

Control con rechazo activo de perturbaciones para el equilibrio de robots humanoides

S.M. Orozco-Soto, J.M. Ibarra-Zannatha

Centro de Investigación y Estudios Avanzados
del Instituto Politécnico Nacional (CINVESTAV),
México

orozco@ctrl.cinvestav.mx, jibarra@cinvestav.mx

Resumen. En este trabajo se presenta una estrategia de control para el equilibrio de robots humanoides mientras se encuentra de pie sin caminar, utilizando rechazo activo de perturbaciones. Dicha técnica de control regula la posición del ZMP del humanoide para mantenerlo dentro del polígono de soporte por medio de un regulador cuadrático lineal en tiempo discreto adicionado con detección y corrección de perturbaciones en línea basadas en un observador de estado extendido. La estrategia de control propuesta se implementó en un robot humanoide Nao utilizando lenguaje *Python*, mostrando resultados satisfactorios al mantener el robot de pie a pesar de ser sometido a fuerzas externas que alteran la posición de su ZMP. Los resultados antes mencionados motivan al uso de esta técnica de control para el rechazo de perturbaciones durante otras actividades complejas que realizan los robots humanoides, como el caminar, o su aplicación en el control de otro tipo de sistemas robóticos.

Palabras clave: robots humanoides, control ADRC, control basado en ZMP, control de equilibrio.

Active Disturbance Rejection Control for Humanoid Robots Balance

Abstract. This paper presents a control strategy for the balance of humanoid robots while they are standing without walking, using the active disturbance rejection control approach. Such control technique regulates the position of the ZMP of the humanoid in order to keep it within the support polygon by means of a discrete-time linear quadratic regulator enhanced with on-line disturbance detection and correction using an extended-state observer. The proposed control strategy was implemented on a Nao humanoid robot using *Python* and showing successful results by keeping the robot standing despite of external forces that affect the position of its ZMP. The results encourage the use of this controller for disturbance rejection meanwhile other complex activities are performed by humanoids, such as walking, or its application in the control of different kinds of robots.

Keywords: humanoid robots, ADRC control, ZMP-based control, balance control.

1. Introducción

Los robots humanoides son máquinas desarrolladas para operar en los mismos entornos en los que los seres humanos llevan a cabo sus actividades [1], en los cuales sufren diferentes perturbaciones inherentes a las tareas que realizan mientras están de pie o mientras caminan, además de que dichos entornos son dinámicos y por tal motivo, el humanoide debe ser capaz de reaccionar para mantenerse en equilibrio. En la literatura se encuentran reportadas diversas técnicas que le permiten a un robot humanoide permanecer en equilibrio mientras está de pie, como los controladores reportados en [2,3,4], en los cuales se utilizan los conceptos de centro de presión y reguladores cuadráticos lineales, además de otras técnicas que se utilizan mientras el robot camina, tales como las reportadas en [5], y otras más que se han utilizado para el rechazo de perturbaciones como las presentadas en [1,6,7,8].

En este trabajo, se propone el uso de una técnica de control llamada *Control con rechazo activo de perturbaciones* para garantizar el balance de robots humanoides mientras está de pie realizando alguna tarea. Esta técnica ha sido utilizada recientemente para distintas aplicaciones debido a que ofrece resultados satisfactorios con un costo computacional bajo, implementación rápida y con análisis matemático moderado debido a que está basada en las formas canónicas y en observadores de estado para la detección y corrección en línea de las señales de perturbación [9,10].

El controlador propuesto regula la posición del punto de momentos cero (ZMP) para mantenerla dentro del polígono de soporte del robot y así asegurar su equilibrio mientras está de pie; asimismo, la estrategia de control fue diseñada utilizando modelos simplificados de robots humanoides reportados en la literatura, además de incluir técnicas de control óptimo adicionadas con el rechazo activo de perturbaciones basado en observadores de estado extendido en tiempo discreto. La estrategia de control abordada se implementó en un robot humanoide Nao de *Aldebaran Robotics* y fue programada en lenguaje *Python* por la compatibilidad con dicha plataforma robótica.

Este trabajo está organizado de la siguiente manera: En la sección 2 se aborda brevemente la teoría del control con rechazo activo de perturbaciones. En la sección 3 se describe el robot humanoide y su modelo matemático. La sección 4 está dedicada al diseño del controlador propuesto. En la sección 5 se presentan los resultados de la implementación y, finalmente, en la sección 6 se presentan las conclusiones.

