

O USO DA TÉCNICA ZERO MOMENT POINT EM ROBÔS HUMANOIDES

Marciano F. de Macedo, Isaac J. da Silva, Reinaldo A. C. Bianchi, *Membro, IEEE*

Abstract— Advances in the development of bipedal humanoid robots have brought about the need for research to improve their walking performance. This article describes the walking process of bipedal humanoid robots, using the Zero Moment Point (ZMP) technique in the Matlab simulator. The result presented shows that the ZMP is a useful tool for humanoid robots.

I. INTRODUÇÃO

Este artigo descreve o processo de caminhada de robôs humanóides bípedes, baseado na distribuição de forças entre os pés do robô. Basicamente, existem dois tipos de caminhada para robôs humanóides bípedes:

- “Static Walking” - a projeção do centro de massa do robô sempre dentro da área de suporte dos pés. Fácil implementação, porém, com desempenho lento devido aos passos curtos.
- “Dynamic Walking” - a projeção do centro de massa caminha fora da área de suporte dos pés, utilizando conceitos como ZMP (Zero-Moment Point) ou FRI (Foot-Rotation Indicator) para atingir a estabilidade do robô durante a caminhada.

II. ESTADO DA ARTE

O artigo “ZMP trajectory from human body locomotion dynamics evaluated by Kinect-based motion capture system”, apresenta os métodos de avaliação da trajetória utilizando ZMP, por meio do processamento de dados biomecânicos utilizando o sensor Kinect.

O artigo “Reference ZMP manipulation for energetic and computationally efficient walking using ZMP preview control”, utiliza o ZMP para controle da caminhada dos robôs lembrando que existe uma ineficiência energética no resultado. Com referência a abordagens modernas de caminhada como Model Predictive Control (MPC), os autores descrevem como obter padrões de caminhada mais naturais através da manipulação do ZMP.

O artigo “Control of a humanoid robot based on the ZMP method”, descreve o uso do ZMP para o controle de caminhada de um robô humanóide bípede, para uso doméstico.

O artigo “Natural Motion Trajectory Generation of Biped Locomotion Robot using Genetic Algorithm through Energy Optimization” utiliza o ZMP com Algoritmo Genético para gerar trajetórias de robôs humanóides bípedes com movimento mais natural e minimização de energia.

III. DEFINIÇÃO

Ao apoiar o pé sobre uma superfície e exercendo a força relativa ao peso do robô, o solo reagirá contra o pé com forças opostas. A soma de todas estas forças, sob o pé do robô, produz uma força resultante R (figura 1).

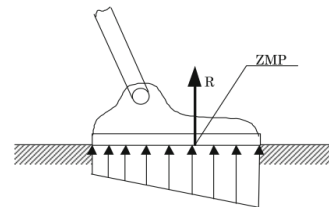


Figura 1: Reação de forças contra o pé e resultante R .
Fonte: Kajita, 2014.

O sistema de controle do robô deverá mover as juntas das pernas (cinemática inversa), posicionando a resultante R ao longo de um ponto seguro, enquanto o outro pé encontra-se em voo. Quando este pousar, poderá fazer a troca para o outro pé, mantendo o conjunto em equilíbrio. A resultante R é definida como ponto de momento zero, ou ZMP.

IV. ZMP E POLÍGONO DE SUPORTE

A. Polígono de suporte

É a área de suporte do robô durante a caminhada, compreende o espaço entre os pés. O centro de massa do robô alterna sua posição entre os pés, porém sempre dentro do polígono de suporte. O polígono de suporte compreende todo espaço dentro dos limites em vermelho (figura 2).



Figura 2: Polígono de suporte
Fonte: Kajita, 2014

B. ZMP

O ZMP serve como limite (restrição) para os cálculos de cinemática inversa que farão o robô caminhar (figura 3).

