

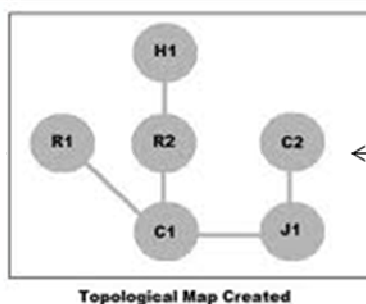
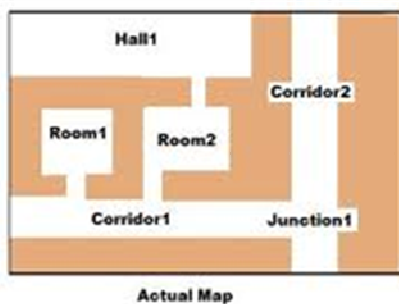
## **MODELAGEM DO AMBIENTE**

Robôs executam tarefas em ambientes povoados de objetos diversos. As tarefas geralmente envolvem se locomover no ambiente e manipular objetos. Para poder planejar e executar as tarefas, é necessário que o robô disponha de um modelo do ambiente, constituído por estruturas de dados apropriadas, a partir do qual o robô possa executar algoritmos adequados para fins de auto localização, navegação, manipulação de objetos, etc. Diferentes forma de representar o ambiente são utilizadas, dependendo da aplicação. Um modelo do ambiente é uma descrição aproximada do mundo real. A precisão desta descrição depende de vários fatores. Se o modelo é construído pelo próprio robô, a partir de dados sensoriais, a precisão dos sensores é um limitante para a própria precisão da descrição do ambiente resultante das medições efetuadas pelo robô. Por exemplo, sensores de alcance que apresentam erros de medição da ordem de 1 cm, resultarão em incertezas da mesma ordem na localização de obstáculos. Por outro lado, dependendo da tarefa a ser realizada, limitações nos recursos computacionais necessários para executá-la podem limitar a qualidade da representação do ambiente. Por exemplo, na representação aproximada da superfície de um objeto 3D por uma malha de triângulos, limitações de espaço de armazenamento e de capacidade de processamento podem limitar o número de triângulos com que a superfície é aproximada, possibilitando modelos com resolução restrita a um limite superior.

**Mapas** são formas tradicionais de representar o ambiente em que um robô executa tarefas de navegação. Tradicionalmente, os mapas são classificados em mapas métricos, mapas topológicos e mapas híbridos.

- **Mapa Métrico:** representa as dimensões físicas dos objetos presentes no ambiente e fornece informações espaciais detalhadas dos mesmos.

- **Mapa Topológico:** representa as relações de conectividade das regiões navegáveis do ambiente, geralmente na forma de um grafo.
- **Mapa Híbrido Métrico/Topológico:** incorpora informações métricas e de conectividade do ambiente. Procura aliar as vantagens de precisão dos mapas métricos às vantagens de representação de alto nível dos mapas topológicos.

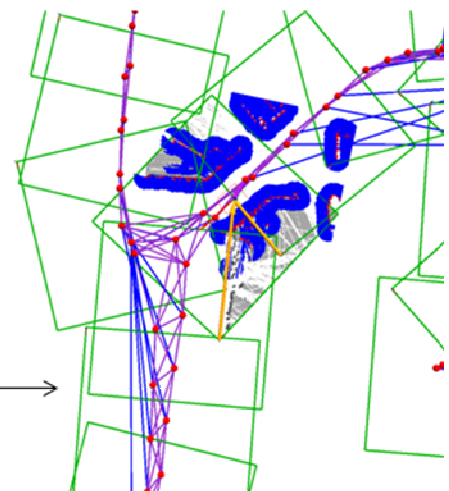


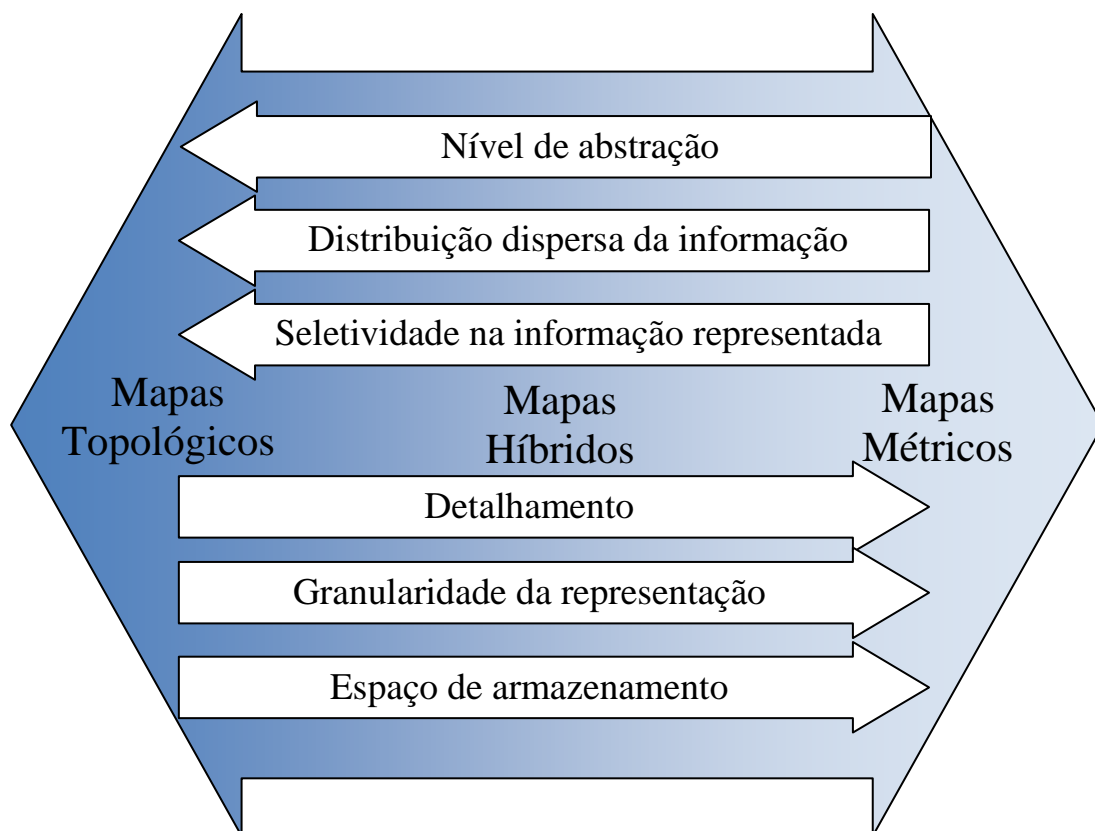
Topológico



Métrico

Híbrido



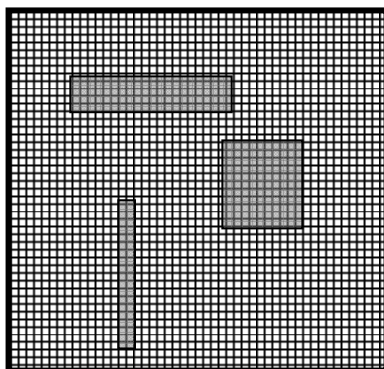


### **Mapas Métricos:**

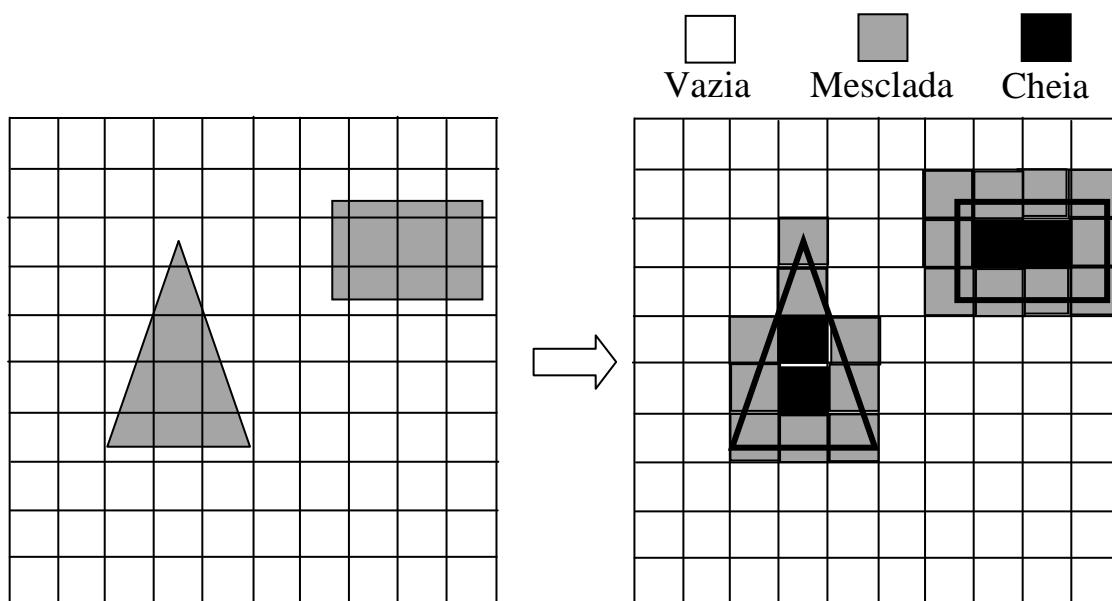
Em um mapa métrico, o ambiente é descrito de forma detalhada, incorporando informações geométricas geralmente precisas dos objetos presentes no ambiente. A granularidade da representação é alta, exigindo espaço de armazenamento grande. Os mapas métricos apresentam baixo nível de abstração e pouca seletividade em relação às informações representadas. Assim, regiões distintas costumam receber graus de importância parecidos e são representadas de maneira geralmente uniforme. A finalidade dos mapas métricos é fornecer uma descrição detalhada do ambiente, necessária para a execução das tarefas, como, por exemplo, detectar limites de áreas navegáveis ou deduzir características geométricas específicas de objetos a serem manipulados. A literatura descreve vários tipos de mapas métricos, que se caracterizam pela granularidade da representação, dentre os quais destacamos as grades de ocupação, os mapas sensoriais e os mapas de características.

