q_i, q_j \in K_S$ jest odcinkiem prostej. Sąsiedztwo komórek (K_i, K_j) opisuje graf spójności, w którym wierzchołki połączone krawędzią reprezentują sąsiednie komórki. Kształt komórek odwzorowuje rzeczywisty kształt fragmentów przestrzeni swobodnej C_{free} a ich suma: $\bigcup_i K_i = C_{free}$ daje całą przestrzeń.

- **Metody rastrowe**

Stosują podział całej przestrzeni konfiguracyjnej C na komórki równomiernej i poprzez odpowiednie kodowanie tych komórek wyznaczają sposób (kierunek) poruszania się robota w każdym punkcie przestrzeni swobodnej.

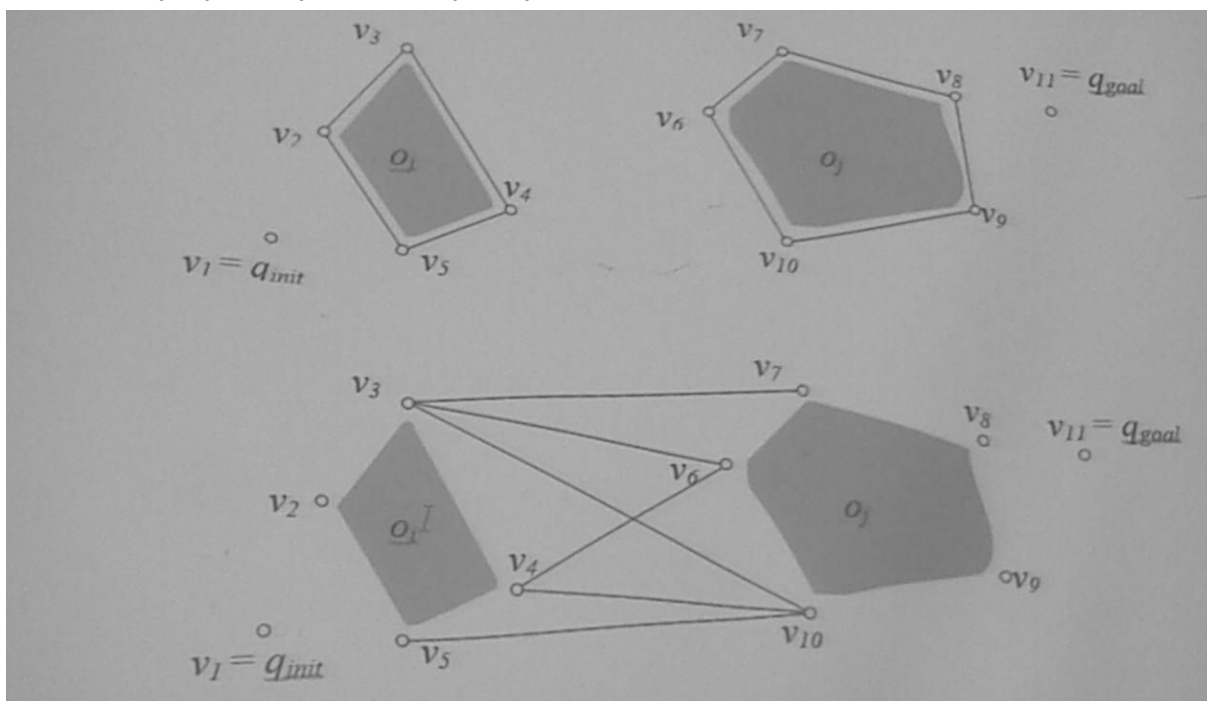
- **Metody pól potencjałów**