2. Control con rechazo activo de perturbaciones

El Control con Rechazo Activo de Perturbaciones (ADRC por *Active Disturbance Rejection Control*) es una estrategia de control capaz de manejar diversas incertidumbres paramétricas y perturbaciones de forma eficiente desde el punto de vista energético y de implementación de un sistema de control, debido a que está basado en el aprovechamiento de dos de las mejores prestaciones de la teoría de control moderna: la representación de sistemas en formas canónicas y los observadores de estado [9]. En la Fig. 1 se ilustra un diagrama a bloques del esquema de control ADRC en donde se puede apreciar que el controlador utiliza dos tipos de realimentación: la perturbación estimada o detectada por el observador de estado extendido y la salida de la planta.

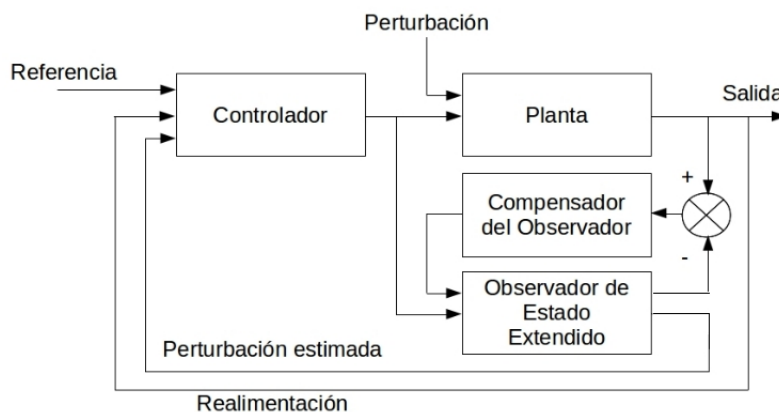


Fig. 1. Diagrama a bloques del ADRC.

2.1. Teoría general del ADRC

Sea el siguiente sistema dinámico sometido a perturbaciones:

$$\begin{aligned} \dot{x} &= f(t, x, u, w), \\ y &= h(t, x, u, w), \end{aligned} \tag{1}$$

donde $x \in \mathbb{R}^n$ es el vector de estado, $u \in \mathbb{R}^p$ es el vector de entradas y $w \in \mathbb{R}^q$ es el vector de perturbaciones, el cual es usualmente desconocido pero acotado. Independientemente de su naturaleza lineal o no lineal, este sistema puede ser representado bajo ciertas transformaciones difeomórficas del tipo $z = T(x)$ como una cadena de integradores de la forma:

$$\begin{aligned}
\dot{z}_1 &= z_2, \\
&\vdots \\
\dot{z}_{n-1} &= z_n, \\
\dot{z}_n &= g(t, x, u, w), \\
y &= h(z).
\end{aligned} \tag{2}$$

Considerando \dot{z}_n como una perturbación total, es posible construir un observador de estado extendido (ESO), como el mostrado en la Fig. 1, expresado como:

$$\begin{aligned}
\dot{\hat{z}}_1 &= \hat{z}_2 - l_1(\hat{e}), \\
&\vdots \\
\dot{\hat{z}}_{n-1} &= \hat{z}_n - l_n(\hat{e}), \\
\dot{\hat{z}}_n &= g(\hat{z}, t, u, w) - l_{n+1}(\hat{e}),
\end{aligned} \tag{3}$$

donde $\hat{e} = \hat{z} - z$ y $l(\hat{e})$ es una función vectorial que hace $\hat{z} \rightarrow z$ asintóticamente. Entonces, se propone una ley de control basado en el observador (3) de la forma:

$$u = u_0 - K_s \hat{z}, \tag{4}$$

donde K_s es la ganancia del controlador con realimentación de estados observados y u_0 es un controlador para una de cadena de integradores tal que:

$$u_0 \approx \dot{z}_{n-1}. \tag{5}$$

Cabe mencionar que u_0 considera la transformación difeomórfica inversa $x = T^{-1}(z)$. Además, este controlador no requiere un conocimiento preciso de la perturbación generalizada ni de los parámetros del modelo de la planta, solo del orden de la misma.