## Grade de Ocupação:

Uma grade de ocupação descreve o ambiente com um baixíssimo nível de abstração. A grade de ocupação representa o espaço 2D ou 3 D como uma união de células de formato regular, geralmente do mesmo tamanho, analogamente a pixels ou a voxels.

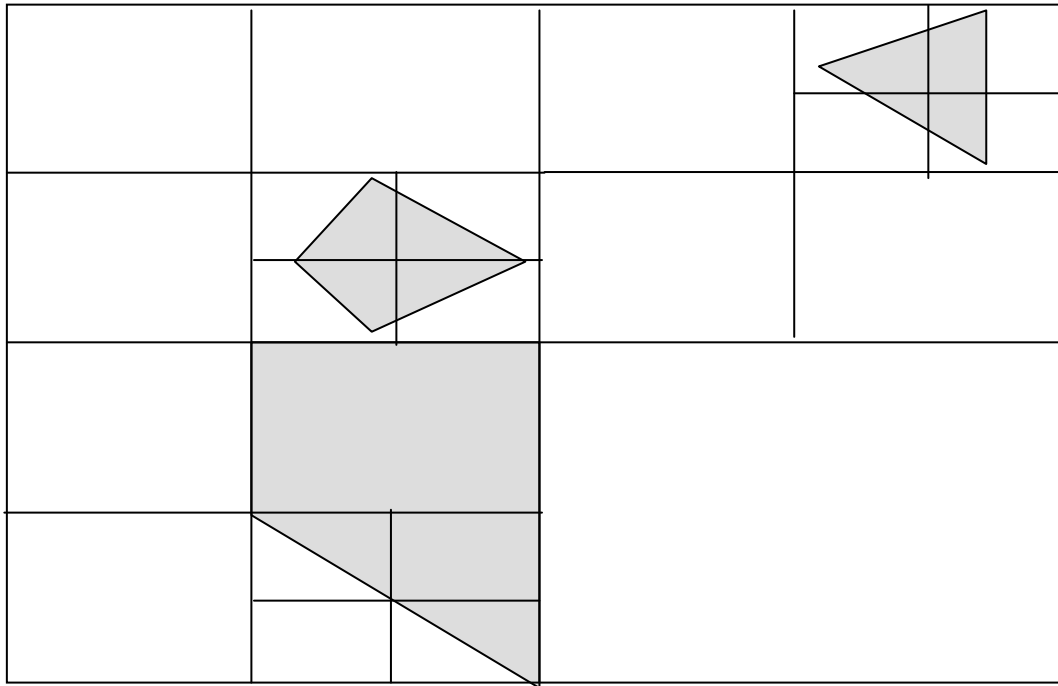


A cada célula é associado um valor de ocupação. Em uma representação determinística, a célula pode estar cheia (ocupada completamente por um obstáculo), ou vazia. Células parte ocupadas e parte vazias, são denominadas mescladas e normalmente são aproximadas como células cheias, resultando em uma descrição conservadora aproximada do ambiente.



Em aplicações onde a grade é construída a partir de medições ruidosas obtidas através do sistema sensorial do robô, uma abordagem probabilística pode ser adotada, atribuindo valores de probabilidade de ocupação a cada célula. Assim, uma célula pode assumir valores maiores ou menores de probabilidade de ocupação, conforme a confiabilidade das informações sensoriais obtidas sobre o estado da mesma. Uma célula é cheia se for estimada uma probabilidade de 100% de estar ocupada e uma célula é considerada vazia se a probabilidade de ocupação estimada para ela for de 0%. Valores de probabilidade intermediários indicam o grau de incerteza do seu estado. Células sobre as quais não se possui nenhuma informação de ocupação obtida a partir dos sensores recebem um valor de probabilidade de ocupação de 50%. A Grade de Evidência, desenvolvida pelo brasileiro Alberto Elfes, é um exemplo clássico deste tipo de mapa, [A. Elfes, “Sonar-Based Real-World Mapping and Navigation”, IEEE Journal of Robotics and Automation, vol. 3, no. 3, pp.249–265, 1987. 215].

Grades regulares geralmente requerem grandes recursos computacionais, destacando-se o espaço de memória necessário para armazenar este tipo de estrutura. É fácil perceber que, em muitos casos, este tipo de representação geralmente utiliza um grande número de células para representar regiões uniformes, (ou constituídas totalmente por células cheias, ou por células todas vazias). Desta forma, em algumas aplicações adota-se uma representação em grade de ocupação hierárquica, com células de formatos idênticos, mas de tamanhos diferentes. Assim, regiões uniformes de tamanho grande são decompostas usando células grandes. Regiões com grande densidade de obstáculos de formato complexo são modeladas usando uma maior resolução, com células de menor tamanho. Geralmente, na decomposição utilizam-se árvores do tipo *quadtree* (em espaço 2D) ou *octree* (em espaço 3D), onde as arestas das células de um nível hierárquico possuem comprimento igual à metade do comprimento das arestas das células do nível hierárquico anterior.



### **Mapas Sensoriais:**

Mapas sensoriais são um tipo de mapa métrico em que cada localização do robô representada no mapa associa-se a informação sensorial que seria percebida pelo robô naquele ponto. Assim, em lugar de características locais do ambiente independentes da percepção do robô, o mapa incorpora informação diretamente percebida pelo mesmo. O mapa sensorial é um tipo pouco usual de mapa métrico. Um exemplo desse tipo de mapa é o Mapa de Aspecto Visual, onde, a cada localização do robô registrada no mapa, associa-se a imagem que seria capturada pela câmera do robô quando posicionado naquele ponto.

### **Mapas de Características:**

Os mapas de características representam o ambiente de forma parcial. Ao contrário da representação detalhada das grades de ocupação, que exigem grande espaço de armazenamento, os mapas de características se concentram na representação algumas estruturas especiais do ambiente, tais como quinas portas, paredes, luminárias, ou outros objetos característicos presentes no mesmo. As informações relativas a estas estruturas especiais são modeladas na forma de atributos geométricos, que representam as mesmas de forma condensada, economizando recursos

computacionais. Um exemplo típico de mapa de características é a planta baixa de um prédio, onde paredes, portas, corredores e dependências são modelados usando linhas retas. Assim, elementos geométricos simples, descritos por poucos parâmetros, podem ser utilizados para descrever o ambiente. Mapas de características construídos por robôs dotados de câmeras podem usar pontos característicos destacados no ambiente para modelar o mundo. Os detectores de características locais SIFT e SURF são bastante utilizados para extrair e detectar pontos característicos em imagens do ambiente, estabelecendo descritores bastante compactos para os mesmos, garantindo um desempenho bastante robusto frente a diferentes transformações de imagens.

Estruturas geométricas mais complexas do que pontos, descritas por um pequeno número de parâmetros, podem ser utilizadas para obter representações mais compactas. Linhas, em espaços 2D e planos em espaços 3D, são bastante utilizados para representar paredes em ambientes prediais. Uma descrição bastante utilizada para mapear ambientes internos em um espaço 2D é o espaço poligonal, onde estruturas do ambiente são aproximadas por polígonos. Além de planos, em espaços 3D, estruturas geométricas mais complexas, tais como hiperquádricas. (que modelam elipsóides, hiperbolóides ou parabolóides), podem ser utilizadas.

Detectar estruturas geométricas complexas no ambiente exige esforço computacional. Assim, deve-se estabelecer um compromisso entre este esforço computacional e os ganhos em espaço de armazenamento derivados de uma representação mais compacta usando formas geométricas mais complexas. Assim, reconhecer linhas ou formas geométricas planas em imagens usando Transformada de Hough demanda menos esforço computacional que extrair uma representação em modelo de superquádrica de uma nuvem de pontos capturada por uma câmera RGB-D. Por outro lado, o reconhecimento de objetos geométricos em um espaço 3D, permite construir representações do ambiente mais poderosas, uma vez que informação semântica pode ser incorporada aos mesmos. Por exemplo se um objeto

cilíndrico é identificado como um cesto de lixo, esta informação de alto nível pode ser utilizada pelo robô na execução das suas tarefas.

### **Mapas Topológicos:**

Os mapas topológicos fornecem informações relativas às relações entre locais ou marcos presentes no ambiente, sem necessariamente incorporar informações geométricas sobre os mesmos. As relações entre as entidades mapeadas geralmente são modeladas na forma de grafos. Locais ou marcos são mapeados em nós do grafo, enquanto que relações entre os mesmos são mapeadas nos arcos do grafo.

Dependendo da aplicação, as relações no mapa topológico podem ser de vários tipos, por exemplo: relações de adjacência ou existência de um caminho entre dois locais, comandos que movem um robô de um local para outro, ou até mesmo instruções de alto nível que indicam as tarefas a serem realizadas pelo robô para ir de um local para outro, como, por exemplo: “ir até o cruzamento, dobrar à direita e andar três quarteirões”.

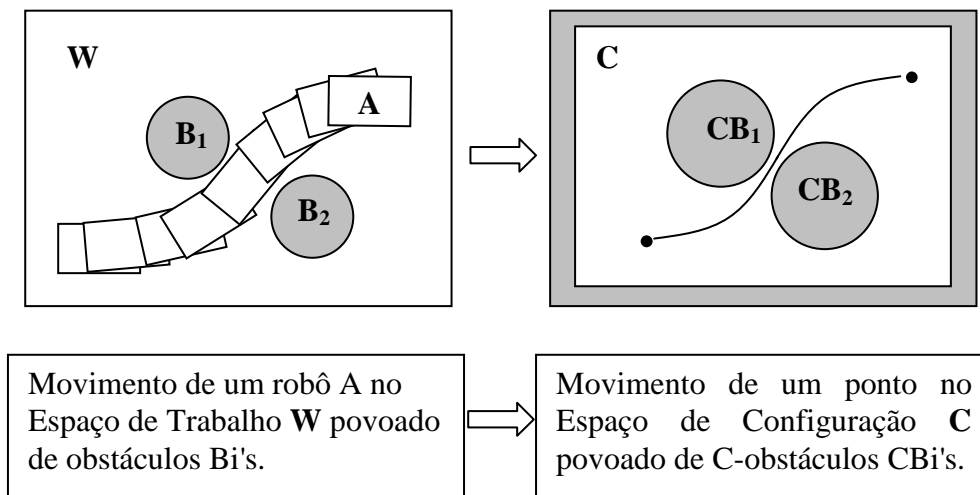
A representação do ambiente em mapas topológicos é mais abstrata do que sua representação em mapas métricos, de forma que é muito mais compacta, exigindo menor espaço de armazenamento. Por outro lado e pelo mesmo motivo, a construção de um mapa topológico a partir da informação sensorial do robô é um processo mais complexo, uma vez que envolve construir uma representação mais abstrata a partir de dados sensoriais brutos, extraíndo informações sobre os locais mais relevantes e característicos, assim como sobre as suas relações.