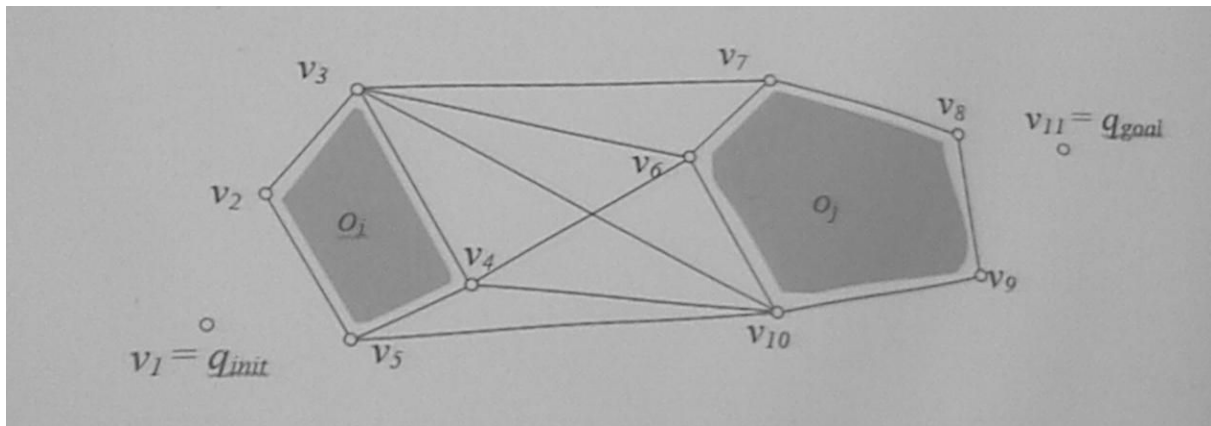
Niech robot będzie robotem mobilnym o tylko dwóch stopniach swobody (tylko translacje), w przestrzeni konfiguracji, która leży poza przeszkodami powiększonymi o rozmiar robota. Wówczas robota można traktować jako punkt. W punkcie startowym ustawiamy pewien dodatni, odizolowany ładunek elektryczny. Następnie powiększone przeszkody ładujemy dodatnio, a pozycję docelową ujemnie. Wówczas taki odizolowany ładunek będzie w naturalny sposób przyciągany do celu i jednocześnie odpychany od przeszkód.

- **Metody wektorowe**

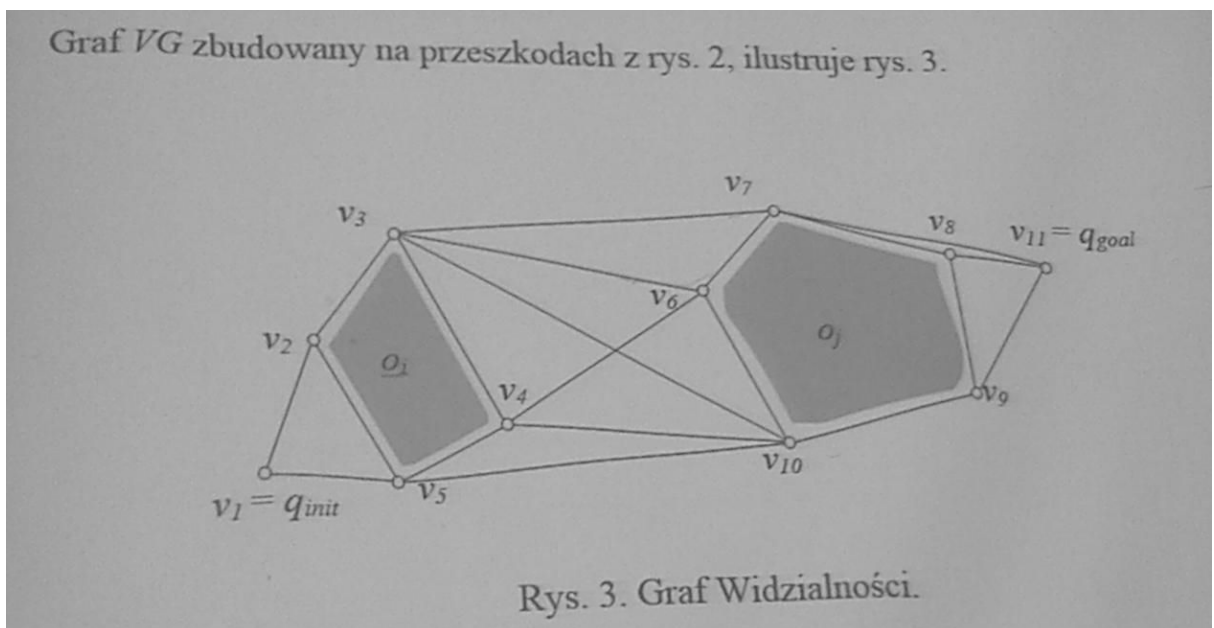
6. Metoda VG grafu widzialności (koncepcja).

