



## **SISTEMA DE PLANEJAMENTO DE TRAJETÓRIAS PARA ROBÔS MÓVEIS COM BASE EM VISÃO GLOBAL**

**Waldir L. Roque e Dionísio Doering**

Instituto de Matemática, Universidade Federal do Rio Grande do Sul, 91509-900 Porto Alegre, RS.  
roque@mat.ufrgs.br.

***Resumo.** Um dos principais problemas para robôs móveis autônomos está relacionado ao planejamento de trajetórias, livres de colisão, entre duas configurações (posição e orientação) no espaço de trabalho. O Diagrama de Voronoi Generalizado (DVG) é uma importante técnica para geração do mapa de trajetórias com máxima desobstrução, além do mesmo ser conexo para um ambiente de trabalho fechado. No entanto, a construção do mapa de trajetórias com o DVG requer um conhecimento prévio da estrutura geométrica do espaço de trabalho. Uma forma de contornar esta dificuldade é utilizando-se visão computacional global, a qual identificará a estrutura geométrica do espaço de trabalho. Neste artigo, discutimos o desenvolvimento de um sistema de planejamento de trajetórias baseado no DVG integrado a um módulo de visão global (MVG). Os objetos presentes no cenário, inclusive o robô, são segmentados e suas características geométricas identificadas. Os dados geométricos são lidos pelo módulo planejador de trajetórias, o qual gera as trajetórias de menor comprimento e os atalhos de visibilidade entre as configurações inicial e alvo. O MVG permite resolver o problema de localização do robô. A configuração alvo é especificada pelo usuário diretamente na área de trabalho do sistema. Alguns testes e validações foram conduzidos com a utilização do mini-robô Khepera e os resultados são apresentados.*

***Palavras-chave:** Robótica móvel, planejamento de trajetórias, visão computacional, diagrama de Voronoi.*

### **1. INTRODUÇÃO**

Nos últimos anos a robótica móvel autônoma tem tido um desenvolvimento extraordinário e as suas aplicações estão surgindo em diversas áreas com uma maior frequência (World Robotics, 2001). No entanto, por diversas razões, a robótica móvel ainda não alcançou o estágio de aplicabilidade esperado, sendo a complexidade dos problemas envolvidos uma dessas principais razões. Para uma pessoa normal a tarefa de sair do sofá da sala para pegar um copo com água na geladeira é algo trivial. Porém, visto em termos de uma tarefa a ser executada por um robô autônomo é algo bastante complexo, passando por diversos níveis de ações. Dentre as várias ações envolvidas está o planejamento do caminho a ser percorrido da posição inicial do indivíduo até alcançar o alvo, que neste caso seria a geladeira, na cozinha. Claramente a pessoa ao chegar à geladeira deseja estar posicionado de forma que a sua frente esteja dirigida para o alvo. Vemos que não se trata apenas de sair do sofá e chegar na posição em frente à geladeira, mas a orientação da pessoa é também fundamental. Além disso, o deslocamento do indivíduo requer cuidados para que não haja colisões com os objetos presentes ao longo do percurso.

A execução da tarefa acima por um robô requer o planejamento detalhado de possíveis trajetórias para alcançar o alvo. Dentre as possíveis trajetórias, o robô deve ser capaz de escolher aquela que minimiza o custo de execução da tarefa. Para a construção do mapa de trajetórias se faz necessário o conhecimento do ambiente em que o robô está inserido. A forma de se adquirir esta

informação pode ser via diversos tipos de sensores, como no caso de uma pessoa que utiliza principalmente a visão para construir o mapa do ambiente (sala e cozinha).

Várias técnicas vêm sendo empregadas para o planejamento de trajetórias (Latombe, 1991; Roque, 1996) de robôs móveis autônomos. Dentre estas, a técnica que permite a construção do mapa de trajetórias a partir da geração do Diagrama de Voronoi Generalizado (DVG) tem demonstrado um grande potencial de aplicabilidade. Uma das vantagens desta técnica é que as trajetórias geradas correspondem a caminhos com máxima desobstrução com relação aos obstáculos presentes no ambiente e que os caminhos são conexos para ambientes fechados.

A geração do mapa de trajetórias com base no DVG requer o conhecimento da geometria do ambiente de trabalho. Em geral, este conhecimento é obtido a priori, podendo ser apenas local ou global. Em ambos os casos é necessário a utilização de sensores para a aquisição da informação sobre a geometria do ambiente. Fazendo um paralelo com o problema da pessoa acima, a mesma utiliza o seu sistema de visão para formar a priori um mapa do ambiente e em seguida planeja a melhor trajetória para navegar do sofá até a geladeira. Neste caso, o sistema sensorial do indivíduo, a visão, está diretamente acoplado ao próprio indivíduo. A construção do mapa do ambiente é neste caso local. No caso do robô, um sistema de visão computacional pode ser acoplado diretamente ao robô, estando, portanto, embarcado. Por outro lado, podemos dispor de um sistema de visão externa ao robô, o qual é capaz de visualizar todo o ambiente de trabalho, constituindo um sistema de visão global.