# ESPAÇO DE CONFIGURAÇÃO

## Espaço de Configuração:

Uma técnica muito utilizada no planejamento de tarefas é o mapeamento do problema originalmente especificado em espaço de trabalho ( $W$  - subconjunto de  $\mathbf{R}^2$  ou  $\mathbf{R}^3$ ) para espaço de configuração ( $C$ ). Neste espaço, o robô ( $A$ ), que no espaço de trabalho original é um objeto extenso, é transformado em um ponto. Por sua vez, os obstáculos presentes no espaço de trabalho ( $B$ ) são mapeados em uma nova representação em espaço de configuração (C-obstáculos –  $CB$ ). Assim, o problema de movimentar um robô de dimensões finitas em um espaço de trabalho povoado de obstáculos extensos é transformado no problema mais simples de movimentar um ponto em um espaço de configuração povoado de C-obstáculos.



Mapeamento para espaço de configuração.

Dados  $\{A\}$ , Referencial fixo em um robô  $A$ , e  $\{W\}$ , Referencial fixo no espaço de trabalho  $W$ , define-se a configuração  $q$  de  $A$  como uma especificação da localização (posição e orientação = pose) de  $\{A\}$  em relação a  $\{W\}$ . Por exemplo, para um robô manipulador, o vetor de variáveis de junta  $q$  pode ser uma representação da sua configuração. O espaço  $N$ -dimensional  $C$  de todas as possíveis configurações de  $A$  é denominado "Espaço de Configuração"  $\Rightarrow q \in C \forall q$ .  $A(q)$  é o subconjunto de  $W$  ocupado por  $A$  na configuração  $q$ .

A noção de Espaço de Configuração foi introduzida por Tomás Lozano-Pérez [Tomás Lozano-Pérez. Spatial Planning: A Configuration Space Approach. IEEE Transactions on Computers, Vol. c-32, No. 2, February 1983, pp. 108-120.].

Resumindo:

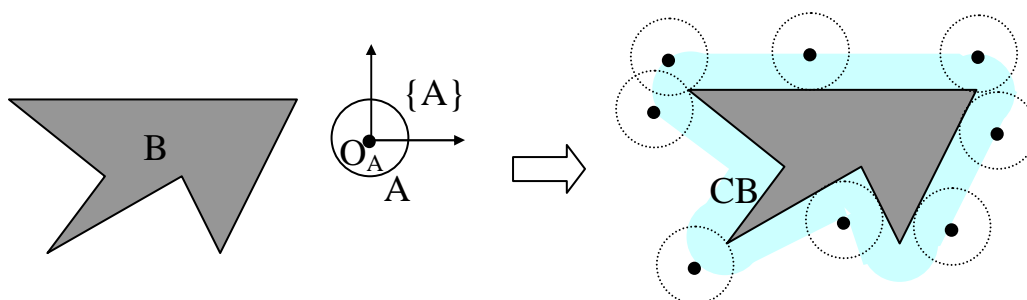
- **Espaço de Trabalho W:** é o espaço físico no qual o robô se movimenta.
- **Robô A:** conjunto de corpos rígidos interligados que pode movimentar-se dentro do Espaço de Trabalho.
- **Obstáculo no Espaço de Trabalho B:** região de **W** na qual não se pode posicionar qualquer ponto do Robô.
- **Configuração q:** Especificação da Posição e Orientação do robô.
- **Espaço de Configuração C:** é o conjunto de todas as possíveis configurações do robô.
- **Espaço de Configuração Livre C<sub>L</sub>:** é o conjunto de todas as possíveis configurações em que o robô não colide com os obstáculos B<sub>i</sub>'s.
- **Obstáculo em Espaço de Configuração CB (C-Obstáculo):** é o conjunto de todas as configurações em que o robô se superpõe parcial ou totalmente ao obstáculo. Um obstáculo **B** no espaço de trabalho **W** pode ser representado de forma equivalente por um C-obstáculo **CB** no espaço de configuração **C**.

### Obstáculos em Espaço de Configuração:

Dado um objeto A(q) modificando a sua configuração q ao movimentar-se em um espaço de trabalho W povoado de obstáculos B<sub>i</sub>'s, um obstáculo B<sub>i</sub> é mapeado em um C-obstáculo CB<sub>i</sub> no espaço de configuração definido como:

$$CB_i = \{q \in C / A(q) \cap B_i \neq \emptyset\}$$

Por exemplo, se um objeto circular se movimenta em um espaço planar povoado de obstáculos, os obstáculos originais devem ser aumentados do tamanho do raio.



C-Obstáculo para objeto circular movendo-se em  $\mathbf{R}^2$ .

### Caso – W Poligonal:

- Região poligonal convexa em  $\mathbf{R}^2$  = subconjunto de  $\mathbf{R}^2$  obtido pela interseção de um número finito de semiplanos fechados.
- Região poligonal  $P$  = União de regiões poligonais convexas.
- Polígono = Região poligonal homeomórfica com o disco unitário fechado.
- Se  $A$  e  $B_i$ 's são polígonos  $\Rightarrow CB$  representado como:
  - a) **Predicado:**  $CB(q)$  = verdadeiro se  $q \in CB$ .
  - b) **Forma explícita:** em termo de faces (curvas), eixos e vértices, incluindo as suas relações de adjacência.
- **Eixo  $E_i$ :** segmento de reta máximo no limite de  $P$ , que não intersecta o resto do limite de  $P$  exceto nos pontos extremos (vértices). Se inclui os vértices é **fechado**, se não, é **aberto**.
- **Linha Suporte  $L_i$ ,** contém o eixo  $E_i$ :  $h_i(x, y) = 0$ .
- **Semi-plano externo  $h_i^+$  de  $E_i$ :**  $h_i(x, y) \geq 0$
- **Semi-plano interno  $h_i^-$  de  $E_i$ :**  $h_i(x, y) \leq 0$
- **Normal externa de  $E_i$ :** vetor unitário normal a  $E_i$  apontando para o interior de  $h_i^+$ ,  $V_i$ .
- **Normal interna de  $E_i$ :** vetor unitário normal a  $E_i$  apontando para o interior de  $h_i^-$ ,  $-V_i$ .

### Representação de um Polígono:

- Lista de vértices enumerados em sentido anti-horário. Cada vértice representado pelas suas coordenadas.
- Conjunto de inequações:  $\cap h_i^-$ .

A complexidade de uma região poligonal é definida como o número de seus eixos e vértices.

## Tipos de Contatos:

### Representação do robô:

- Lista de  $n_A$  vértices  $a_i$ , com  $i = 1, \dots, n_A$ , enumerados em sentido anti-horário, aplicando aritmética módulo  $n_A$ , ( $a_{n_A+1} = a_1$ ).
- Eixo conectando  $a_i$  e  $a_{i+1} = \mathbf{E}_i^A$ .
- Normal externa ao eixo  $\mathbf{E}_i^A = \mathbf{V}_i^A(\mathbf{q})$ , função da orientação.

### Representação do Obstáculo:

- Lista de  $n_B$  vértices  $b_j$ , com  $j = 1, \dots, n_B$ , enumerados em sentido anti-horário, aplicando aritmética módulo  $n_B$ . ( $b_{n_B+1} = b_1$ ).
- Eixo conectando  $b_j$  e  $b_{j+1} = \mathbf{E}_j^B$ .
- Normal externa ao eixo  $\mathbf{E}_j^B = \mathbf{V}_j^B$ .

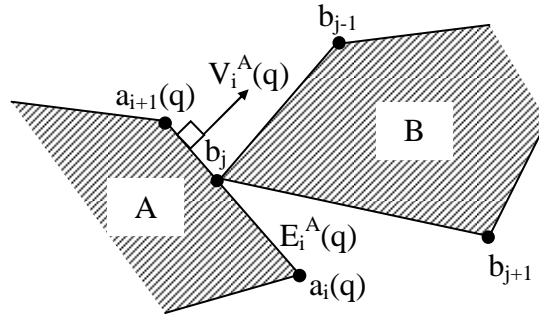
Quando **A** e **B** estão em contato:

$$A(q) \cap B \neq \emptyset \text{ e } \text{int}(A(q)) \cap \text{int}(B) = \emptyset.$$

Existem dois tipos de contato:

- Contato **tipo A**: quando um eixo  $\mathbf{E}_i^A$  de **A** contém um vértice  $b_j$  de **B**.
- Contato **tipo B**: quando um eixo  $\mathbf{E}_j^B$  de **B** contém um vértice  $a_i$  de **A**.
- Quando  $A(q) \cap B$  contém um vértice de **A** e um vértice de **B**, o contato é de ambos os tipos: **tipo A** e **tipo B**.

Contato tipo A: Eixo  $E_i^A$  de **A** contém um vértice  $b_j$  de **B**.



O contato entre  $E_i^A$  e  $b_j$ , sem superposição de **A** e **B**, é possível para uma certa faixa de orientações de **A**, tal que:

$$\begin{aligned} V_i^A(q) \cdot (b_{j-1} - b_j) &\geq 0 \\ V_i^A(q) \cdot (b_{j+1} - b_j) &\geq 0 \end{aligned}$$

$\Rightarrow$  Condição de Aplicação de Contato tipo A entre  $E_i^A$  e  $b_j$  (depende somente da orientação):

$$APL_{ij}^A(q) = [V_i^A(q) \cdot (b_{j-1} - b_j) \geq 0] \wedge [V_i^A(q) \cdot (b_{j+1} - b_j) \geq 0]$$

Deslocamento em contato tipo A  $\Rightarrow$  a configuração do robô está restrita a uma superfície em C (**C-Superfície tipo A**):

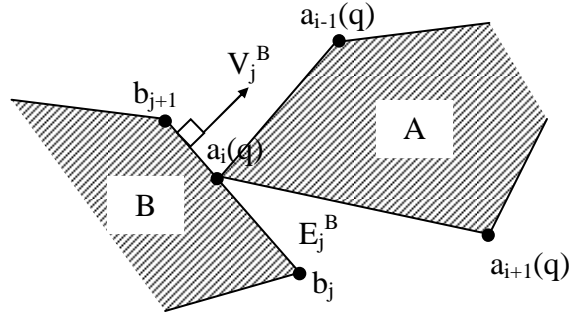
$$f_{ij}^A(q) = V_i^A \cdot (b_j - a_i(q)) = 0$$

A C-superfície tipo A separa C em dois semi-espacos, com CB contido completamente no semi-espaco  $f_{ij}^A \leq 0$ .