3. Descripción del sistema robótico

Un robot humanoide es un sistema altamente no lineal con dinámicas acopladas y complejas, el cual, además, cuenta con múltiples grados de libertad (GDL) que conforman una cadena cinemática arborescente. Debido a dichas características, la tarea de modelar matemáticamente este tipo de sistemas robóticos adquiere un grado de complejidad significativamente alto, por lo que se utilizan métodos de tipo *coarse graining* para simplificar los modelos y que continúen siendo de utilidad en el diseño de los sistemas de control.

Algunos de los modelos más representativos que se encuentran en la literatura son los tipo péndulo invertido lineal (PIL) o los modelos tipo carro-mesa [5],[11]. En este trabajo se utiliza el modelo tipo carro-mesa debido a que los experimentos se llevaron a cabo en un robot humanoide Nao de *Aldebaran Robotics*, el cual cumple con las condiciones requeridas por este modelo de ser un robot completamente actuado y con sus articulaciones controladas en posición [11].

3.1. Robot humanoide Nao

El robot Humanoide Nao es una plataforma de pequeño formato y altas prestaciones conformado por un par de piernas con 5 GDL en cada una, un par de brazos de 3 GDL (versión *Robocup*), una cabeza con 2 GDL y el torso, al cual se le asocia el referencial principal y el cual también está actuado por dos servomotores que ejecutan consignas de 2 GDL simultáneamente. En la Fig. 2 se ilustra el cuerpo completo del humanoide y de las articulaciones de la pelvis que accionan el torso, debido a que son las que se utilizan para la ejecución del controlador propuesto en este artículo. Además, el robot Nao está equipado con un sistema de percepción que incluye diferentes sensores como unidad de medición inercial (IMU), sensores de fuerza resistivos (FSR) en los pies, entre otros, los cuales se utilizan para realimentación del controlador de este trabajo.

3.2. Modelo del robot humanoide

Para llevar a cabo el diseño del controlador para equilibrio, se utilizó el modelo de tipo carro-mesa, el cual considera la masa total del robot como un carro en donde se concentra la masa total del humanoide, el cual corre sobre una mesa de masa despreciable, tal como se ilustra en la Fig. 3, en donde se puede apreciar una altura constante z_c del centro de masa, así como las coordenadas generalizadas de posición y aceleración del carro. A pesar de que el pie de la mesa es significativamente pequeño, si el carro se desplaza con cierta aceleración es posible conseguir balanceo instantáneo; asimismo, la posición x del carro determina el centro de presión que actúa en el suelo, situación que conduce a la siguiente definición:

Definición 3.1: Punto de momentos cero (ZMP) *Punto donde las componentes horizontales del momento de las fuerzas de reacción con el suelo es igual a 0, expresado matemáticamente, en el caso del eje x como:*

$$p_x = x - \frac{z_c}{g} \ddot{x}. \quad (6)$$

Para garantizar que el robot permanezca equilibrado de pie, el ZMP, mismo que tiene unidades de distancia, siempre debe existir dentro del polígono de soporte formado por sus huellas. En la Fig. 3 se ilustra también la ubicación del punto p_x , el cual determina si el par $\tau_{ZMP} = 0$, indicando si el robot está o no balanceado. El sistema carro-mesa se puede representar de la siguiente forma a partir de la ecuación (6) utilizando $ZMP = p_x$ como estado y $u = \ddot{x}$ como entrada de control [5]:

$$\frac{d}{dt} \begin{bmatrix} p_x \\ x \\ \dot{x} \end{bmatrix} = \begin{bmatrix} -1/T & 1/T & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_x \\ x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} -z_c/g \\ 0 \\ 1 \end{bmatrix} u, \quad (7)$$