A construção do DVG local tem sido investigada por Choset (Choset, 1995), utilizando sensores de sonar, e mais recentemente por Mahkovic (Markovic, 2000; Blanco, 2000) utilizando sensores a laser. Por outro lado, alguns trabalhos foram apresentados com propostas de construção do diagrama de Voronoi diretamente a partir de imagens digitais (Arcelli, 1986; Parui, 1993), incluindo um algoritmo paralelo (Sudha, 1999) e a arquitetura de um VLSI destinado a construção do diagrama de Voronoi com base em imagens. Porém, tais algoritmos não foram aplicados ao problema de planejamento de trajetórias de robôs móveis.

Neste trabalho, vamos discutir o desenvolvimento de um sistema de planejamento de trajetórias fundamentado na técnica do DVG integrado a um módulo de visão global. Na seção 2 apresentamos o módulo de visão global; na seção 3 apresentamos os conceitos básicos da geração do DVG e a sua implementação computacional no módulo planejador; na seção 4 mostramos como os módulos são integrados ao módulo de controle do robô, formando o sistema **RoTA** de planejamento de trajetórias, e apresentamos alguns resultados da aplicação deste sistema para a navegação do mini-robô Khepera em ambientes de trabalho bidimensionais. Finalmente, na seção 5 apresentamos algumas conclusões e comentários.

## **2. VISÃO GLOBAL**

O módulo de visão global é constituído por uma única câmara CCD colorida no padrão NTSC, a qual está colocada a uma altura de 1.8m acima do ambiente de trabalho. O ambiente de trabalho é formado por uma arena retangular de 100x90cm e as imagens são capturadas a 30 quadros por segundo no modelo de cor RGB, com resolução de 640x480. Em seguida, a imagem é reduzida para 500x450, resultando em uma escala de 5 pixels/cm.

### **2.1. Processamento da imagem**

A aquisição das imagens é realizada pela câmara e são digitalizadas através de uma placa de captura de vídeo que, através do *video for windows*, permite a exibição das imagens na área de trabalho correspondente.

A imagem da área de trabalho é segmentada e considerando um limiar de corte para identificar o fundo e os objetos, a imagem é binarizada. Inicialmente, visando facilitar esta tarefa, a cor do fundo do ambiente de trabalho foi escolhida como branca e para os obstáculos presentes no

ambiente, os quais são formados por blocos com 3,0cm de altura, a cor preta. O robô recebeu rótulos especiais para facilitar a sua identificação, como veremos mais adiante.

Os obstáculos são identificados através de propriedades de conectividade entre os pixels dos objetos segmentados; em seguida aplicamos um algoritmo para detecção dos pixels (Costa, 2001) que compõem a fronteira dos obstáculos, reduzindo cada obstáculo à sua correspondente seqüência de pixels da borda. Em princípio, o conjunto de pixels da seqüência poderia ser diretamente utilizado para gerar o mapa de trajetórias, mas o custo computacional torna-se elevado. Por outro lado, um polígono pode ser caracterizado univocamente através do conjunto ordenado de seus vértices. Deste modo, a identificação dos vértices de cada obstáculo é necessária, fazendo com que o número de pixels diminua exponencialmente, o que reduz drasticamente o custo computacional envolvido na geração do mapa de trajetórias com o DVG.

Há vários algoritmos propostos para detecção de vértices, os quais podem ser classificados em duas grandes categorias: i) algoritmos com base na seqüência de pixels da fronteira ou ii) algoritmos com base na análise morfológica da imagem. Neste trabalho utilizamos um algoritmo de detecção de vértices partindo da seqüência de pixels já obtida para cada obstáculo. O algoritmo parte do princípio que vértices são pontos onde dois segmentos de reta se interceptam. Em geral, vértices são caracterizados como pontos que apresentam uma grande curvatura. Por esta razão muitos métodos foram desenvolvidos para determinação de vértices com base na análise da curvatura. Para polígonos regulares os métodos com base na curvatura apresentam bons resultados, porém para polígonos generalizados, os quais envolvem segmentos e arcos, tais métodos apresentam resultados espúrios. O algoritmo proposto por Tsai et al. (Tsai, 1999), o qual analisa os autovalores da matriz de covariância dos pontos de uma curva, é mais robusto para polígonos generalizados e círculos.

De uma forma sintética, este algoritmo considera a seqüência de  $n$  pixels que forma a borda do objeto  $P$  dada por  $P = \{p_i = (x_i, y_i); i = 1, 2, \dots, n\}$ , onde o pixel  $p_{i+1}$  é vizinho do pixel  $p_i$  módulo  $n$ , e  $(x_i, y_i)$  corresponde a coordenada Cartesiana do pixel  $p_i$  na imagem. Define-se uma região de suporte do pixel  $p_i$  da forma  $S_k(p_i) = \{p_j; j = i - k, i - k + 1, \dots, i + k - 1, i + k\}$ , onde  $k$  é um número inteiro. A região de suporte nada mais é do que um conjunto de pixels vizinhos ao pixel  $p_i$ . Através da análise do valor dos autovalores  $\lambda_L$  e  $\lambda_S$  da matriz de covariância da região de suporte, um ponto  $p_i$  é considerado um vértice se  $\lambda_L$  ou  $\lambda_S$  assumem valores superiores a um valor limiar predeterminado e os vértices estão separados por pelo menos um conjunto  $k$  de pixels. Observa-se que pontos sobre um segmento de reta possuem valores de  $\lambda_S$  muito próximos a zero e que valores elevados de  $\lambda_S$  estão associados a pontos angulosos (ver detalhes em Tsai, 1999). Com a utilização deste algoritmo e tomando o valor do parâmetro de suporte  $k = 6$  e um limiar da ordem de  $10^{-1}$  obtemos bons resultados na detecção de vértices, tanto para ângulos agudos como para rasos.