**C-Restrição tipo A:**

$$RESTR_{ij}^A(q) = APL_{ij}^A(q) \Rightarrow [f_{ij}^A \leq 0]$$

Contato tipo B: Eixo  $E_j^B$  de **B** contém um vértice  $a_i$  de **A**.



O contato entre  $E_j^B$  e  $a_i$ , sem superposição de **A** e **B**, é possível para uma certa faixa de orientações de **A**, tal que:

$$\begin{aligned} (a_{i-1}(q) - a_j(q)) \cdot V_j^B &\geq 0 \\ (a_{i+1}(q) - a_j(q)) \cdot V_j^B &\geq 0 \end{aligned}$$

$\Rightarrow$  Condição de Aplicação de Contato tipo B entre  $E_j^B$  e  $a_i$  (depende somente da orientação):

$$APL_{ij}^B(q) = [(a_{i-1}(q) - a_j(q)) \cdot V_j^B \geq 0] \wedge [(a_{i+1}(q) - a_j(q)) \cdot V_j^B \geq 0]$$

Deslocamento em contato tipo B  $\Rightarrow$  a configuração do robô está restrita a uma superfície em C (**C-Superfície tipo B**):

$$f_{ij}^B(q) = V_j^B \cdot (a_i(q) - b_j) = 0$$

A C-superfície tipo B separa C em dois semi-espacos, com CB contido completamente no semi-espaco  $f_{ij}^B \leq 0$ .

**C-Restrição tipo B**:

$$RESTR_{ij}^B(q) = APL_{ij}^B(q) \Rightarrow [f_{ij}^B \leq 0]$$

## Representação de C-Obstáculos Através de C-Restrições:

**Lema:** Dois polígonos convexos não se intersectam se e somente se um deles contém um eixo **E** tal que o outro polígono está contido no semiplano externo aberto de **E**.

**Teorema:** Dados dois polígonos convexos **A** e **B**, O C-Obstáculo  $CB = \{q \in C / A(q) \cap B \neq \emptyset\}$  é tal que,

$$q \in CB \Leftrightarrow CB(q) = (\bigwedge \text{RESTR}_{ij}^A(q)) \wedge (\bigwedge \text{RESTR}_{ij}^B(q))$$

Assim, avaliando sucessivamente as C-Restrições, pode-se verificar se a configuração  $q$  está ou não no C-Obstáculo:

- Se todas são verdadeiras,  $q \in CB$ .
- Se ao menos uma é falsa,  $q \notin CB$ . Assim:  $q \in C_L \Leftrightarrow \neg CB(q)$
- No pior caso, o número de C-restrições a serem avaliadas (e, conseqüentemente, a complexidade computacional) é  $2.n_A.n_B$ .

A configuração  $q$  está no limite de  $CB$  se e somente se:

- $\exists i, j$  tais que  $APL_{ij}^A(q)$  (ou  $APL_{ij}^B(q)$ ) é verdadeira e  $f_{ij}^A(q) = 0$  (ou  $f_{ij}^B(q) = 0$ ),
- $\forall i', j'$  tais que  $APL_{i'j'}^A(q)$  é verdadeira,  $f_{i'j'}^A(q) \leq 0$ .
- $\forall i', j'$  tais que  $APL_{i'j'}^B(q)$  é verdadeira,  $f_{i'j'}^B(q) \leq 0$ .

Se **A** e **B** são não convexos, podem ser decompostos em um conjunto de polígonos convexos:  $A = \cup A_k$  e  $B = \cup B_l$ . Definindo:

$$CB_{kl} = \{q \in C / A_k(q) \cap B_l \neq \emptyset\}$$

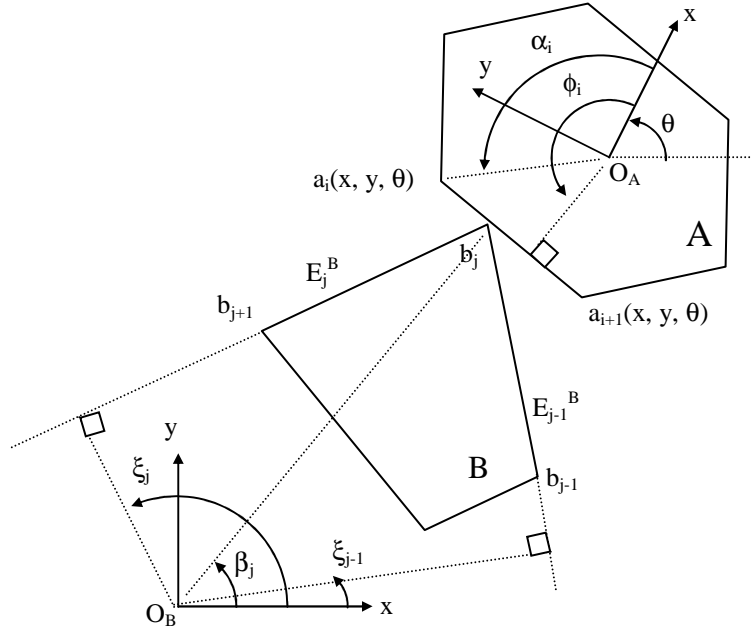
$$\Rightarrow CB = \cup CB_{kl}$$

$$\Rightarrow CB(q) = \bigvee CB_{kl}(q)$$

Assim, o Espaço de Configuração Livre é:  $q \in C_L \Leftrightarrow \neg CB(q)$ .

## Parametrização das C-Restrições: $q = (x, y, \theta)$

### Contato tipo A:



$$V_i^A = (\cos(\phi_i + \theta), \sin(\phi_i + \theta))$$

$$V_j^B = (\cos \xi_j, \sin \xi_j)$$

$$a_i(x, y, \theta) = (x + \|a_i\| \cdot \cos(\alpha_i + \theta), y + \|a_i\| \cdot \sin(\alpha_i + \theta))$$

$$b_j = (\|b_j\| \cdot \cos(\beta_j), \|b_j\| \cdot \sin(\beta_j))$$

- Condição de Aplicabilidade:

$$APL_{ij}^A(q) = [V_i^A(q) \cdot (b_{j-1} - b_j) \geq 0] \wedge [V_i^A(q) \cdot (b_{j+1} - b_j) \geq 0]$$

$$\Rightarrow APL_{ij}^A(q) = \theta \in [\xi_{j-1} - \phi_i - \pi, \xi_j - \phi_i + \pi] \pmod{2\pi}$$

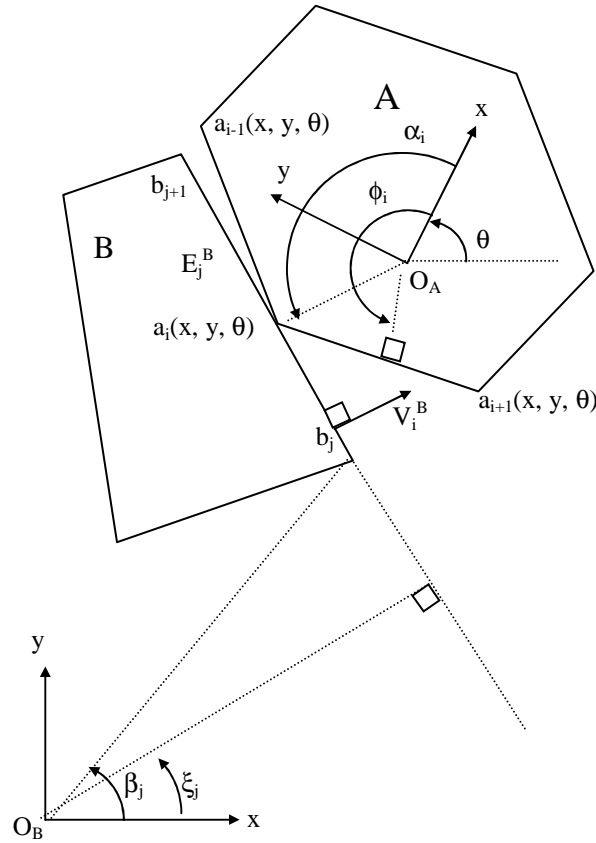
- Inequação do C-Obstáculo:

$$f_{ij}^A(q) = V_i^A \cdot (b_j - a_i(q)) \leq 0$$

$$\Rightarrow -x \cdot \cos(\phi_i + \theta) - y \cdot \sin(\phi_i + \theta) + \|b_j\| \cdot \cos(\phi_i + \theta - \beta_j) - \|a_i\| \cdot \cos(\phi_i - \alpha_i) \leq 0$$



### Contato tipo B:



$$V_i^A = (\cos(\phi_i + \theta), \sin(\phi_i + \theta))$$

$$V_j^B = (\cos \xi_j, \sin \xi_j)$$

$$a_i(x, y, \theta) = (x + \|a_i\| \cdot \cos(\alpha_i + \theta), y + \|a_i\| \cdot \sin(\alpha_i + \theta))$$

$$b_j = (\|b_j\| \cdot \cos(\beta_j), \|b_j\| \cdot \sin(\beta_j))$$

- Condição de Aplicabilidade:

$$APL_{ij}^B(q) = [(a_{i-1}(x, y, \theta) - a_i(x, y, \theta)) \cdot V_j^B \geq 0] \wedge [(a_{i+1}(x, y, \theta) - a_i(x, y, \theta)) \cdot V_j^B \geq 0]$$