$$y = [1 \ 0 \ 0] \begin{bmatrix} p_x \\ x \\ \dot{x} \end{bmatrix}.$$

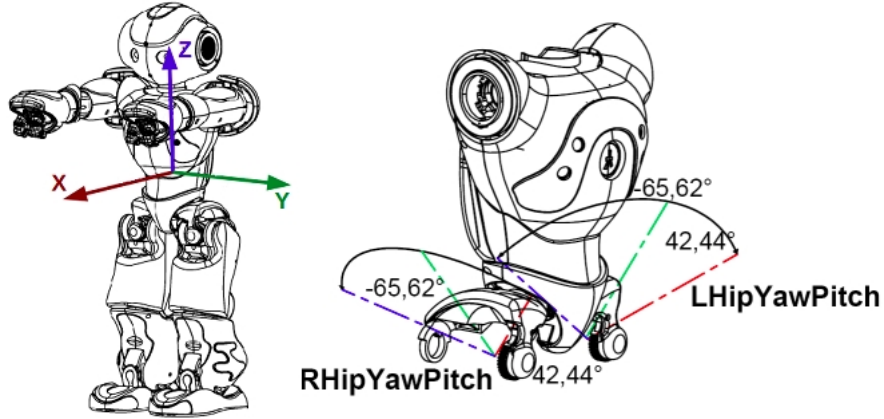


Fig. 2. Cuerpo completo del humanoide Nao y sus articulaciones de la pelvis [12].

4. Diseño del controlador

Considerar los estados del sistema (7) representados en tiempo discreto: $x_1(k) = p_x(k)$ es el ZMP, $x_2(k) = x(k)$ es la posición instantánea del centro de masa (carro), $x_3(k) = \dot{x}(k)$ es la velocidad del carro y $x_4(k) = w(k)$ es la perturbación detectada. De esta forma, representación de estado extendido es:

$$x(k+1) = \begin{bmatrix} (1-T) & T & T^2/2 & 0 \\ 0 & 1 & T & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} -(z_c T^3)/(6g) \\ T^2/2 \\ T \\ 0 \end{bmatrix} u(k), \quad (8)$$

donde T es el periodo de muestreo. A continuación se presenta la forma en como se calculan los estados a partir de las mediciones de los sensores del robot.

4.1. Cálculo del ZMP y de la posición del centro de masa

La ubicación del ZMP para el robot humanoide Nao se calcula de la siguiente forma:

$$zmp = \frac{1}{\sum_1^4 F_{Lfsr_i}} \begin{bmatrix} \sum_1^4 F_{Lfsr_i} dL_{x_i} \\ \sum_1^4 F_{Lfsr_i} dL_{y_i} \end{bmatrix} + \frac{1}{\sum_1^4 F_{Rfsr_i}} \begin{bmatrix} \sum_1^4 F_{Rfsr_i} dR_{x_i} \\ \sum_1^4 F_{Rfsr_i} dR_{y_i} \end{bmatrix}, \quad (9)$$

donde F_{Lfsr_i} es la fuerza del i -ésimo FSR del pie izquierdo, F_{Rfsr_i} representa dicha fuerza para el pie derecho, dL_{y_i} es la ubicación en el eje y del i -ésimo FSR del pie izquierdo respecto a un referencial arbitrario, dL_{x_i} es la ubicación mencionada pero en el eje x , dR_{y_i} y dR_{x_i} son sus contrapartes del pie derecho.

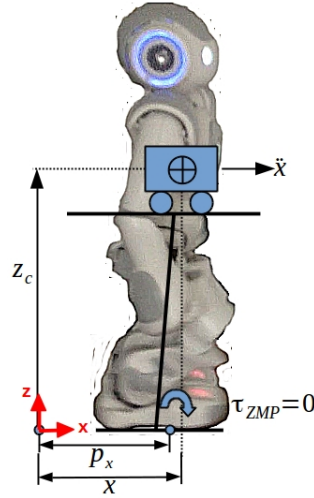


Fig. 3. Humanoide Nao representado como un sistema carro-mesa.

En este trabajo, solo se considera el ZMP en el eje x , por lo que se consideran las componentes en $y = 0$ y se asumirá que $p_x = (6)$. En la Fig. 4 se ilustra el concepto de ubicación de los FSR utilizado para el cálculo del ZMP, además, se puede observar el referencial a partir de donde se calculan las distancias de dichas ubicaciones. Asimismo, la posición instantánea $x(k)$ del carro sobre la mesa se calcula de la siguiente forma a partir de la ecuación (6):

$$x(k) = p_x(k) + \frac{z_c}{g} acc_x(k), \quad (10)$$

donde $acc_x(k)$ se obtiene a partir de las mediciones realizadas con el acelerómetro integrado en la IMU del robot. La velocidad $x(k + 1)$ se puede obtener con técnicas de filtrado, pero en este caso, se utiliza la señal calculada con el observador.