A determinação dos vértices e arcos permite caracterizarmos minimamente a forma geométrica do obstáculo e assim reconstruí-lo a partir desta mesma estrutura geométrica. Os dados geométricos são então utilizados para a geração do DVG, como veremos na seção 3.

### 3. MAPA DE TRAJETÓRIAS COM DVG

O Diagrama de Voronoi é uma estrutura geométrica fundamentada no conceito de métrica. Em termos simples, o diagrama de Voronoi toma como base o *princípio do vizinho mais próximo*. As aplicações deste conceito estão presentes em várias áreas e, em particular, na robótica devido a algumas propriedades desta estrutura. O diagrama de Voronoi pode ser definido no espaço  $n$ -dimensional. A seguir, introduzimos este conceito para o caso planar.

#### 3.1. Diagrama de Voronoi planar

Formalmente, o diagrama de Voronoi planar (Okabe, 1992) para um conjunto de pontos não colineares  $P = \{p_1, p_2, \dots, p_n\}$  é definido como o particionamento do plano em regiões

$R_i = \{p; d(p, p_i) \leq d(p, p_j), \forall p_i \neq p_j\}$ , chamadas de regiões de Voronoi para o gerador  $p_i$ , onde  $d(p, p_i)$  é a distância Euclideana do ponto  $p$  aos pontos do conjunto de pontos  $P$ .

O diagrama de Voronoi  $V(P)$  para o conjunto de pontos  $P = \{p_1, p_2, \dots, p_n\}$  é obtido pela união de todas as regiões de Voronoi, i.e.,  $V(P) = \bigcup_{i=1}^n R(p_i)$ . As arestas entre duas regiões de Voronoi são

chamadas de arestas de Voronoi e os pontos onde 3 ou mais arestas se interceptam são chamados de vértices de Voronoi. Um gerador de Voronoi  $p_i$  é adjacente a  $p_j$  quando as suas regiões de Voronoi compartilham uma mesma aresta. De acordo com esta definição, qualquer ponto sobre uma aresta de duas regiões adjacentes está equidistante dos geradores de Voronoi correspondentes. Se pensarmos em termos de robótica, podemos imaginar que os geradores de Voronoi são obstáculos pontuais e as arestas formam caminhos com máxima desobstrução com relação aos obstáculos.

O conceito do diagrama planar pontual pode ser estendido para objetos extensos como segmentos, arcos, círculos ou polígonos. Considere o conjunto de objetos do tipo segmentos ou arcos  $L = \{L_1, L_2, L_3, \dots, L_n\} \subset \mathbb{R}^2$  tal que estes não se interceptem  $L_i \cap L_j \neq \emptyset$ , com  $i \neq j$ . Definimos a distância de um ponto  $p$  a  $L_i$  como sendo a menor distância entre  $p$  e um ponto  $p_i$  sobre  $L_i$  da forma:  $d_s(p, L_i) = \min_{x_i} \{\|x - x_i\|; x_i \in L_i\}$ , onde  $x$  e  $x_i$  são os vetores posição de  $p$  and  $p_i$ , respectivamente. A região de Voronoi  $R(L_i)$  é dada por  $R(L_i) = \{p; d_s(p, L_i) \leq d_s(p, L_j), j \neq i, j \in I_n\}$  e a união de todas estas regiões forma o diagrama de Voronoi  $V(L) = \bigcup_{i=1}^n R(L_i)$  para o conjunto  $L$ .

De forma semelhante, podemos estender este conceito para polígonos generalizados e círculos. Considere o conjunto de objetos  $A = \{A_1, A_2, \dots, A_n\}$  em  $\mathbb{R}^2$ , onde  $A_i$  corresponde a área de cada objeto. Assumindo que tais objetos não se interceptam, definimos a distância de um ponto  $p$  a uma área  $A_i$  como a menor distância entre  $p$  e  $p_i$  em  $A_i$ , dada por  $d_s(p, A_i) = \min_{x_i} \{\|x - x_i\|; x_i \in A_i\}$ , onde  $x$  e  $x_i$  são os vetores posição de  $p$  and  $p_i$ , respectivamente. A região de Voronoi  $R(A_i)$  associada ao objeto  $A_i$  é determinada por  $R(A_i) = \{p; d_s(p, A_i) \leq d_s(p, A_j), j \neq i, j \in I_n\}$  e o diagrama de Voronoi generalizado (DVG) é o conjunto  $V(A) = \bigcup_{i=1}^n R(A_i)$ . Em termos de robótica, os segmentos e arcos podem ser vistos como obstáculos do tipo paredes e os polígonos generalizados como a projeção de obstáculos sobre o plano. As arestas de Voronoi para um conjunto de objetos forma um mapa de trajetórias com máxima desobstrução.