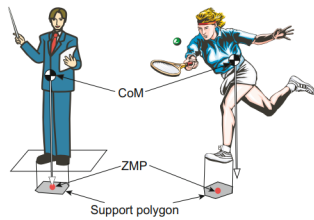


Figura 3– COM, ZMP e polígono de suporte (restrição)
Fonte: Kajita, 2014

“As discussões como “Dependendo do movimento do robô, o ZMP pode sair do polígono de suporte?” ocorre frequentemente. Claro, a conclusão é que “a ZMP nunca existe fora do polígono de suporte”. (Kajita, 2007)

C. Restrição

O ZMP pode não ser tão efetivo em um terreno acidentado ou em uma escada. Quando o robô caminha agarrado a um corrimão, o contato pode ser mais estável, mas o ZMP não garante a estabilidade do contato. Várias tentativas foram feitas para eliminar estas limitações, mas um novo critério ainda não foi estabelecido.

“O ZMP não pode ser utilizado nos seguintes casos:

[A] determinar se a sola desliza na superfície do solo.

[B] a superfície do solo não é plana.

[C] os braços ou mãos de um robô humanoide entram em contato com o meio ambiente.” (Kajita, 2007)

V. CÁLCULO DO ZMP

O Cálculo do ZMP consiste em várias etapas. A primeira, e muito importante, é o cálculo do Centro de Massa (COM) do robô. Na sequência tem-se o Momento Linear e Momento Angular do robô, baseados na distribuição de forças sob os pés do robô. Os parâmetros do robô devem estar contidos numa tabela, os quais podem ser acessados durante os cálculos.

A. Centro de Massa (COM)

O Centro de Massa do robô é um ponto no espaço cartesiano, representando o corpo do robô. Resultado da soma das massas de todos os pontos do robô, incluindo sua posição cartesiana (figura 4).

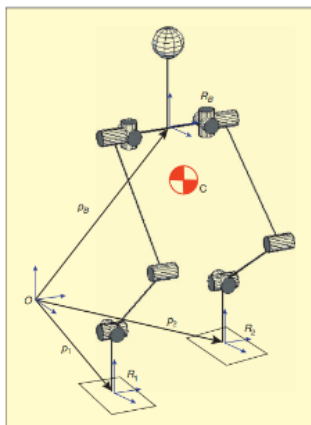


Figura 4; Centro de Massa (COM)
Fonte: Kajita, 2007

A Equação (1) faz o cálculo do COM e utiliza os seguintes parâmetros:

$$c = \sum_{j=1}^N m_j c_j / M. \quad (1)$$

c = Centro de Massa (COM)

m = massa do ponto ‘j’

c = coordenada cartesiana

‘j’ varia de 1 ao número de pontos dentro do corpo do robô

M = massa total

As partes do robô necessárias para o cálculo do COM encontram-se na figura 5.

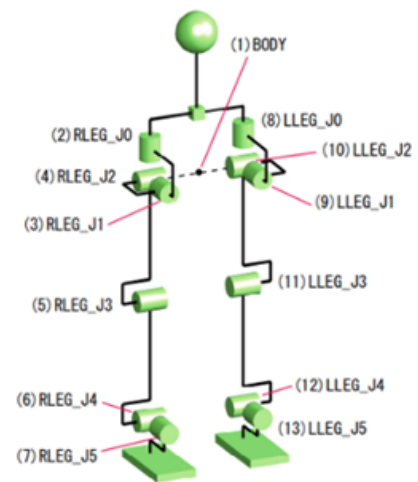


Figura 5 - Componentes do robô
Fonte: Kajita, 2014

A massa de cada parte do robô (m) é apresentada na tabela 1:

name	m
'BODY'	10
'RLEG_J0'	5
'RLEG_J1'	1
'RLEG_J2'	5
'RLEG_J3'	1
'RLEG_J4'	6
'RLEG_J5'	2
'LLEG_J0'	5
'LLEG_J1'	1
'LLEG_J2'	5
'LLEG_J3'	1
'LLEG_J4'	6
'LLEG_J5'	2

Tabela 1 – Tabela com as massas de cada junta do robô
Fonte: Kajita, 2014

A massa total (M) pode ser calculada com o código recursivo (Código 1), utilizando os parâmetros individuais de cada junta.

```

function m = TotalMass(j)
global uLINK
if j == 0
    m = 0;
else
    m = uLINK(j).m + TotalMass(uLINK(j).sister) ...
        + TotalMass(uLINK(j).child);
end

```

Código 1 - Cálculo de massa (M)
Fonte: Kajita, 2014

Após o processamento tem-se a massa total (M) igual a 50.