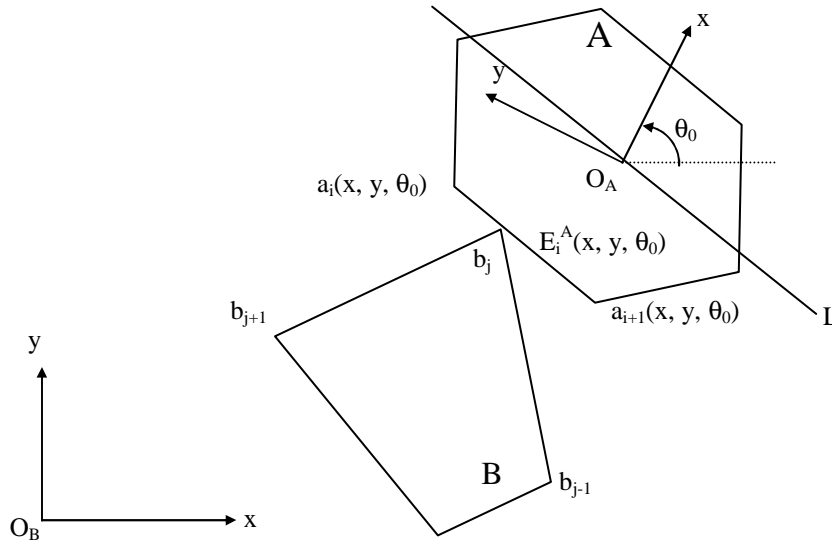
$$\Rightarrow APL_{ij}^B(q) = \theta \in [\xi_j - \phi_i - \pi, \xi_j - \phi_{i-1} - \pi] \pmod{2\pi}$$

- Inequação do C-Obstáculo:

$$f_{ij}^B(q) = V_j^B \cdot (a_i(x, y, \theta) - b_j) \leq 0$$

$$\Rightarrow x \cdot \cos(\xi_j) + y \cdot \sin(\xi_j) + \|a_i\| \cdot \cos(\alpha_i + \theta - \xi_j) - \|b_j\| \cdot \cos(\beta_j - \xi_j) \leq 0$$

## Interpretação Geométrica do Contato tipo A:

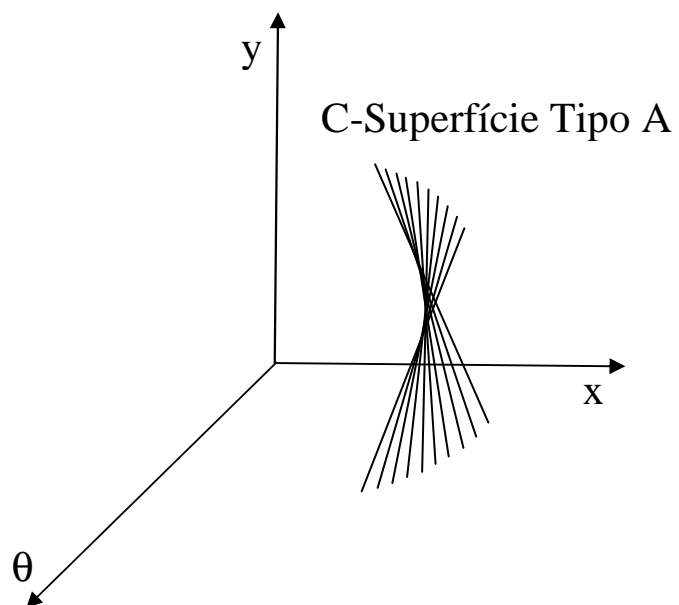


Um deslocamento em contato tipo A, mantendo uma orientação constante  $\theta_0$ , corresponde à equação de uma reta  $L$  em  $\mathbf{R}^2$  paralela ao eixo  $E_i^A(x, y, \theta_0)$ :

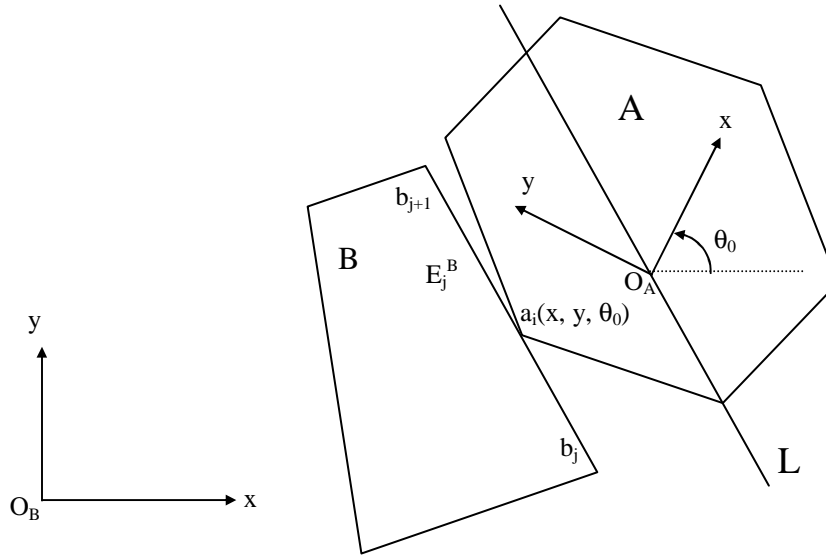
$$f_{ij}^A(x, y, \theta_0) = 0$$

$$\Rightarrow -x \cdot \cos(\phi_i + \theta_0) - y \cdot \sin(\phi_i + \theta_0) + \|b_j\| \cdot \cos(\phi_i + \theta_0 - \beta_j) - \|a_i\| \cdot \cos(\phi_i - \alpha_i) = 0$$

- Quando o robô gira um ângulo  $d\theta$ , a reta  $L$  gira o mesmo ângulo tendo o vértice  $b_j$  como centro de rotação.
- As faces do C-Obstáculo correspondentes a deslocamentos em contato tipo A estão contidas em C-Superfícies helicoidais (C-Superfícies tipo A).



## Interpretação Geométrica do Contato tipo B:

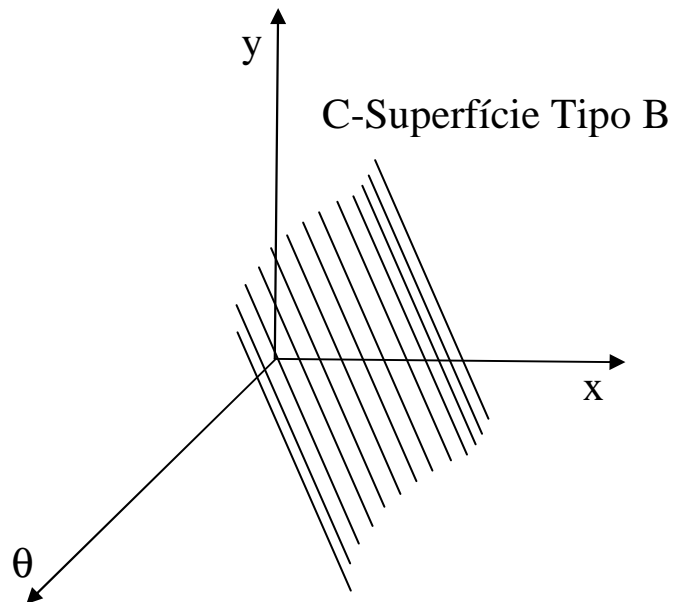


Um deslocamento em contato tipo B, mantendo uma orientação constante  $\theta_0$ , corresponde à equação de uma reta L em  $\mathbf{R}^2$  paralela ao eixo  $E_j^B$ :

$$f_{ij}^B(x, y, \theta_0) = 0$$

$$\Rightarrow x \cdot \cos(\xi_j) + y \cdot \sin(\xi_j) + \|a_i\| \cdot \cos(\alpha_i + \theta_0 - \xi_j) - \|b_j\| \cdot \cos(\beta_j - \xi_j) = 0$$

- Quando o robô gira um ângulo  $d\theta$ , a reta L sofre uma translação, modificando a sua distância em relação ao eixo  $E_j^B$ , mas mantendo-se paralela ao mesmo.
- As faces do C-Obstáculo correspondentes a deslocamentos em contato tipo B estão contidas em C-Superfícies curvas em uma única dimensão (C-Superfícies tipo B), semelhantes à superfície de um tobogã.



Ver exemplo de vídeos:

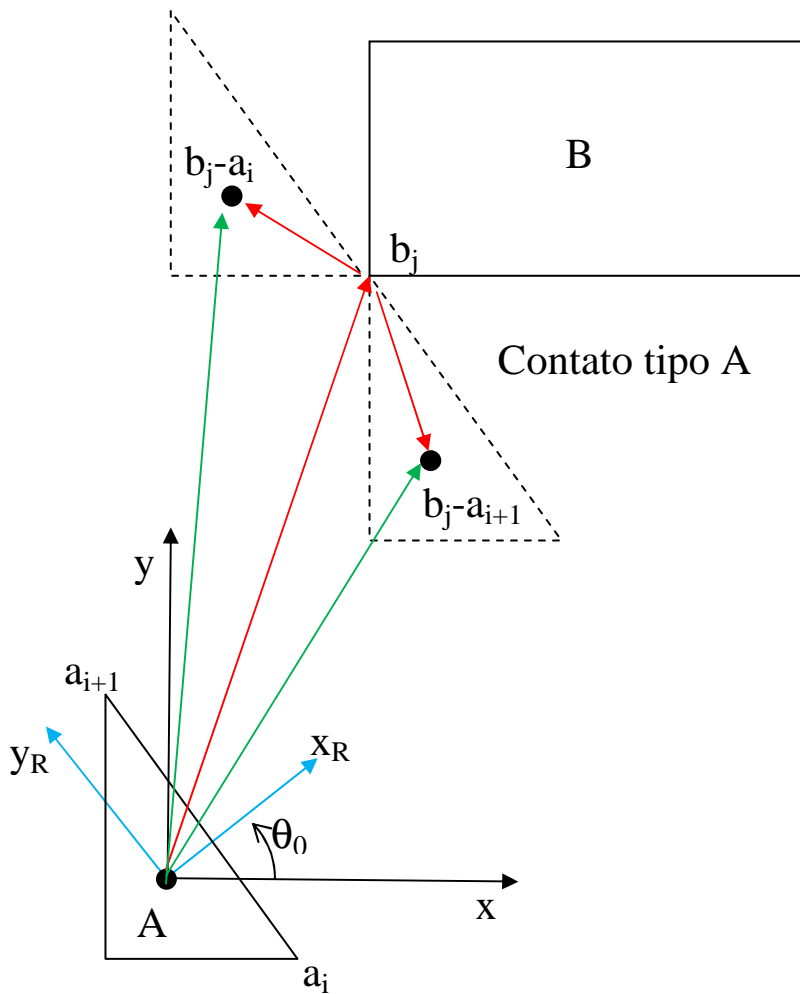
<http://www.youtube.com/watch?v=BsCcy0vYerw>