### 3.2. Implementação do DVG

Há vários algoritmos propostos para implementação da técnica de geração DVG. Neste trabalho o algoritmo utilizado é uma extensão do algoritmo incremental proposto por Sugihara (Sugihara, 1992) para a caso de pontos no plano. Uma das vantagens deste algoritmo é a sua complexidade, da ordem de  $O(n)$ , onde  $n$  corresponde ao número de pontos dos geradores. A geração do DVG pode ser obtida considerando os objetos como sendo formado por uma seqüência de pontos com um afastamento  $\delta$  entre si. Esta extensão permite a construção aproximada do DVG, cuja resolução é tanto melhor quanto menor for o valor de  $\delta$ . Porém, valores muito pequenos de  $\delta$  aumentam o custo computacional da geração do DVG. A Figura 1 ilustra o DVG para um conjunto de objetos.

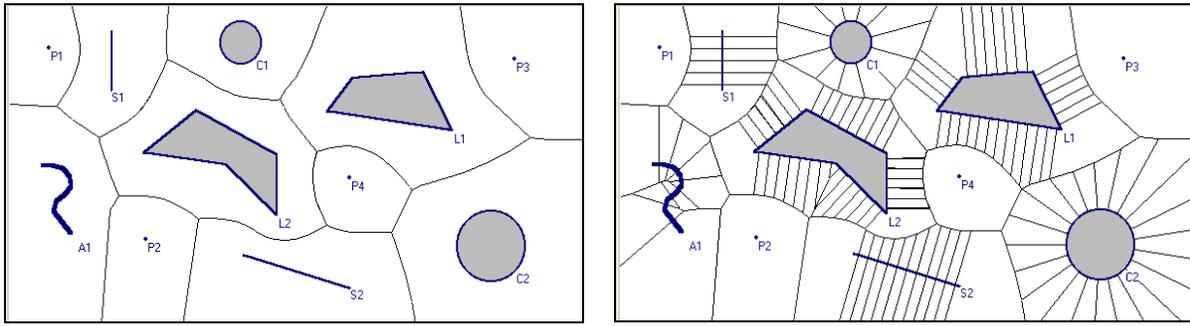


Figura 1. DVG para objetos extensos. A figura da direita mostra as arestas geradas com o afastamento  $\delta$ . Tais arestas são consideradas inválidas pois interceptam os objetos.

#### 4. O SISTEMA RoTA

O problema do planejamento de trajetórias de robôs móveis consiste em: dadas duas configurações  $q_i$  inicial e  $q_a$  alvo, o sistema de navegação deve ser capaz de determinar um caminho livre de colisões entre as duas configurações, ou reportar falha caso este caminho não exista.

##### 4.1. Estrutura geométrica

Com base na técnica do DVG um sistema computacional para o planejamento de trajetórias foi desenvolvido em C++ para plataformas Windows. O sistema, denominado de RoTA, gera o mapa de trajetórias a partir da especificação da geometria dos obstáculos presentes no ambiente de trabalho. A estrutura dos dados geométricos do DVG da Figura 2 é apresentada abaixo.

```
VdView_Data_File V3.4
E 1 1 50
R 0
P (41,41) 0
P (141,231) 0
P (521,52) 0
P (351,170) 0
S (106,25) (106,84) 0
S (242,247) (352,280) 0
C (239,36) 20 0
C (497,238) 34 0
A 0 9 (35,158) (46,158) (56,161) (63,169) (64,179) (57,187) (49,193) (46,203) (64,225)
L 0 4 (354,71) (328,104) (457,123) (427,65)
L 0 5 (139,145) (193,103) (276,147) (276,207) (224,157)
```

Na estrutura acima E corresponde aos parâmetros de escala nas direções dos eixos e ao valor da malha, P indica objeto do tipo ponto seguido de suas coordenadas, S indica objeto do tipo segmento seguido das coordenadas das suas extremidades, C indica objeto do tipo círculo com as coordenadas do centro e o raio, A indica objeto do tipo arco com o número de segmentos e a seqüência de coordenadas que formam a poligonal, e L indica objeto do tipo polígono com o número de vértices e a seqüência de coordenadas de cada um dos vértices.

No sistema RoTA assumimos que o robô é não-holonômico do tipo disco. O raio do robô é especificado pelo parâmetro R. Quando o valor de R é diferente de zero, isto é, o robô é um objeto extenso, a técnica de expansão isotrópica dos obstáculos é utilizada de forma a manter o robô como um objeto pontual e envelopar os obstáculos com o valor do raio do robô. Desta maneira, preservamos a estrutura do mapa de trajetórias.