O cálculo da segunda parte da fórmula (MC), corresponde à somatória dos produtos das massas individuais de cada parte do robô por suas coordenadas cartesianas. A tabela 2 apresenta os parâmetros.

name	m	c	p	R
'BODY'	10	[0;0;0.300000000000000]	[0.2218569910 [1,0,0;0,1,0;0,0,1]	
'RLEG_J0'	5	[0;0;0]	[0.2218569910 [1,-9.7578195523]	
'RLEG_J1'	1	[0;0;0]	[0.2218569910 [1,-9.4664753187]	
'RLEG_J2'	5	[0;0;0]	[0.2218569910 [0.687183707833]	
'RLEG_J3'	1	[0;0;0]	[0.4398020980 [0.687183707833]	
'RLEG_J4'	6	[0;0;0]	[0.2218569910 [1,-9.4664753187]	
'RLEG_J5'	2	[0.0500000000000000;0;-0.04000]	[0.2218569910 [1,-2.7845252778]	
'LLEG_J0'	5	[0;0;0]	[0.2218569910 [1,6.1796347458]	
'LLEG_J1'	1	[0;0;0]	[0.2218569910 [1,6.0417914795]	
'LLEG_J2'	5	[0;0;0]	[0.2218569910 [0.999614733456]	
'LLEG_J3'	1	[0;0;0]	[0.2135302498 [0.534607593138]	
'LLEG_J4'	6	[0;0;0]	[-0.039999876 [1,6.0417914795]	
'LLEG_J5'	2	[0.0500000000000000;0;-0.04000]	[-0.039999876 [1,2.3377599013]	

Tabela 2 - Tabela com demais parâmetros do robô
Fonte: Kajita, 2014

m: massa da parte do robô
c: centro de massa da parte do robô
p: posição da parte do robô
R: matriz de rotação da parte do robô

O código 2 calcula o produto da massa individual por sua coordenada cartesiana.

```

function mc = calcMC(j)
global uLINK
if j == 0
    mc = 0;
else
    mc = uLINK(j).m * (uLINK(j).p + uLINK(j).R * uLINK(j).c);
    mc = mc + calcMC(uLINK(j).sister) + calcMC(uLINK(j).child);
end

```

Código 2 - Código recursivo para cálculo do MC
Fonte: Kajita, 2014

Após o processamento o resultado (mc) será:

[0,2682; -1,7036; 20,5989].

Por último é calculado o Centro de Massa (COM) do robô (figura 6), com resultado apresentado, com MC/M , cujo resultado será:

$COM = [0,0054; 0,0341; 0,4120]$

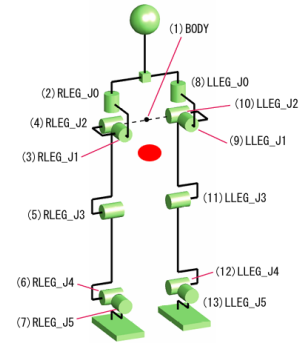


Figura 6 - Centro de Massa (COM) em relação ao robô
Fonte: Kajita, 2014

B. Distribuição de forças no espaço

Quando o pé do robô toca a superfície aparecerá força de reação contrária, devido a seu peso: reação vertical (a) e reação horizontal (b) conforme figura 7:

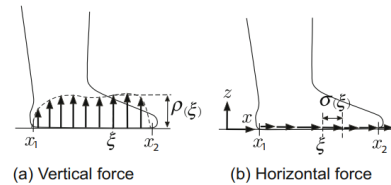


Figura 7 – Forças de reação da superfície no espaço 2D
Fonte: Kajita, 2014

As forças de reação do solo produzem uma resultante R, a qual aparece ao longo do pé do robô em função da distribuição da força peso, podendo ser homogênea (a), distribuída com ponto acentuado (b) ou concentrada (c), figura 8:

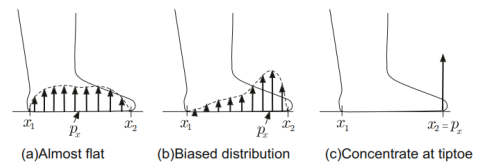


Figura 8 - distribuição de forças de reação
Fonte: Kajita, 2014

Durante a caminhada do robô, a resultante R caminha entre o pé de apoio, enquanto o outro pé encontrava-se em voo (figura 9). No espaço 2D, a força (f) e o momento ($\tau(px)$) no ponto px do solo podem ser expressos por (figura 9):

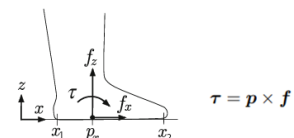


Figura 9 - Força de reação com solo no espaço 2D
Fonte: Kajita, 2014

O ponto p_x é o lugar onde o momento se torna igual a zero. Sendo, $\tau(p_x) = 0$, o ponto p_x pode ser obtido conforme a Equação (2):

$$p_x = \frac{\int_{x_1}^{x_2} \xi \rho(\xi) d\xi}{\int_{x_1}^{x_2} \rho(\xi) d\xi} \quad (2)$$

“ $\rho(\xi)$ é equivalente à pressão, pois é a componente vertical das forças por unidade de comprimento. Assim, p_x definido em (2) é o centro de pressão e é o ZMP.” (Kajita, 2007)

No espaço 3D, surge mais uma coordenada, figura 10.

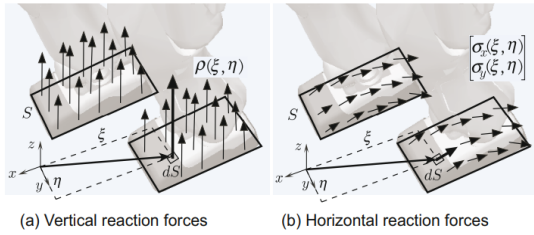


Figura 10 - Força de reação do solo no espaço 3D
Fonte: Kajita, 2014

A soma das forças verticais pode ser expressa conforme a Equação (3):

$$f_z = \int_S \rho(\xi, \eta) dS \quad (3)$$

A integral de S denota a área de integração no contato entre a sola e o solo. O ponto onde o momento da componente vertical da força de reação do solo se torna zero pode ser expresso em (4) e (5), onde a integral dos momentos (x e y) é dividida pela integral da área de integração S das forças verticais (separando os componentes x e y):

$$p_x = \frac{\int_S \xi \rho(\xi, \eta) dS}{\int_S \rho(\xi, \eta) dS} \quad (4)$$

$$p_y = \frac{\int_S \eta \rho(\xi, \eta) dS}{\int_S \rho(\xi, \eta) dS} \quad (5)$$

O sistema deve fazer o cálculo da cinemática inversa entre cada Momento, para definir a nova posição de cada junta relacionada ao movimento do robô.

C. Momento Linear

Quando o robô caminha para frente, uma força surge puxando-o no sentido frontal (Momento Linear), conforme figura 11:

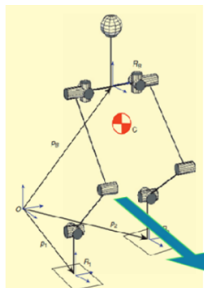


Figura 11 - Momento Linear
Fonte: Kajita, 2007

O cálculo do Momento Linear é efetuado com a fórmula da Equação (6), e utiliza os seguintes elementos:

$$\mathcal{P} = \sum_{j=1}^N m_j \dot{c}_j \quad (6)$$

m_j	massa do componente do robô
$\dot{c}_j = v_j + \omega_j \times (R_j \bar{c}_j)$	composto pelos elementos abaixo:
$v_j = v_i + \omega_i \times R_i b_j$	velocidade linear da junta
$\omega_j = \omega_i + R_i a_j \dot{q}_j$	velocidade angular da junta
R_j	matriz de rotação da junta
c_j	centro de massa da junta