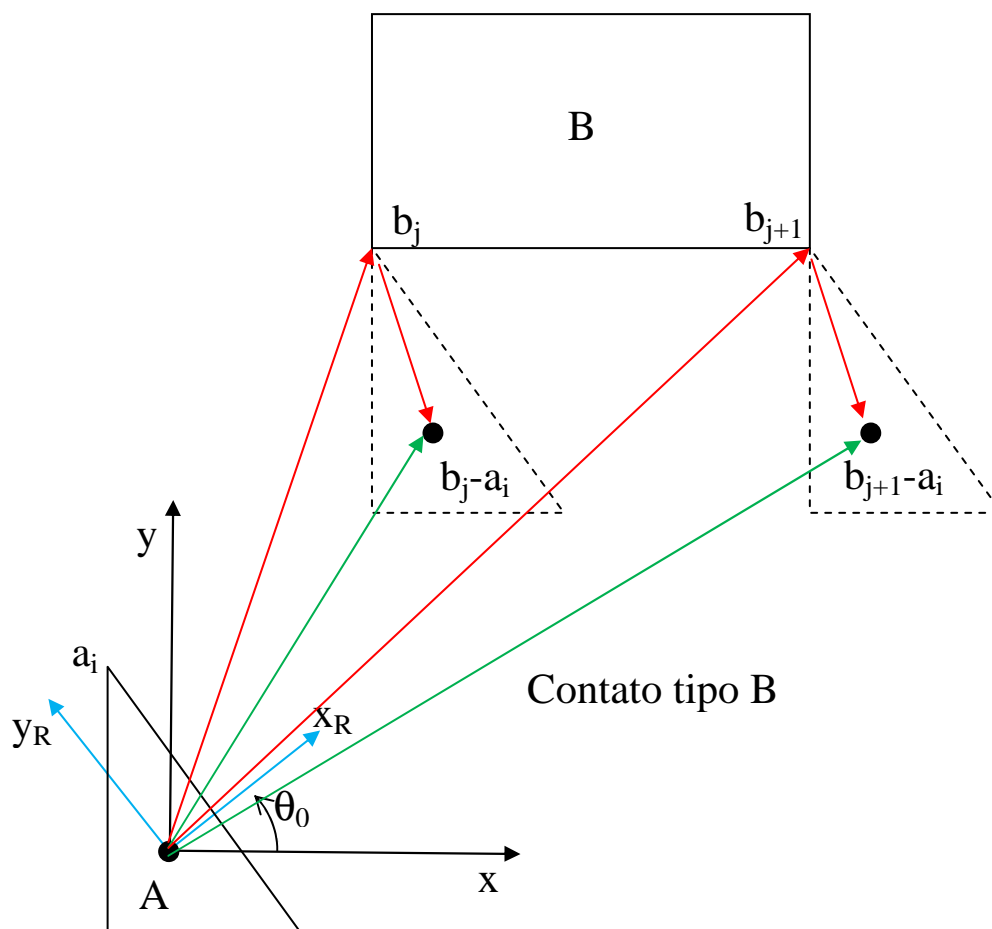
<http://www.youtube.com/watch?v=SBFwgR4K1Gk>

### Representação dos Limites de CB (Caso Translacional):

Para uma orientação fixa  $\theta_0$ , com  $q_0 = (0,0,\theta_0)$ :

- Se  $APL_{ij}^A(\theta_0)$  é verdadeira,  $(b_j - a_i(q_0))$  e  $(b_{j+1} - a_{i+1}(q_0))$  são vértices de  $CB(\theta_0)$ .
- Se  $APL_{ij}^B(\theta_0)$  é verdadeira,  $(b_j - a_i(q_0))$  e  $(b_{j+1} - a_i(q_0))$  são vértices de  $CB(\theta_0)$ .





Procedimento para construir  $CB(\theta_0)$ :

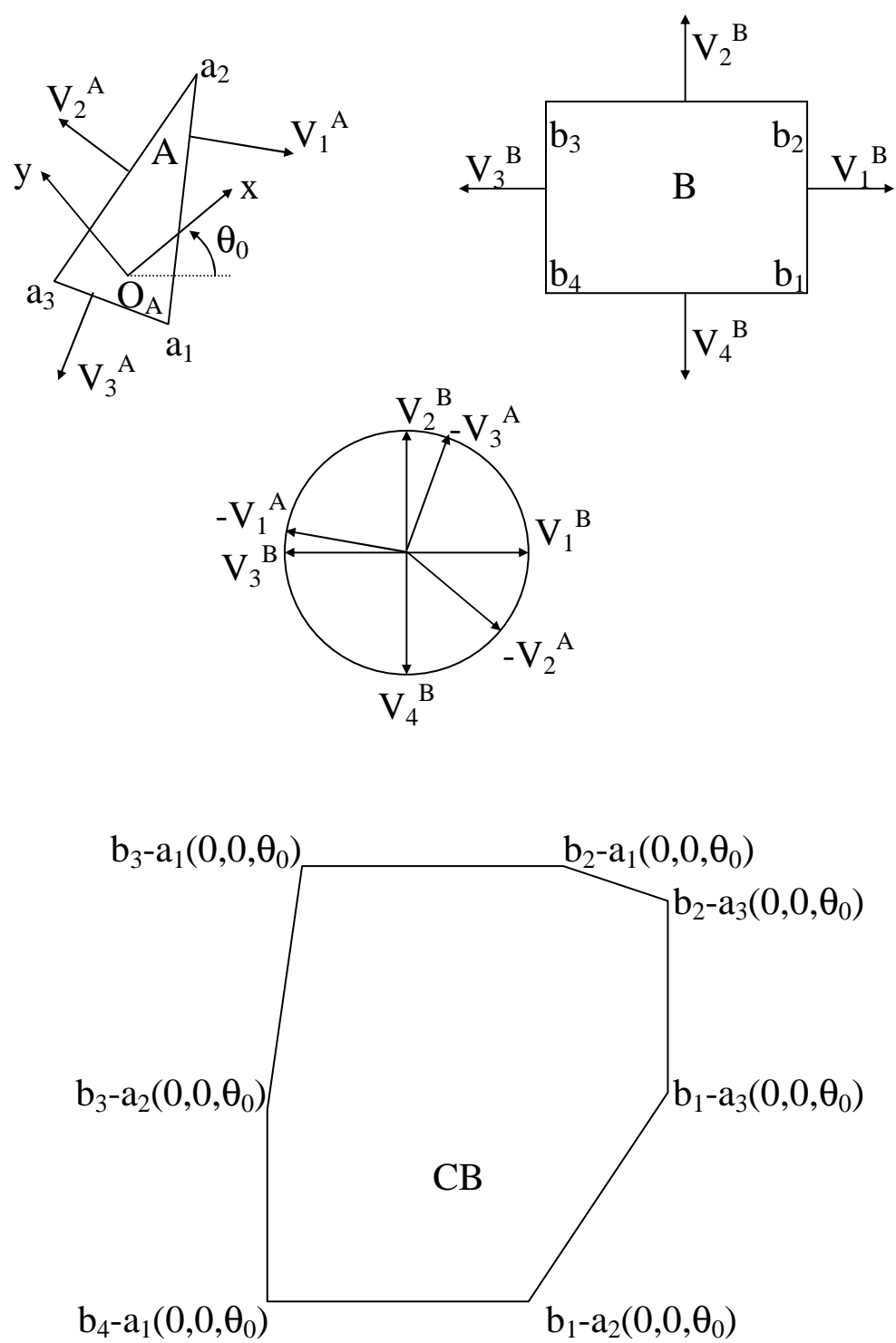
1. Fixar os vetores normais  $-V_i^A$  (com  $i=1,\dots,n_A$ ) e  $V_j^B$  (com  $j=1,\dots,n_B$ ) no círculo unitário  $S^1$ .
2. Varrer o círculo unitário em sentido anti-horário e verificar a condição de aplicabilidade de acordo com os Vetores Normais:
  - $APL_{ij}^A(\theta_0)$  é verdadeira se  $-V_i^A$  está entre  $V_{j-1}^B$  e  $V_j^B$ .
  - $APL_{ij}^B(\theta_0)$  é verdadeira se  $V_j^B$  está entre  $-V_{i-1}^A$  e  $-V_i^A$ .
3. De acordo com a condição de aplicabilidade, criar os  $n_A+n_B$  vértices de  $CB(\theta_0)$ :
  - Se  $-V_i^A$  está entre  $V_{j-1}^B$  e  $V_j^B$ , criar o vértice  $(b_j-a_i(q_0))$ .
  - Se  $V_j^B$  está entre  $-V_{i-1}^A$  e  $-V_i^A$ , criar o vértice  $(b_j-a_i(q_0))$ .

Observações:

- Para uma orientação crítica  $\theta_0$ , tal que  $E_i^A(q_0)$  se desloca paralelamente em contato com  $E_j^B$ , temos  $-V_i^A = V_j^B$ . Os pontos  $(b_j-a_i)$ ,  $(b_{j+1},a_i)$ ,  $(b_j-a_{i+1})$  e  $(b_{j+1},a_{i+1})$  são co-lineares. Os pontos  $(b_{j+1},a_i)$  e  $(b_j-a_{i+1})$  não são vértices de  $CB(\theta_0)$ .
- A complexidade do algoritmo é de ordem  $O(n_A+n_B)$ .
- Para  $A$  e  $B$  não convexos, decompor em componentes convexas  $A_i$  e  $B_j$ . Computar  $CB_{ij}$  para cada  $(A_i, B_j)$ .  $CB = \cup CB_{ij}$ .