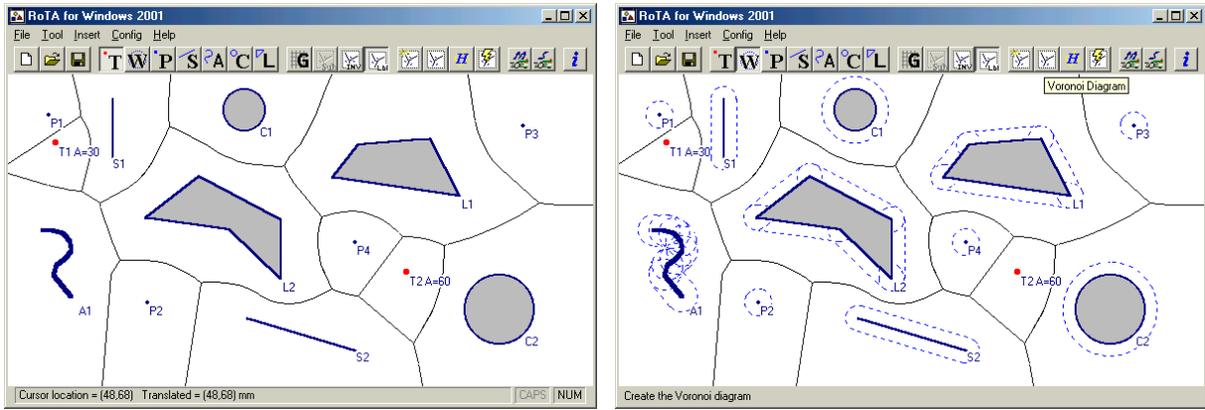


Figura 2. Mapa de trajetórias com as configurações inicial T1 e alvo T2 (esquerda), e com o robô como um objeto extenso (direita).

## 4.2. Trajetória de menor caminho

O mapa de trajetórias do DVG possui a estrutura de um grafo não dirigido, onde os vértices de Voronoi correspondem aos nós do grafo. Desta forma, as técnicas e algoritmos aplicados a grafos podem ser utilizadas para o DVG. Considerando o ambiente de trabalho com fronteira, o sistema RoTA é capaz de determinar a trajetória de menor caminho e máxima desobstrução (TMCMD) para duas configurações quaisquer no espaço de trabalho. Embora a TMCMD represente um caminho seguro para navegação, a mesma não é a trajetória ótima entre as configurações, homotopicamente falando. Partindo-se da TMCMD uma otimização é obtida pelo sistema computando o *atalho de visibilidade*, que é uma trajetória ainda mais curta e com menos conexões do que a TMCMD. A diminuição do número de conexões que liga a configuração  $q_i$  à  $q_a$  é importante uma vez que esta reduz o número de pontos angulosos na trajetória, diminuindo o número de rotações necessárias a ser realizadas pelo robô e em consequência a precisão da navegação.

Na Figura 3 ilustramos o mapa de trajetórias, a TMCMD e o atalho de visibilidade entre duas configurações. Uma configuração  $q$  corresponde ao vetor  $(x, y, \theta)$ , onde  $x, y$  são as coordenadas espaciais e  $\theta$  a orientação do robô.

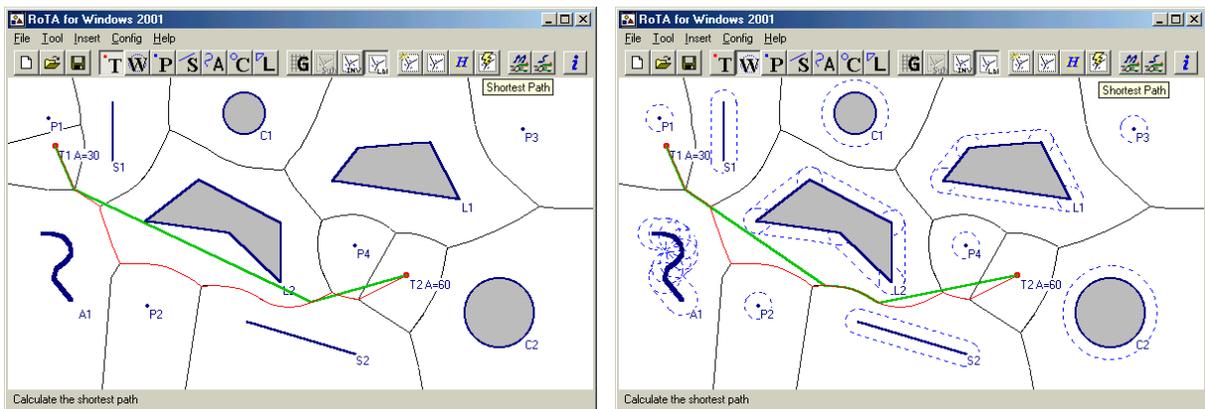


Figura 3. Mapa de trajetórias exibindo a TMCMD e o atalho de visibilidade entre as configurações T1 e T2. O valor da orientação do robô é visto em  $A = \theta$ , com ângulo em graus. Na figura da direita o robô é considerado um objeto extenso.

### 4.3. Integração com a visão global

O módulo de visão global processa os dados das imagens capturadas do espaço de trabalho e gera a estrutura de dados geométricos. Esta estrutura é utilizada pelo módulo de geração do mapa de trajetórias com o DVG. O sistema RoTA permite configura-lo de modo a exibir na área de trabalho a imagem dos dados geométricos ou a imagem real do ambiente de trabalho. Desta forma, o usuário pode visualizar em tempo-real o ambiente de trabalho do robô e pode acompanhar a sua navegação ao longo da trajetória que é definida pelo atalho de visibilidade.

Na Figura 4 vemos a captura de tela do RoTA com a exibição da imagem geométrica (esquerda) e da imagem real do ambiente de trabalho (direita). Na imagem geométrica, o robô é identificado pelas coordenadas da sua configuração inicial, vista como T1 na figura da esquerda. Na imagem real, o robô é identificado pelas características do círculo e triângulo, que geram a configuração T1.