Graf widzialności jest grafem nieskierowanym, którego wierzchołkami są konfiguracja początkowa, docelowa oraz wszystkie wierzchołki C-przeszkód, a krawędziami odcinki proste łączące wszystkie widzące się wierzchołki.

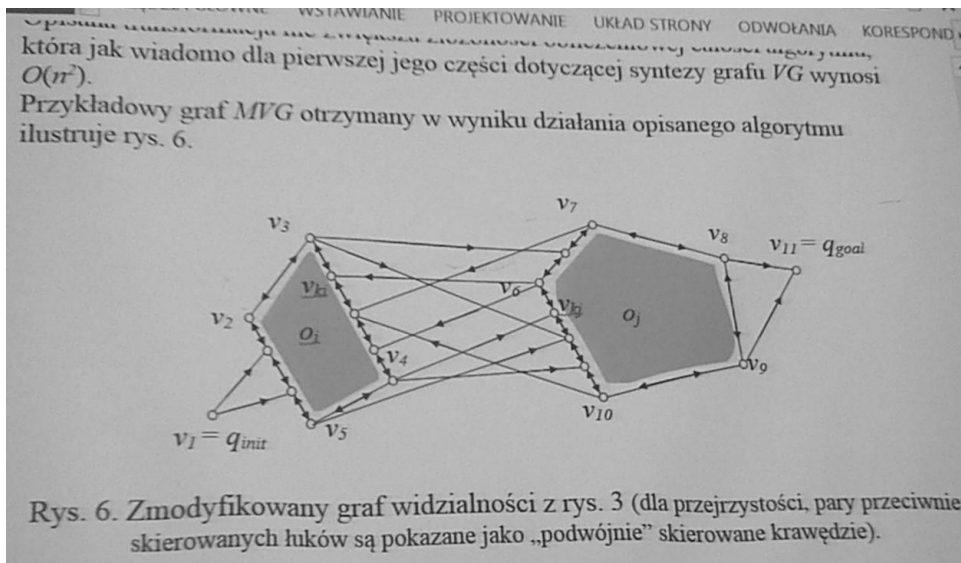




Sieć R stanowi punkt wyjścia dla określenia grafu widzialności VG. W ujęciu formalnym graf taki jest trójką (Q, S, F) , gdzie Q - zbiór wierzchołków, S - zbiór krawędzi, F - funkcja przyporządkowująca krawędziom koszt związany z ich długością.



Zmodyfikowany graf widzialności



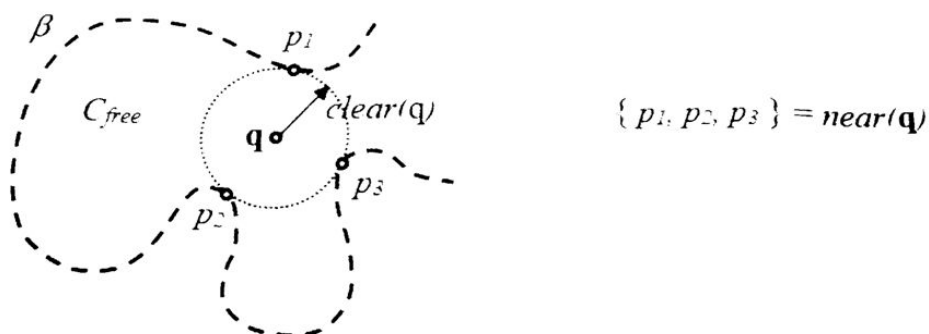
7. Metoda diagramu Voronoi'a (koncepcja).

Diagram Voronoi'a jest zbiorem konfiguracji zachowujących minimalny dystans do więcej niż jednego punktu otoczenia B przestrzeni C_{free} .

Diagramem Voronoi'a przestrzeni C_{free} nazywamy zbiór konfiguracji:

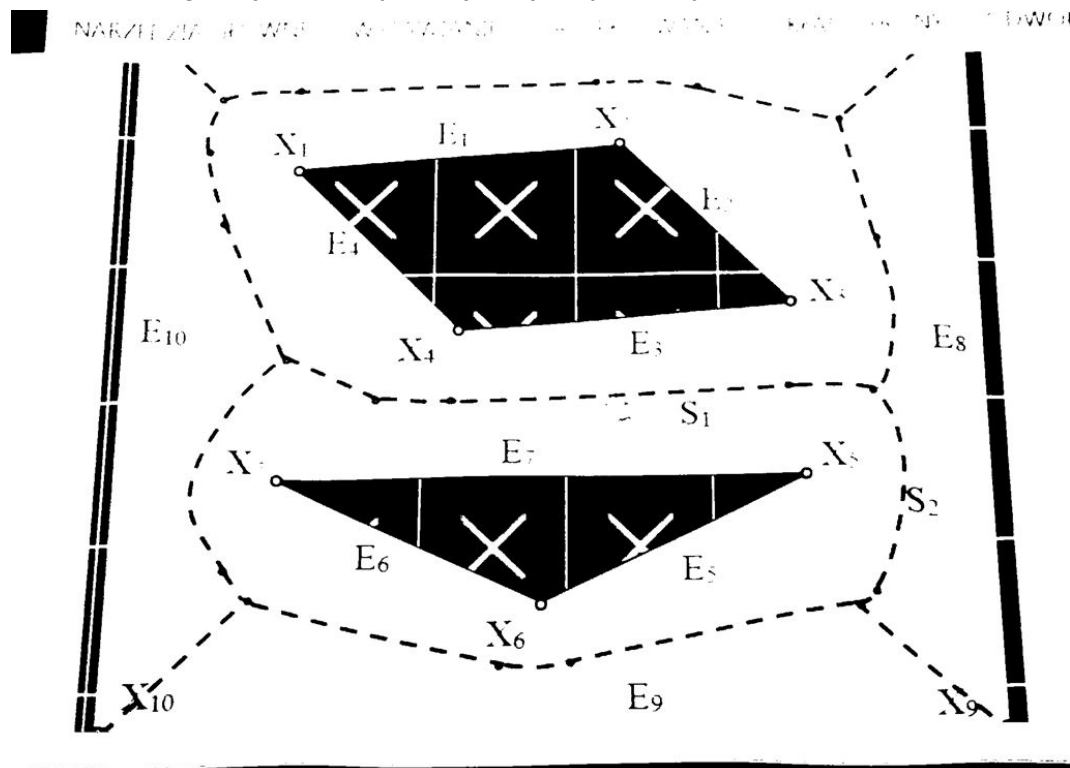
$$Vor(C_{free}) = \{ \mathbf{q} \in C_{free} \mid \text{card}(\text{near}(\mathbf{q})) > 1 \}.$$

gdzie: $\text{card } E$ – oznacza licznosc (cardinality) zbioru E .



Konstrukcja diagramu

Gdy przestrzeń swobodna C_{free} jest ograniczona wielobokami, to diagram składa się z odcinków prostych i parabol. Każdy odcinek prostej jest zbiorem konfiguracji położonych najbliżej pary "krawędź-krawędź" lub "wierzchołek-wierzchołek". Każdy odcinek paraboli jest zbiorem konfiguracji położonych najbliżej pary "krawędź-wierzchołek".



8. Metoda Retrakcji.

Metoda ta należy do grupy metod Mapy Dróg. Polega na zdefiniowaniu ciągłego odwzorowania wolnej przestrzeni C_{free} na jednowymiarową sieć krzywych R położoną w C_{free} . Retrakcja jest klasycznym pojęciem w topologii.

Def: Niech X będzie przestrzenią topologiczną/ Niech $Y \subset X$. Odwzorowanie $p: X \rightarrow Y$, które jest ciągle i którego ograniczenie do Y jest odwzorowaniem identycznym nazywane jest retrakcją X w Y .

Tw: Niech $p: C_{free} \rightarrow R$, gdzie $R \subset C_{free}$ jest siatką 1-wymiarowych krzywych, będzie zachowującą spójność retrakcją. Pomiędzy dwiema konfiguracjami q_i, q_j należącymi do C_{free} istnieje ścieżka wtedy, gdy istnieje ścieżka pomiędzy ich odwzorowaniami.

9. Metody Dekompozycji Komórkowej.

Podejście dekompozycji komórkowej polega na podziale przestrzeni swobodnej na proste obszary, zwane **komórkami** k , $k \subset C_{free}$, o takiej własności, że we wnętrzu (interior) $\text{int}(k_s)$, każdej komórki można łatwo określić ścieżkę τ łączącą dowolne dwie konfiguracje $q_i, q_j \in k_s$.