O código 3 permite o cálculo do Momento Linear, utilizando os parâmetros do robô:

```
function P = calcP(j)
global uLINK

if j == 0
    P = [0 0 0]';
else
    c1 = uLINK(j).R * uLINK(j).c;
    P = uLINK(j).m * (uLINK(j).v + cross(uLINK(j).w, c1));
    P = P + calcP(uLINK(j).sister) + calcP(uLINK(j).child);
end
```

Código 3 - Cálculo do momento linear
Fonte: Kajita, 2014

A Equação de Movimento Translacional, Equação (7), corresponde à soma das forças internas e externas aplicadas ao robô. A força gravitacional é aplicada igualmente a todos os componentes do robô, e sua soma pode ser considerada como a força Mg aplicada no centro de massa do robô.

$$\dot{P} = Mg + f \quad (7)$$

A força da gravidade (g) varia em função do lugar onde o robô se encontra (Tabela 3), atuando no eixo z .

$[0 \ 0 \ -9.8]^T$	$[m/s^2]$ on the Earth,
$[0 \ 0 \ -1.7]^T$	$[m/s^2]$ on the Moon,
$[0 \ 0 \ -3.6]^T$	$[m/s^2]$ on Mars

Tabela 3 - Força da gravidade em diferentes planetas

A força (f) corresponde a outras forças, além da gravidade (forças externas).

D. Momento Angular

Quando o robô caminha para frente, uma segunda força surge tentando empurrá-lo para frente no sentido angular (figura 12).

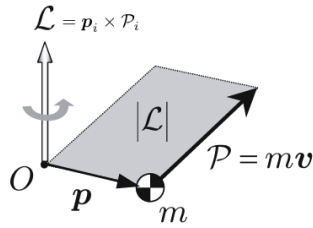


Figura 12 – Relação entre Momento Linear e Angular
Fonte: Kajita, 2014

O momento angular não é apenas uma propriedade do movimento rotacional. Pode-se calcular o momento angular de uma massa pontual movendo-se em linha reta com velocidade constante. Nesse caso, o momento angular se mantém constante (conservação do momento angular), figura 13:

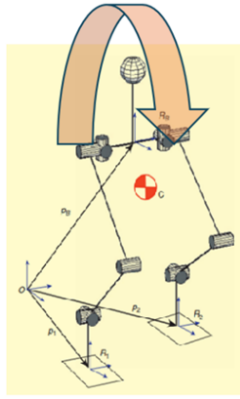


Figura 13 - Momento Angular
Fonte: Kajita, 2007

A Equação (8) faz o cálculo do Momento Angular, utilizando os componentes abaixo:

$$\mathcal{L} = \sum_{j=1}^N \mathcal{L}_j \quad (8)$$

$$\mathcal{L}_j = c_j \times P_j + R_j \bar{I}_j R_j^T \omega_j$$

c_j	massa pontual (p)
P_j	Momento Linear
$R_j \bar{I}_j R_j^T$	Tensor de inércia
ω_j	Velocidade angular da junta

O código 4 permite o cálculo do Momento Angular, utilizando os parâmetros do robô:

```
function L = calcL(j)
global uLINK
if j == 0
    L = 0;
else
    c1 = uLINK(j).R * uLINK(j).c;
    c = uLINK(j).p + c1;
    P = uLINK(j).m * (uLINK(j).v + cross(uLINK(j).w, c1));
    L = cross(c, P) + uLINK(j).R * uLINK(j).I * uLINK(j).R' * uLINK(j).w;
    L = L + calcL(uLINK(j).sister) + calcL(uLINK(j).child);
end
```

Código 4 - Código recursivo para cálculo do Momento Angular
Fonte: Kajita, 2014

A Equação de movimento rotacional, equação (9), é a soma dos momentos angulares (internos e externos) aplicadas ao robô. A força gravitacional é aplicada igualmente a todos os componentes do robô, e sua soma pode ser considerada como a força Mg aplicada no centro de massa do robô. Se $(c.Mg)$ estiver desbalanceado com T , o robô cai para frente.

$$\dot{\mathcal{L}} = c \times Mg + \tau \quad (9)$$

O produto $(c.Mg)$ é o momento gerado pela força gravitacional (g), ou seja, forças internas. T é o momento sem o efeito da gravidade (forças externas).