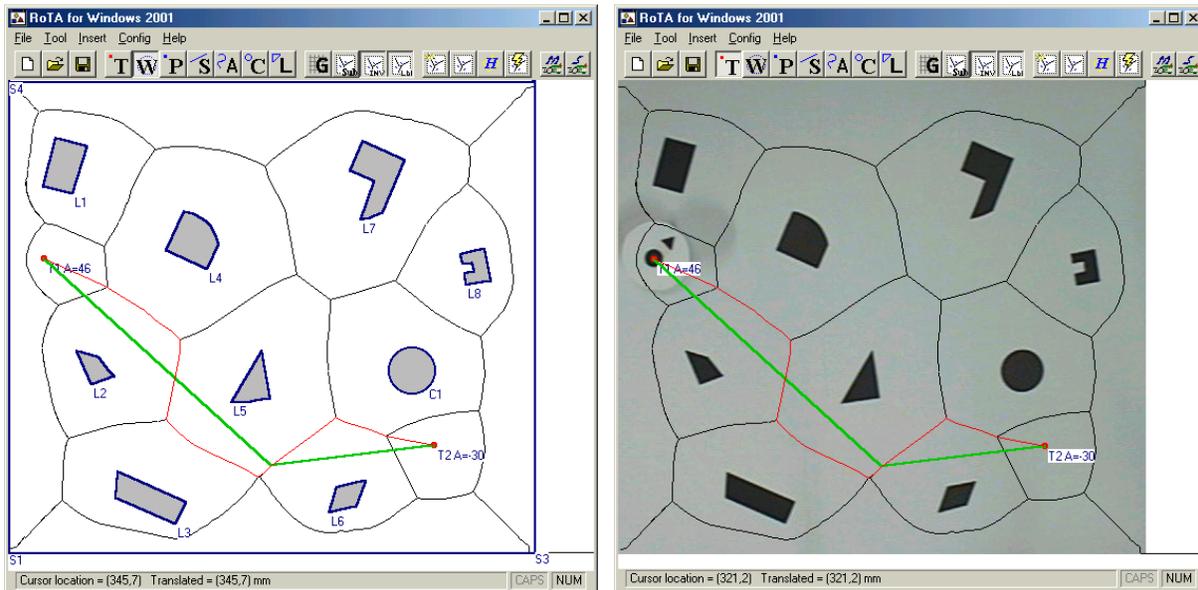


Figura 4. Imagem da estrutura geométrica (esquerda) e imagem real do ambiente de trabalho do robô (direita). O robô é identificado através do conjunto formado pelo círculo-triângulo.

### 4.4. O mini-robô Khepera

O mini-robô Khepera (Mondada, 1999; K-Team, 1998) é um robô miniatura circular com 5,5cm de diâmetro e 3,0cm de altura, com dois motores de passo independentes, controlados por um processador Motorola. O Khepera possui dois sensores incrementais nas rodas e 8 sensores de infravermelho (proximidade e luminosidade) distribuídos ao longo do seu perímetro. Outros sensores e módulos podem ser instalados no robô, como câmaras e garra, por exemplo. Uma interface via cabo serial RS-232 permite a conexão do Khepera com um computador hospedeiro, via a porta serial. Esta conexão umbilical também supre o robô com energia elétrica, embora o robô disponha de baterias recarregáveis. Uma sistema de comunicação por rádio, usando o protocolo SerCom, permite a comunicação remota entre o computador hospedeiro e o robô.

Uma série de comandos básicos em ASCII estão disponíveis para controle de funcionalidades do Khepera, via o protocolo de comunicação. Através destes comandos, é possível desenvolvermos novas funções de modo a permitir um controle mais sofisticado do robô.

O sistema RoTA dispõe de uma função que, uma vez especificada a configuração alvo, permite a navegação autônoma do robô ao longo da trajetória dada pelo atalho de visibilidade. Quando o robô alcança a configuração alvo, o módulo de visão atualiza a sua configuração, bem como a dos

obstáculos e todo o processamento continua até que uma nova configuração alvo seja especificada pelo usuário. Atualmente, durante a navegação o processamento de imagens é interrompido, embora a visualização em tempo-real do cenário esteja sendo exibida na área de trabalho do sistema ROTA, permitindo que o usuário visualize a navegação do robô.

Para a navegação do robô no ambiente de trabalho e a visualização simultânea na área de trabalho do sistema ROTA, uma função que define o fator de escala foi criada de modo que a relação entre o número de pixels da área de trabalho seja compatível com a dimensão do ambiente de trabalho do robô. De acordo com os dados citados anteriormente, o fator de escala é de 5 pixels/cm em ambas as direções.

Na Tabela 1 apresentamos os resultados dos desvios observados entre os valores das configurações alvo especificadas pelo usuário e os valores de suas atualizações determinados pelo módulo de visão global, após a navegação. A navegação do Khepera foi realizada usando-se a comunicação por rádio. Os resultados apresentados na Tab.(1) foram decorrentes da realização de 26 experimentos de navegação controlado pelo sistema ROTA. A Figura 5 mostra os gráficos dos valores das coordenadas da configuração alvo e os medidos pelo módulo de visão global.