Następnie można określić i przeszukiwać nieskierowany graf G reprezentujący sąsiedztwo komórek k . Taki graf jest nazywany **grafem spójności**. Jego węzłami są komórki wolnej przestrzeni. Dwa węzły są połączone łukiem, wtedy i tylko wtedy, gdy odpowiadające im komórki sąsiadują ze sobą.

Wynikiem przeszukiwania grafu spójności jest sekwencja komórek zwana **korytarzem**. Na podstawie tej sekwencji komórek można wyliczyć ciągłą ścieżkę swobodną.

Metody dekompozycji komórkowej można podzielić na:

- (a) **metody rzeczywistej dekompozycji komórkowej** – dzielące przestrzeń swobodną na komórki, których kształt odwzorowuje konfigurację ograniczeń przestrzeni swobodnej C_{free} , a ich suma: $\bigcup_i k_i = C_{free}$ daje całą przestrzeń.

Do tej grupy metod należą m.in.:

- dekompozycja trapezoidalna (wertykalna);
- metoda c-komórek;
- metoda maksymalnych obszarów wypukłych,

- (b) **metody przybliżonej dekompozycji komórkowej** – dzielące całą przestrzeń konfiguracyjną C na komórki o z góry ustalonym kształcie (komórki rastra); kształt ten nie zależy kształtu ograniczeń przestrzeni; suma komórek reprezentujących przestrzeń swobodną jest podzbiorem tej przestrzeni $\bigcup_i k_i \subset C_{free}$.

Do tej grupy metod należy dekompozycja „quad tree” (drzewa czwórkowego).

10. Metody Rastrowe planowania trasy.

- **Metoda transformaty odległości**

Metoda transformaty odległości polega na propagacji odległości, począwszy od komórki, w której znajduje się cel, przez całą przestrzeń swobodną (każda komórka ma ośmiu sąsiadów). Czoło powstałej w ten sposób fali opływa przeszkody i przechodzi przez wszystkie wolne komórki zaznaczając w nich ich odległość od celu (wartość transformaty odległości). Przy takim kodowaniu najkrótsza ścieżka do celu przebiega po malejących wartościach transformaty, przechodząc po najbardziej stromym spadku.

- **Metoda transformaty ścieżki**

W tej metodzie używa się fali, która jest kombinacją odległości od celu i miary wyrażającej poruszanie się zbyt blisko przeszkód. W efekcie tworzy to specyficzną transformatę wykazującą cechy metod pól potencjalnych. Wyliczenie tej transformaty przebiega dwuetapowo. Najpierw obliczana jest transformata przeszkód, w której komórki przeszkód są celami. W efekcie otrzymuje się funkcję kosztu „obstacle(c)” wyrażającą wpływ najbliższej przeszkody na komórkę c.

- **Metoda sztucznych pól potencjalnych**

Niech robot będzie robotem mobilnym o tylko dwóch stopniach swobody (tylko translacje), w przestrzeni konfiguracji, która leży poza przeszkodami powiększonymi o rozmiar robota. Wówczas robota można traktować jako punkt. W punkcie startowym ustawiamy pewien dodatni, odizolowany ładunek elektryczny. Następnie powiększone przeszkody ładujemy dodatnio, a pozycję docelową ujemnie. Wówczas taki odizolowany ładunek będzie w naturalny sposób przyciągany do celu i jednocześnie odpychany od przeszkód.

- **Metoda Jacobi'ego**

Aby obliczyć wartości pola w obszarze Ω , na model otoczenia zostaje nałożona siatka o określonych rozmiarach, a na niej punkt docelowy oraz granice obszaru, do których należą powierzchnie przeszkód. Funkcja $\Phi(x, y)$ jest, więc określona wartościami w kolejnych, dyskretnych punktach. Jeśli wielkości oczek siatki są równe w całym obszarze, wówczas stosując metodę różnicową, równanie Laplace'a dla przypadku dwuwymiarowego może być przedstawione jako następująca dyskretna (iteracyjna) zależność:

$$\Phi_{i,j}^{n+1} = \frac{1}{4}(\Phi_{i+1,j}^n + \Phi_{i-1,j}^n + \Phi_{i,j+1}^n + \Phi_{i,j-1}^n)$$

- **Metoda Gauss-Seidel'a**

- Zmieniona metoda Gauss-Seidel'a
- Metoda SOR (Simultaneous Over-Relaxation)
- Metoda Elementów Skończonych (Finite Element Method)

Czy pytania będą łatwe?