Exemplo:



A Seção transversal  $CB(\theta_0)$  é a região de  $\mathbf{R}^2$  definida por:

$$(x, y) \in CB(\theta_0) \Leftrightarrow (\bigwedge \text{RESTR}_{ij}^A(x, y, \theta_0)) \wedge (\bigwedge \text{RESTR}_{ij}^B(x, y, \theta_0))$$

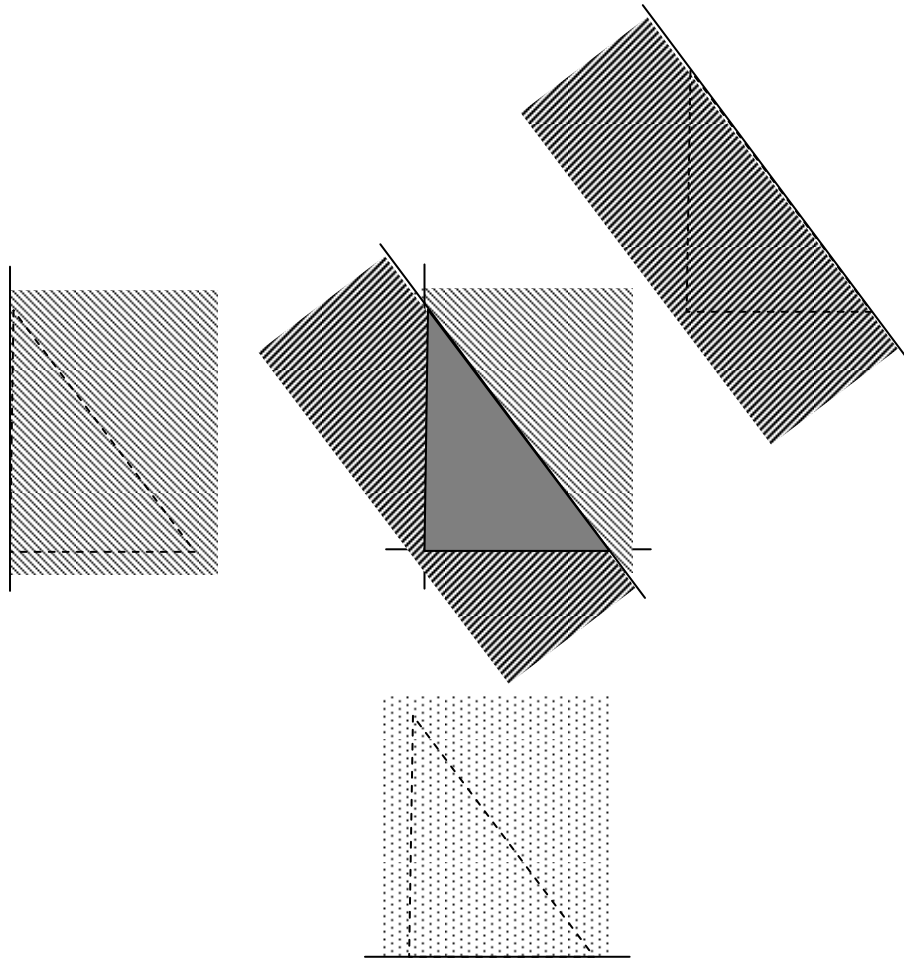
- $CB(\theta_0)$  é a interseção de um número finito de semi-planos fechados limitados por retas:  $f_{ij}^A(x, y, \theta_0) = 0$  ou  $f_{ij}^B(x, y, \theta_0) = 0$ .

$$f_{ij}^A(x, y, \theta_0) = 0$$

$$\Rightarrow -x \cdot \cos(\phi_i + \theta_0) - y \cdot \sin(\phi_i + \theta_0) + \|b_j\| \cdot \cos(\phi_i + \theta_0 - \beta_j) - \|a_i\| \cdot \cos(\phi_i - \alpha_i) = 0$$

$$f_{ij}^B(x, y, \theta_0) = 0$$

$$\Rightarrow x \cdot \cos(\xi_j) + y \cdot \sin(\xi_j) + \|a_i\| \cdot \cos(\alpha_i + \theta_0 - \xi_j) - \|b_j\| \cdot \cos(\beta_j - \xi_j) = 0$$

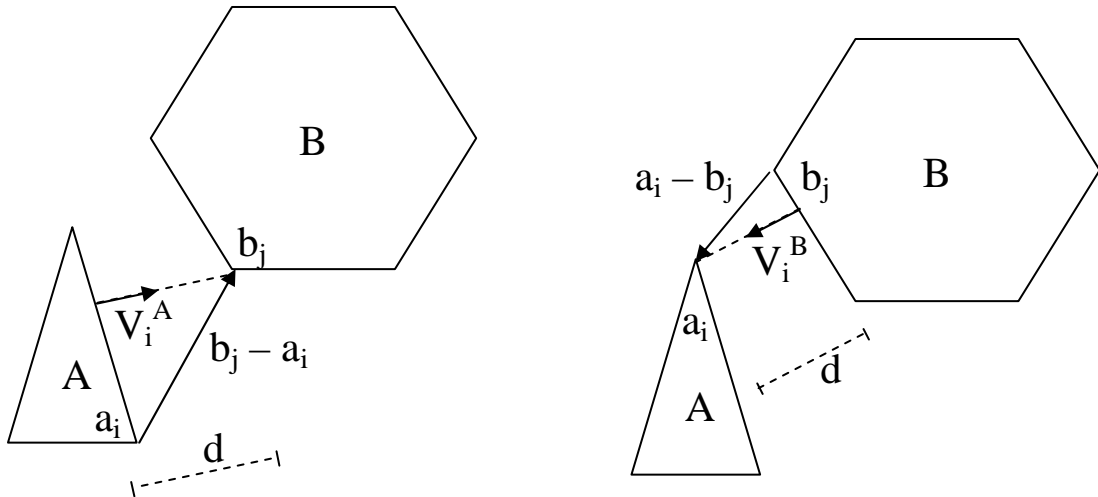


- A distância euclidiana de um ponto  $(x, y) \in \mathbf{R}^2$  a estas retas é:

$$|f_{ij}^A(\mathbf{x}, \mathbf{y}, \theta_0)| = |\mathbf{V}_i^A(\theta_0) \cdot (\mathbf{b}_j - \mathbf{a}_i(\mathbf{x}, \mathbf{y}, \theta_0))|$$

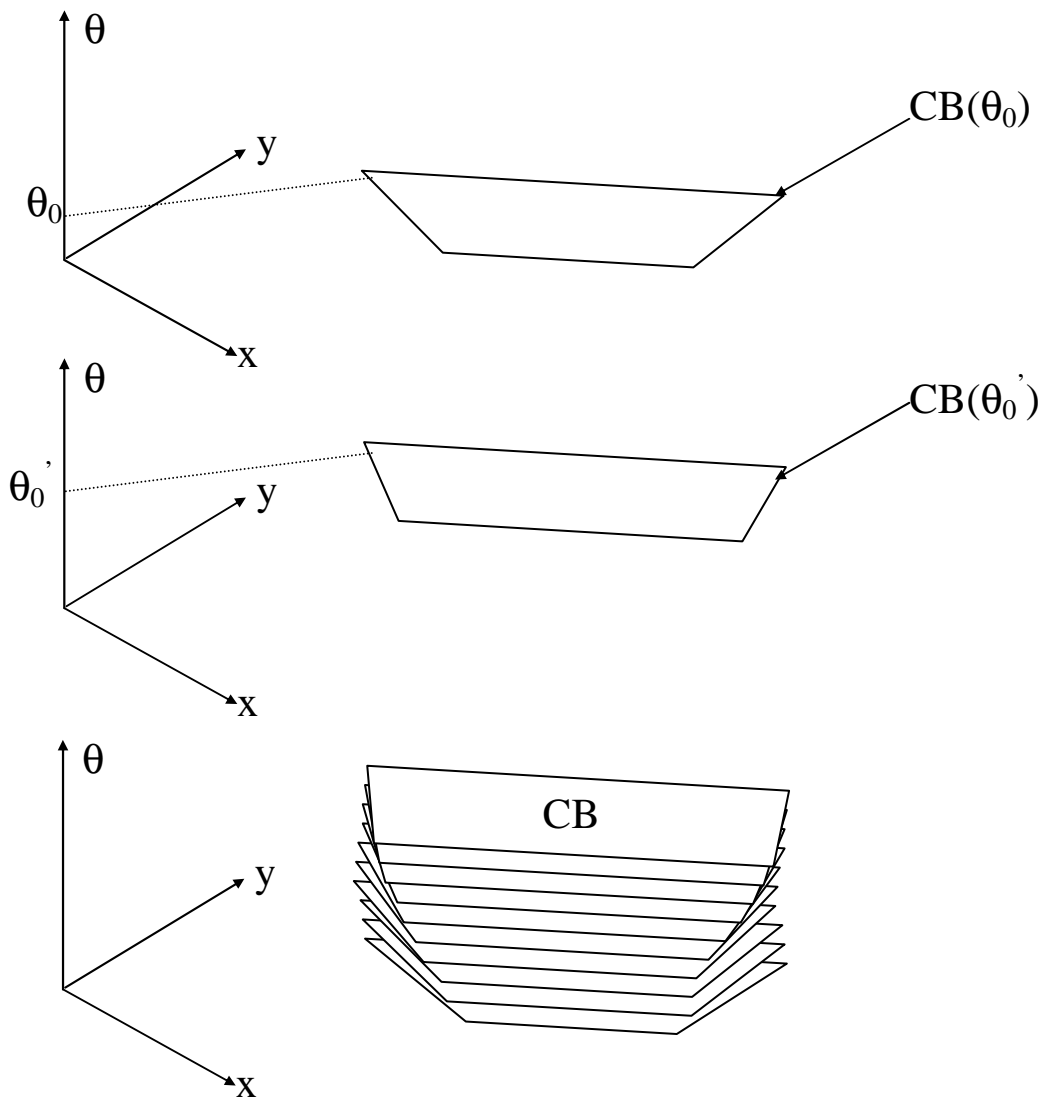
$$|f_{ij}^B(\mathbf{x}, \mathbf{y}, \theta_0)| = |\mathbf{V}_j^B \cdot (\mathbf{a}_i(\mathbf{x}, \mathbf{y}, \theta_0) - \mathbf{b}_j)|$$

Estes valores podem ser usados para calcular a distância da configuração  $(x, y, \theta_0)$  ao C-Obstáculo no subespaço  $\theta = \theta_0$ . A menor de todas as distâncias  $(E_i^A, b_j)$ ,  $(E_j^B, a_i)$  é a distância entre o robô e o obstáculo nessa configuração.