Tabela 1. Resultados de 26 experimentos de navegação do robô.

Coordenadas da configuração	$x$	$y$	$\theta$
Intervalo de confiança da média (95%)	0,1237±0,0638	0,1183±0,0336	0,0261±0,0095
Desvio padrão	0,1579	0,0833	0,0234

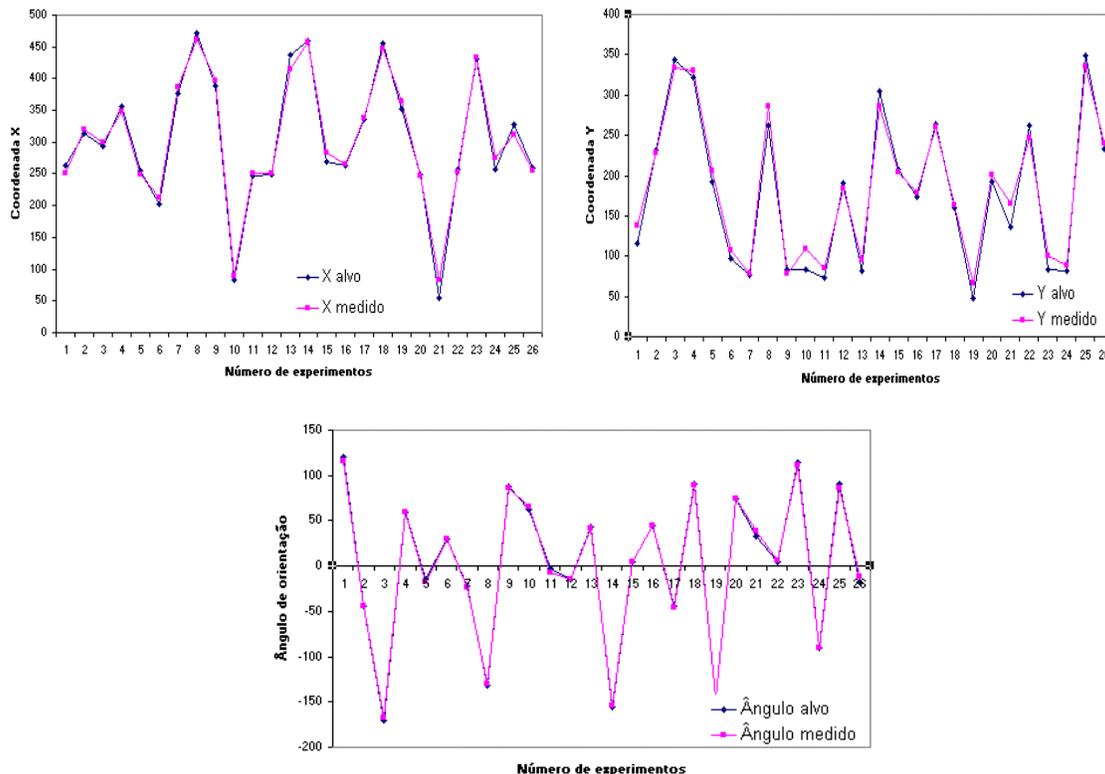


Figura 5. Os gráficos acima mostram os desvios sofridos pelas coordenadas da configuração.

## 5. CONCLUSÃO

Neste trabalho apresentamos o estágio atual de desenvolvimento de um sistema de planejamento de trajetórias para robôs móveis autônomos, o qual utiliza a técnica de geração de mapas de trajetórias a partir da construção geométrica do diagrama de Voronoi generalizado (DVG). A estrutura geométrica do espaço de trabalho é obtida através de um módulo de visão global, composto por uma câmara de vídeo (NTSC, RGB, 640x480) colocada no centro e a uma altura de 1.8m acima da arena que forma o ambiente de trabalho do robô.

O módulo de visão global captura e processa as imagens de forma a extrair a estrutura geométrica dos obstáculos, os quais são objetos poligonais generalizados e círculos. O robô é identificado pelo módulo de visão global a partir das características especiais, as quais estão colocadas sobre o mesmo. Como o robô é do tipo disco, o módulo de visão determina o seu centróide e o vetor que indica a sua frente. Desta forma, o problema de localização do robô no espaço de trabalho é resolvido através da determinação da sua configuração  $q = (x, y, \theta)$ .

Os dados da estrutura geométrica são utilizados pelo módulo de geração do mapa de trajetórias com o DVG. O sistema de planejamento de trajetórias RoTA integra os módulos de visão global, geração do mapa de trajetórias, determinação da trajetória de menor caminho com máxima desobstrução (TMCMD) e do atalho de visibilidade, e o das funções de controle do robô. O RoTA está sendo escrito em C++ para plataformas Windows. O módulo responsável pelo controle do robô é flexível podendo ser adaptado para outros tipos de robôs móveis. Como exemplo de aplicação, estamos simultaneamente trabalhando no módulo de controle do robô móvel que vem sendo completamente desenvolvido pelo GEAR da PUCMG. Infelizmente, algumas dificuldades impossibilitaram a realização de testes de navegação deste robô com o sistema RoTA, cujos resultados deverão ser apresentados futuramente.