E. Cálculo do ZMP

“Como o ZMP é equivalente ao centro de pressão, o seu significado físico é muito claro. Além disso, a relação entre o ZMP e o momento linear/angular do robô pode ser expressa por uma equação simples.” (Kajita, 2007)

Agrupando as seguintes equações: Equação do Momento da força de reação com solo (10), Equação de Movimento Translacional (11) e Equação de Movimento Rotacional (12), obtém-se a equação final do ZMP (13):

$$\tau = p \times f + \tau_p \quad (10)$$

$$\dot{P} = Mg + f \quad (11)$$

$$\dot{\mathcal{L}} = c \times Mg + \tau \quad (12)$$

$$+\tau_p = +\dot{\mathcal{L}} - c \times Mg + \dot{P} - Mg \times p \quad (13)$$

As Equações (14) e (15) permitem encontrar os pontos ZMP para p_x e p_y :

$$p_x = \frac{Mgx + p_z \dot{P}_x - \dot{\mathcal{L}}_y}{Mg + \dot{P}_z} \quad (14)$$

$$p_y = \frac{Mgy + p_z \dot{P}_y + \dot{\mathcal{L}}_x}{Mg + \dot{P}_z} \quad (15)$$

O código 5 calcula o ZMP para ambas as posições:

```
function [px,py] = calcZMP(c,dP,dL,pz)
global M G
px = (M*G*c(1) + pz * dP(1) - dL(2))/(M*G + dP(3));
py = (M*G*c(2) + pz * dP(2) + dL(1))/(M*G + dP(3));
```

Código 5 – Código para o cálculo do ZMP

Fonte: Kajita, 2014

Os códigos utilizados neste artigo podem ser encontrados em:
<https://github.com/s-kajita/IntroductionToHumanoidRobotics/tree/master>.

F. Resultados

Quanto maior a altura do robô, maior será a ação da força da gravidade sobre as partes mais altas do mesmo, forçando-o a cair. Por isso utiliza-se um segundo sistema de controle que procurará manter as coordenadas 'x' e 'y' do COM dentro de uma margem de segurança. Este sistema é chamado Pendulo Invertido (LIPM – Linear Inverted Pendulum). Em consequência, as coordenadas do centro de massa do robô (COM) devem passar por dois sistemas de controle. Após a execução dos sistemas ZMP e LIPM (em separado), as saídas dos sistemas ZMP e LIPM estão apresentadas no Gráfico 1:

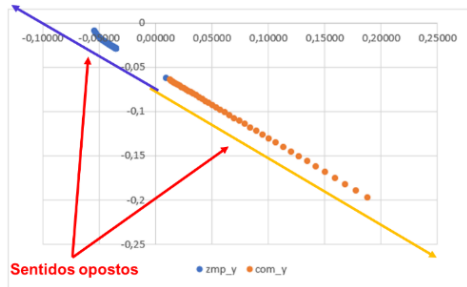


Gráfico 1 - Diferença entre as saídas ZMP e LIPM

Observa-se que a saída do sistema LIPM é contrária à saída do sistema ZMP, pois esta última serve como freio para o primeiro sistema, devendo estar dentro do polígono de suporte. Ambos iniciam no mesmo ponto e seguem direções opostas.

Sobrepondo os pés do robô no gráfico, e fazendo a cinemática inversa **apenas** na saída do sistema de controle LIPM (com), verifica-se o resultado na figura 14, onde as coordenadas ZMP ficam dentro do polígono de suporte.