4.2. Diseño del controlador ADRC

Considerar el modelo del robot humanoide (8), para el cual, se propone el siguiente (ESO):

$$\hat{x}(k+1) = \begin{bmatrix} (1-T) & T & T^2/2 & 0 \\ 0 & 1 & T & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \hat{x}(k) + \begin{bmatrix} -(z_c T^3)/(6g) \\ T^2/2 \\ T \\ 0 \end{bmatrix} u(k) + \begin{bmatrix} \tanh(x_1(k) - \hat{x}_1(k)) \\ \tanh(x_2(k) - \hat{x}_2(k)) \\ \tanh(x_3(k) - \hat{x}_3(k)) \\ \tanh(w(k) - \hat{w}(k)) \end{bmatrix}, \quad (11)$$

donde $w(k) = x_1(k) - \hat{x}_1(k)$. Para rechazar las perturbaciones detectadas se propone la siguiente ley de control:

$$u(k) = u_0(k) - K_o \hat{w}(k), \quad (12)$$

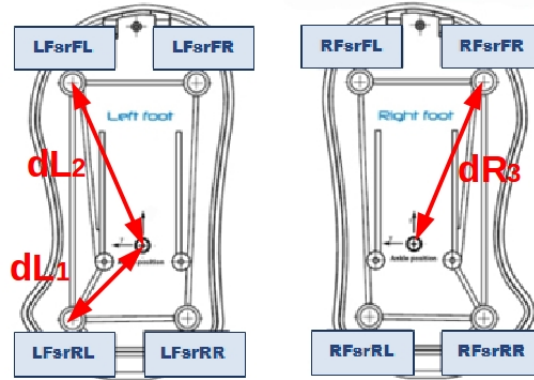


Fig. 4. Ubicación y distancias de los FSR del robot.

donde $K_o \in \mathbb{R}^+$ y u_0 es un controlador para las variables de estado detallado a continuación.

4.3. Diseño del controlador por realimentación de estado

Considerar el sistema (8), el cual es una representación en tiempo discreto extendida de (7). Si se omite el estado $w(k)$, es decir, el sistema sin perturbaciones, se obtiene la siguiente representación de estado:

$$x(k+1) = Ax(k) + Bu(k), \quad (13)$$

$$y(k) = [1 \quad 0 \quad 0]x(k) = p_x(k),$$

donde $A = \begin{bmatrix} (1-T) & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}$ y $B = \begin{bmatrix} -(z_c T^3)/(6g) \\ T^2/2 \\ T \end{bmatrix}$.

Entonces, para que el ZMP del robot siga al ZMP objetivo p_x^{ref} y de esta manera mantener al robot balanceado mientras está de pie, con consumo de energía óptimo, se propone minimizar la siguiente función de costo [5]:

$$J = \sum_{j=1}^{\infty} \left\{ Q(p_{x_j}^{ref} - p_{x_j})^2 + Ru_j^2 \right\}, \quad (14)$$

donde $Q \geq 0 \in \mathbb{R}^{3 \times 3}$ y $R \geq 0$ son constantes de ponderación. Minimizar la función de costo J se traduce en resolver la siguiente ecuación de Riccati:

$$A^T S + SA - SBR^{-1}B^T S + Q = 0. \quad (15)$$

Con lo cual se asegura la estabilidad del sistema en el sentido de Lyapunov. Además, $S = S^T > 0$ es una incógnita. Con esta solución, es posible calcular la ganancia K del regulador cuadrático lineal en tiempo discreto (DLQR) $u_0(k) = Kx(k)$ de la forma:

$$K = R^{-1}B^T S. \quad (16)$$

Finalmente, el controlador ADRC + DLQR general se puede expresar como:

$$u(k) = Kx(k) - K_o\hat{w}(k). \quad (17)$$

4.4. Implementación del controlador

Para llevar a cabo la implementación del control de seguimiento de ZMP, se utilizaron los parámetros del modelo del robot humanoide Nao mostrados en la Tabla 1:

Tabla 1. Parámetros del modelo del robot humanoide.