Visando ajustes, validação e análise de desempenho do sistema RoTA, uma bateria de 26 experimentos de navegação foi realizada, avaliando-se os desvios ocorridos entre as coordenadas da configuração especificada pelo usuário como configuração alvo e a configuração que é lida pelo módulo de visão global, quando o robô atinge o alvo após a sua navegação no espaço de trabalho. Os desvios observados são uma consequência da combinação de vários tipos de erros como aqueles inerentes aos componentes mecânicos, decorrentes da calibragem da escala, da própria iluminação do ambiente, etc. Neste sentido, os resultados apresentados evidenciam uma ótima concordância, garantindo a confiabilidade do sistema de planejamento e navegação.

Com a utilização dos sensores de proximidade, estamos atualmente na fase de desenvolvimento do algoritmo para replanejamento da trajetória quando um obstáculo intruso é detectado pelo robô. Neste caso, o sistema identificará a presença do obstáculo e reconhecerá se o mesmo é um obstáculo conhecido ou intruso. Em ambos os casos, o sistema replanejará a trajetória levando em consideração os dados da nova configuração do robô após o reconhecimento do novo obstáculo. Paralelamente a isso, estamos estendendo o algoritmo de planejamento para múltiplas estações de visita, as quais deverão ser visitadas sequencialmente pelo robô.

## 6. AGRADECIMENTOS

Este projeto recebeu fomento do CTPetro/CNPq sob número 522635/95-4.

## 7. REFERÊNCIAS

- World Robotics 2001, The International Federation of Robotics, United Nations Report, Geneva, Switzerland, 348p.
- Blanco, D., Boada, B.L., Moreno, L and Salichs, M.A., 2000, "Local Mapping from On-Line Laser Voronoi Extraction", Proceedings of the International Conference on Intelligent Robotics and Systems, IEEE/RSJ, pp 103-108.
- Choset H and Burdick J., 1995, "Sensor-Based Planning: The Generalized Voronoi Graph" and "Incremental Construction of the Generalized Voronoi Graph". In Proceedings of the

- International Congress on Robotics and Automation, IEEE, Nagoya, Japan, Vol. 2, pp.1649-1655; pp.1643-1648.
- Costa, L.F. and Cesar Jr., R.M., 2001, "Shape Analysis and Classification: Theory and Practice", CRC Press, Boca Raton, USA, 659p.
- Latombe J-C., 1991, "Robot Motion Planning". Kluwer Academic Publishers, New York, USA.
- Mahkovic, R. and Slivnik, T., 2002, "Constructing the Generalized Local Voronoi Diagram from Laser Range Scanner Data", IEEE Transection on Systems, Man, and Cybernetics, Part A, Vol. 30, pp. 710-719.
- Okabe, A., Boots, B. and Sugihara, K., 1992, "Spatial Tesselation Concepts and Applications of Voronoi Diagrams", John Wiley & Sons, New York.
- Roque, W. L., 1996, "Introdução à Técnicas de Planejamento de Trajetórias de Robôs Móveis", Anais da IV Escola Regional de Informática, Sociedade Brasileira de Computação, Ed. M. Fornari et al., pp. 125-150.
- Sugihara, K. and Iri, M., 1992, "Construction of the Voronoi Diagram for One Million Generators in Single-Precision Arithmetic". Proceedings of the IEEE, Vol. 80, pp. 1471-1484.
- Tsai, D.-M., Hou, H.-T. and Su, H.-J., 1999, "Boundary-Based Corner Detection Using Eigenvalues of Covariance Matrices", Pattern Recognition Letters, Vol. 20, pp. 31-40.
- Mondada, F., Franzi, E. and Guignard, A., 1999, "The Development of Khepera", Proceedings of the First International Khepera Workshop, Eds. A. Löffler, F. Mondada and U. Rückert, Paderborn, Germany, pp. 7-13.
- Khepera User Manual. Version 5.0, K-Team, Lausanne, Switzerland, 1998.

## **8. DIREITOS AUTORAIS**

### **A PATH PLANNING SYSTEM FOR MOBILE ROBOTS BASED ON GLOBAL VISION**

***Abstract.** One of the major problems for autonomous mobile robots is the collision free path planning between two configurations (position and orientation) in the workspace. The Generalized Voronoi Diagram (GVD) is an important technique that allows the generation of a roadmap with maximal clearance from the obstacles and, in addition, it is fully connected for bounded workspaces. However, the construction of the roadmap with the GVD requires that the geometrical structure of the workspace be known a priori. An alternative to circumvent this apparent drawback is making use of a global vision system, which will identify the geometrical structure of the workspace. In this paper we discuss the current development of a path planning system based on the DVG roadmap integrated to a global vision module (GVM). The objects present in the scenario, including the robot, are segmented and their geometrical characteristics are identified. The geometrical data are read by the trajectory planner module that generates the maximal clearance feasible paths and the visibility pathways between the initial and goal configurations. The GVM solves the robot localization o problem. The goal configuration is specified directly by the user on the work area of the planning system. Some experiments for system validation have been conducted with the laboratory miniature robot Khepera and the results are presented.*

***Keywords.** Mobile robots, path planning, global vision, Voronoi diagrams.*