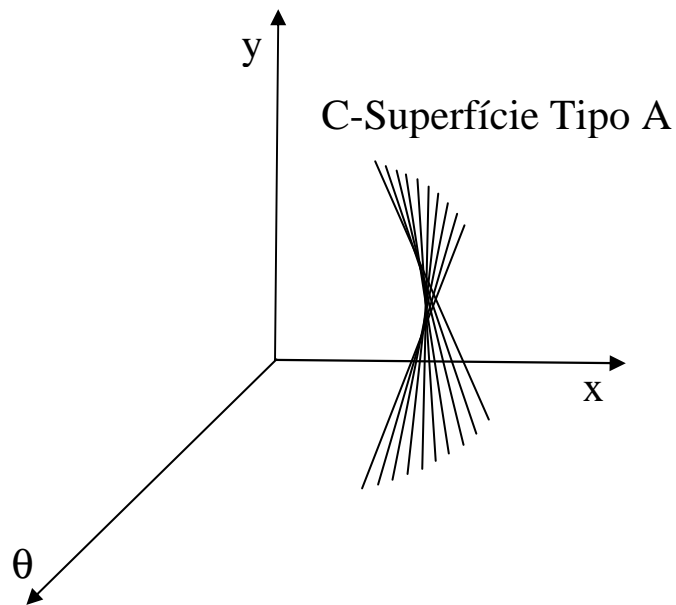


C-Obstáculo em  $\mathbf{R}^2 \times [0, 2\pi)$  para A e B polígonos convexos:

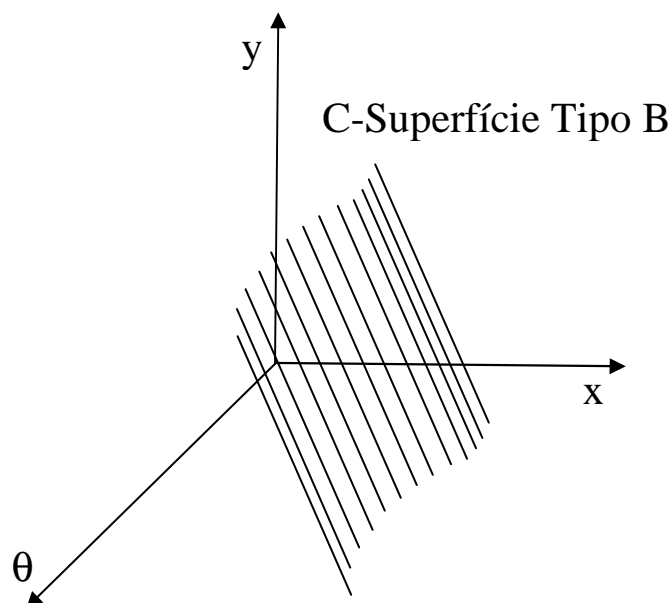
- Para cada valor  $\theta_0$  de  $\theta$ , a seção transversal de CB correspondente é um polígono convexo que representa o C-obstáculo  $CB(\theta_0)$  em  $\mathbf{R}^2$  quando A se desloca com orientação fixa  $\theta_0$ .



- Os limites de CB são as faces, (retalhos de C-Superfícies), correspondentes a contatos do tipo A ou do tipo B.
- Contato tipo A  $\Rightarrow$  Face = retalho de helicóide (gerada por uma reta em rotação e translação, paralela ao plano xy).



- Contato tipo B  $\Rightarrow$  Face = retalho de superfície curva em uma única dimensão (gerada por uma reta em translação, paralela ao plano xy).



### Teste de Penetração de Polígonos:

O planejamento de movimento de um robô em um espaço povoado de obstáculos envolve a determinação de um caminho livre de colisões. No caso em que é possível modelar o problema em um espaço de trabalho poligonal, para detectar a colisão de objetos modelados como polígonos movimentando-se sobre um plano, é necessário testar se estes possuem interseção não nula, ou seja se há interpenetração entre eles. Para realizar este tipo de teste, uma métrica muito utilizada é a distância de um ponto  $(x_0, y_0)$  a uma reta definida por dois pontos  $(x_1, y_1)$  e  $(x_2, y_2)$ . Dada a reta no plano  $xy$  passando por estes dois pontos:

$$a.x + b.y + c = 0$$

os coeficientes  $a$ ,  $b$  e  $c$  são:

$$a = y_1 - y_2$$

$$b = x_2 - x_1$$

$$c = x_1.y_2 - x_2.y_1$$

a distancia  $d(x_0, y_0)$  do ponto  $(x_0, y_0)$  à reta é dada por:

$$d(x_0, y_0) = (a.x_0 + b.y_0 + c)/(a^2 + b^2)^{1/2}$$

Note que  $d(x_0, y_0)$  é uma função que pode admitir valores positivos, negativos ou nulos. Se  $d(x_0, y_0) = 0$ , o ponto  $(x_0, y_0)$  está sobre a reta. Se  $d(x_0, y_0) > 0$ , o ponto está no semiplano à esquerda da reta (de acordo com o sentido do vetor que vai de  $(x_1, y_1)$  ao ponto  $(x_2, y_2)$ ). Se  $d(x_0, y_0) < 0$ , o ponto está no semiplano à direita da reta.

Assim, dado um polígono com  $n_p$  vértices  $P = \{p_k = (x_k, y_k) : 1 \leq k \leq n_p+1\}$ , tal que o vértice  $p_{n_p+1} = p_1$ , o algoritmo a seguir permite determinar se um dado ponto  $(x_0, y_0)$  está ou não dentro do polígono  $P$  e, em caso afirmativo, o grau de penetração do ponto em  $P$  (distância ao lado mais próximo).

Algoritmo de Penetração de um Ponto  $(x_0, y_0)$  em um polígono  $P$ :

1. Inicializar:  $k = 1$ ,  $d_0 = \text{valor real máximo} > \text{perímetro de } P$ .
2. Computar:  
$$a = y_k - y_{k+1}$$
$$b = x_{k+1} - x_k$$
$$c = x_k.y_{k+1} - x_{k+1}.y_k$$
$$d = (a.x_0 + b.y_0 + c)/(a^2 + b^2)^{1/2}$$
3. Se  $d < d_0$ , então faça  $d_0 = d$
4. Faça  $k = k+1$ . Se  $k \leq n_p$ , voltar ao passo 2.

A saída do algoritmo é a distância de penetração  $d_0$ . Se  $d_0 > 0$ , o ponto  $(x_0, y_0)$  está dentro do polígono  $P$  e  $d_0$  é uma medida da penetração do ponto dentro do polígono.

Para testar se dois polígonos convexos  $A$  e  $B$  possuem intersecção não nula, devemos testar qualquer uma das seguintes condições é satisfeita:

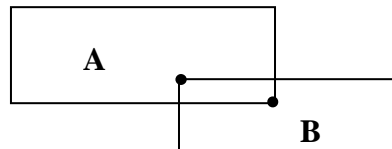
- Verificar se pelo menos um vértice  $a_i$  de A está dentro de B.
- Verificar se pelo menos um vértice  $b_k$  de B está dentro de A.
- Verificar se um pelo menos um lado de A, ( $L_i^A$ , definido pelos vértices  $a_i$  e  $a_{i+1}$ ), cruzar com um lado de B, ( $L_k^B$ , definido pelos vértices  $b_k$  e  $b_{k+1}$ ).

As duas primeiras condições verificam a situação em que um vértice de um dos polígonos penetrou dentro do outro polígono. A terceira condição verifica a situação em que, mesmo sem existir algum vértice de um polígono dentro do outro, há cruzamento de lados, como por exemplo, no caso de dois retângulos superpostos formando uma cruz.

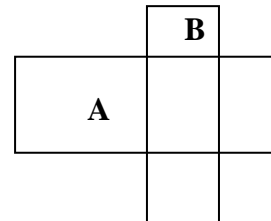
Para testar as duas primeiras condições, devemos executar o algoritmo de penetração para todos os vértices do polígono A em relação ao polígono B e para todos os vértices do polígono B em relação ao polígono A.

Para testar a terceira condição, devemos verificar se cada um dos lados de A intersecta qualquer um dos lados de B. Dados dois lados quaisquer  $L_i^A$  e  $L_k^B$ , dos polígonos A e B, respectivamente, os mesmos se intersectam se os vértices  $a_i$  e  $a_{i+1}$  estão em lados opostos de  $L_k^B$  e, simultaneamente, os vértices  $b_k$  e  $b_{k+1}$  estão em lados opostos de  $L_i^A$ . Ou seja:

Para todo par de lados ( $L_i^A, L_k^B$ ), com  $i = 1, \dots, n_A$  e  $k = 1, \dots, n_B$ , devemos testar se as distâncias  $d(a_i)$  e  $d(a_{i+1})$  a  $L_k^B$  possuem sinais opostos e se as distâncias  $d(b_k)$  e  $d(b_{k+1})$  a  $L_i^A$  também possuem sinais opostos.



(a)



(b)

Situações de teste de colisão.

a) Vértice de A dentro de B ou vértice de B dentro de A; b) Cruzamento de lados.

Na prática, em uma aplicação de planejamento de movimento em que o caminho é constituído de configurações sobre uma curva contínua, supondo que a mesma seja percorrida em pequenos deslocamentos discretos e o teste de colisão seja feito para essa sequência de deslocamentos, dificilmente aparecerá o terceiro caso, pois para chegar a ele, provavelmente deve-se passar antes por algum dos dois primeiros casos. Assim, aplicando o teste de penetração para todos os vértices seria suficiente. Por outro lado, o teste de cruzamento de lados incorpora os dois primeiros casos, uma vez que quando um vértice está dentro de um polígono, os dois lados dos quais o mesmo é extremo cruzarão algum dos lados desse polígono (Devem ser tratadas ainda os casos excepcionais quando um lado de um polígono cruza o limite de outro polígono exatamente através de um vértice).