Parámetro	Símbolo	Valor	Unidades
Masa total del robot	m	5.182530	Kg
Altura del CdM	z_c	0.335	m
Aceleración de la gravedad	g	9.81	m/s ²
Periodo de muestreo	T	100	ms

Los parámetros del DLQR utilizados son los siguientes:

$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, R = 1e - 3, P = \begin{bmatrix} 480.026 & 234.299 & 22.775 \\ 234.299 & 1244.482 & 46.328 \\ 22.775 & 46.328 & 34.337 \end{bmatrix},$$

$$K = [6.2917(g/z_c)T \quad 37.7014T \quad 32.5359T], \quad (18)$$

$$z = \begin{bmatrix} 0.96887 \\ 0.99889 + 0.00044i \\ 0.99889 - 0.00044i \end{bmatrix},$$

donde z son los polos en lazo cerrado del sistema, los cuales son estables, además, como Q y R ponderan el efecto de la dinámica de la planta y el gasto de energía asociado a la señal de control respectivamente, se eligieron tales que dicha entrada de control consuma la menor cantidad de energía posible.

Cabe mencionar que el controlador se programó en lenguaje *Python* debido a que es compatible con la plataforma Nao. La señal de control se implementó utilizando los actuadores para mover las articulaciones de la cadera mostradas en la Fig. 2, las cuales, una vez detectada la perturbación por medio del ESO, calculan la aceleración que debe suministrarse en sentido contrario dicha perturbación.

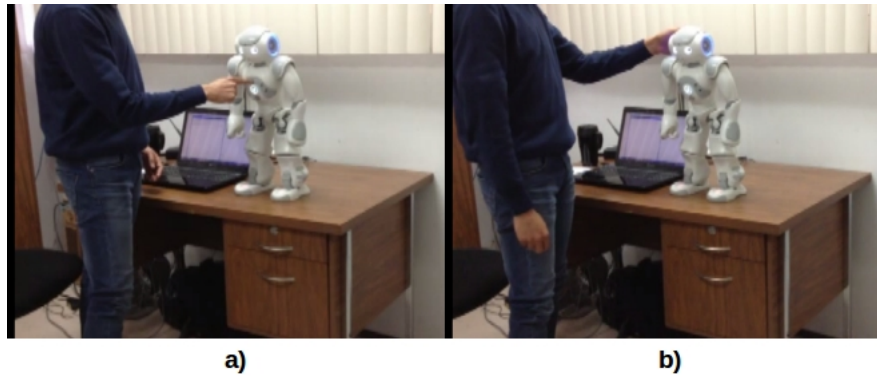


Fig. 5. Perturbaciones aplicadas al humanoide. a) Perturbación frontal. b) Perturbación trasera. Video disponible en [Video disponible en https://www.youtube.com/watch?v=B9_JOH3KX0I](https://www.youtube.com/watch?v=B9_JOH3KX0I).

5. Resultados

Como se mencionó en la sección anterior, el controlador ADRC propuesto se implementó utilizando lenguaje *Python* en el robot humanoide. Se propuso una posición objetivo del ZMP en $p_x^{ref} = 0.02m$ respecto al referencial de la IMU del robot. El experimento consistió en llevar el ZMP del robot por medio de la inclinación del torso a la consigna mencionada anteriormente, para posteriormente, aplicar una fuerza relativamente constante empujando al robot durante 2 segundos en el pecho como se muestra en la Fig. 5 a), al igual que en la espalda del robot también durante 2 segundos, tal como se ilustra en la Fig. 5 b), con el objetivo de perturbar la posición del ZMP y que el robot se mantenga de pie. Este experimento se ejecutó durante 20 segundos con el control DLQR y 20 segundos con el controlador ADRC propuesto.

En la Fig. 6 a) se presentan los resultados del experimento realizado con control DLQR, en donde se puede apreciar que el humanoide presentó oscilaciones posteriores a la perturbación frontal, además de presentar mayor número de oscilaciones y de mayor amplitud al aplicarse la perturbación trasera.

Asimismo, se puede observar la señal de perturbación detectada en la Fig. 6 b). En la Fig. 6 c) y d) se presentan los estados calculados y estimados tanto de posición como velocidad respectivamente, en donde se aprecia que el observador propuesto funciona correctamente para la estimación de posición.

En la Fig. 7 a) se presenta el comportamiento del ZMP con control ADRC, en donde se observan menor cantidad de oscilaciones y de menor magnitud para ambas perturbaciones, mismas que se ilustran en la Fig. 7 b). En la Fig. 7 c) se ilustra el observador funcionando correctamente para la estimación de posición.