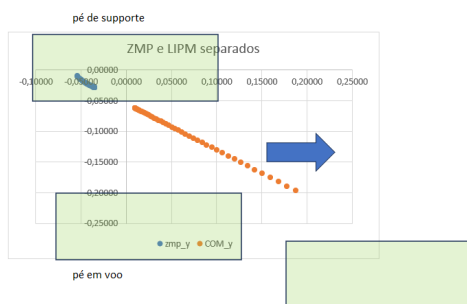


Figura 14 - Posição final do robô em relação às saídas dos sistemas ZMP e LIPM separadas

Colocando ambos os controles **operando em conjunto**, de forma que as coordenadas ZMP são subtraídas da LIPM, tem-se a correção da coordenada do COM final, linha cinza, (figura 15).

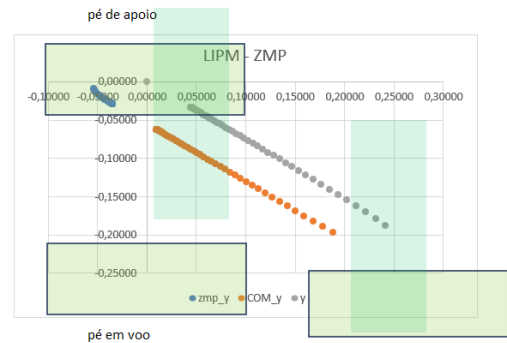


Figura 15 – Subtraindo o ZMP da saída LIPM

Conforme a figura 16, no início da caminhada, os pés estão sobre a superfície (lado esquerdo). A caminhada inicia erguendo o pé esquerdo (voo), e utilizando o pé direito como suporte no chão. O COM se desloca em direção ao pé de apoio, e faz o trajeto em direção ao pé esquerdo. Quando o pé esquerdo estiver no chão, o COM estará sobre o mesmo, e assim sucessivamente.

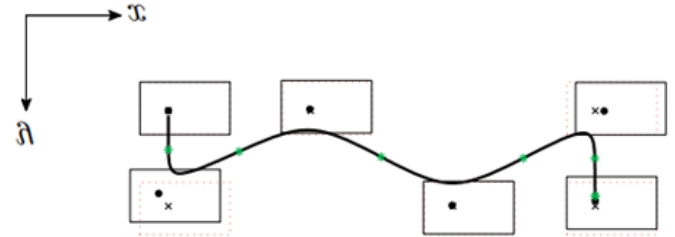


Figura 16 – Caminhada do robô utilizando o ZMP
Fonte: Kajita, 2014

REFERENCIAS

- [1] Shuuji Kajita, Hirohisa Hirukawa, Kensuke Harada, and Kazuhito Yokoi, "Introduction to Humanoid Robotics", Springer Trats in Advanced Robotics 101. London, 2014.
- [2] Marcos R. O. A. Maximo, Carlos H. C. Ribeiro, Rubens J.M. Afonso. "Reference ZMP manipulation for energetic and computationally efficient walking using ZMP preview control", XIII Simpósio Brasileiro de Automação Inteligente.RGS, Porto Alegre, 2017.
- [3] Igor Danilov, Bulat Gabbasov, Ilya Afanasyev, and Evgeni Magid." ZMP trajectory from human body locomotion dynamics evaluated by Kinect-based motion capture system". International Conference on Computer Vision Theory and Applications. Italy, Rome. 2016.
- [4] Timofte Gabriel, Man-Wook Han." Control of a humanoid robot based on the ZMP method". IFAC Proceedings Volumes, 2008.
- [5] Shuuji Kajita, Takashi Nagasaki, Kenji Kaneko, Hirohisa Hirukawa. "ZMP-Based Biped Running Control". IEEE Robotics & Automation Magazine (Volume: 14, Issue: 2, June 2007).
- [6] Shuuji Kajita; Mitsuharu Morisawa; Kensuke Harada; Kenji Kaneko; Fumio Kanehiro; Kiyoshi Fujiwara; Hirohisa Hirukawa." Biped Walking Pattern Generator allowing Auxiliary ZMP Control". IEEE/RSJ International Conference on Intelligent Robots and Systems October 9- 15, 2006, Beijing, China.
- [7] Mattias Wahde and Jimmy Pettersson." A brief review of bipedal robotics research". Proceedings of the 8th UK Mechatronics Forum International Conference, 2002.