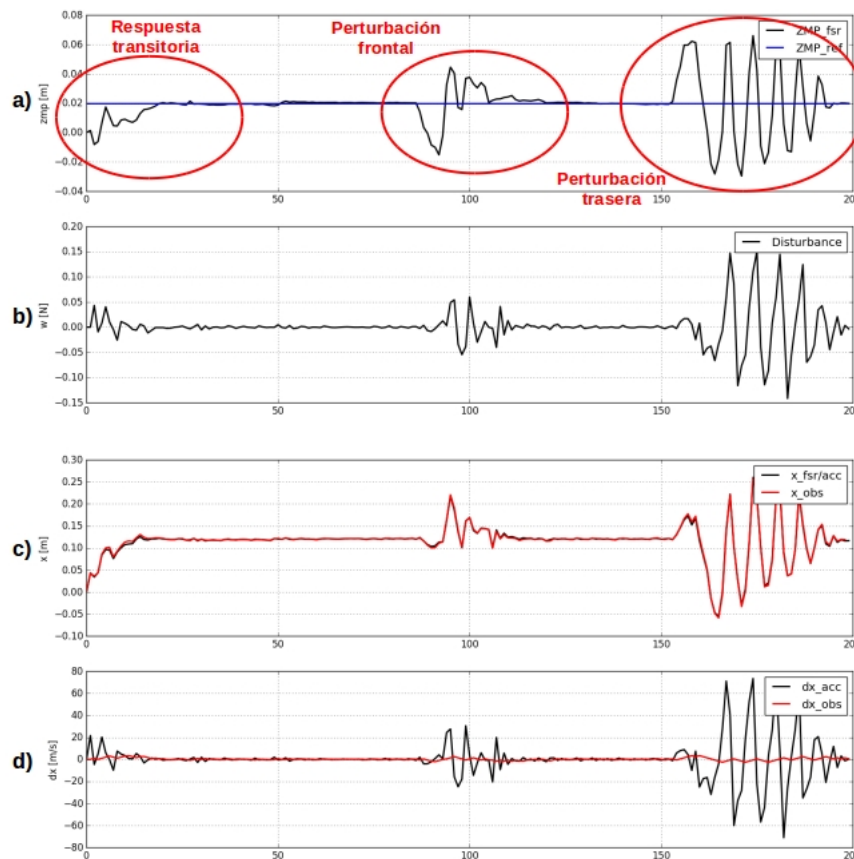


Fig. 6. Resultados con control DLQR. a) Control del ZMP. b) Perturbación detectada $w(k)$. c) Posiciones $x(k)$ medida y estimada. d) Velocidades $x(k+1)$ medida y estimada.

6. Conclusiones

En este trabajo se presentó un controlador en tiempo discreto para el equilibrio de robots humanoides utilizando la técnica ADRC. Este controlador es capaz de detectar las perturbaciones a las que está sometido un sistema en línea y de corregirlas gracias a un observador de estado extendido.

Dicha estrategia de control se diseñó utilizando el modelo carro-mesa de robots humanoides con el objetivo de regular la posición del ZMP y así garantizar el equilibrio del robot aún en presencia de perturbaciones. El controlador propuesto se implementó en un robot humanoide Nao programado en lenguaje *Python* y se comparó su desempeño con un regulador cuadrático lineal, mostrando mejores

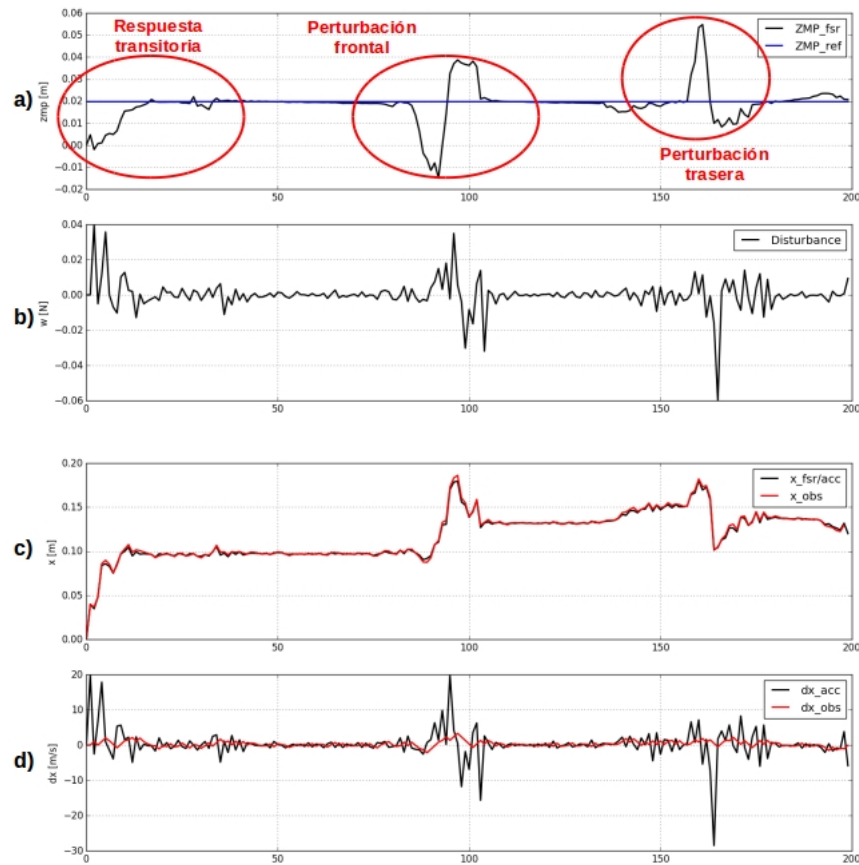


Fig. 7. Resultados con control DLQR. a) Control del ZMP. b) Perturbación detectada $w(k)$. c) Posiciones $x(k)$ medida y estimada. d) Velocidades $x(k+1)$ medida y estimada.

resultados el ADRC debido a que el robot oscila en menor cantidad y con menor magnitud ante las perturbaciones gracias a que el observador de estado detecta las perturbaciones y suministra la energía adecuada para corregirlas.

Estos resultados satisfactorios de la implementación del control ADRC para el equilibrio de robots humanoides motivan a su implementación para el caminado del robots Nao, así como para su uso en otros humanoides instrumentados adecuadamente.

Referencias

1. Orozco-Soto, S. M., Núñez-Cruz, R. S., Ibarra-Zannatha, J. M.: Active disturbance rejection control for humanoid stable walking. In: 13th IEEE International Conference on Electrical Engineering, Computing Science and Automatic Control, pp. 1–6, IEEE Press, Ciudad de México (2016)
2. Nakaura, S., Sampei, M.: Balance control analysis of humanoid robot based on ZMP feedback control. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 3 pp. 2437–2442, IEEE Press, Suiza (2002)
3. Stephens, B.: Integral control of humanoid balance. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4020–4027, IEEE Press, San Diego (2007)
4. Lee, S. H., Goswami, A.: Ground reaction force control at each foot: A momentum-based humanoid balance controller for non-level and non-stationary ground. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3157–3162, IEEE Press, Taipei (2010)
5. Kajita, S., Hirukawa, H., Harada, K., Yokoi, K.: Introduction to humanoid robotics. Springer, Heidelberg (2014)
6. Hyong, S., Cheng, G.: Disturbance rejection for biped humanoids. In: IEEE International Conference on Robotics and Automation, pp. 2668–2675, IEEE Press, Roma (2007)
7. Alcaraz-Jiménez, J. J., Missura, M., Martínez-Barberá, H., Behnke, S.: Lateral disturbance rejection for the Nao robot. In: RoboCup 2012: Robot Soccer World Cup XVI, 1–12, Springer, Heidelberg (2012)
8. Orozco Soto, S.M., Ibarra Zannatha, J.M.: Control con rechazo activo de perturbaciones para el caminado estable de humanoides basado en el pndulo invertido lineal. En: XVIII Congreso Mexicano de Robótica, pp. 1–6, Asociación Mexicana de Robótica e Industria, Mazatlán (2016)
9. Huang, Y., Xue, W.: Active disturbance rejection control: methodology and theoretical analysis. *ISA Transactions* 53(4), 963–976 (2014)
10. Zheng, Q., Gao, Z.: On practical applications of active disturbance rejection control. In: IEEE 29th Chinese Control Conference, pp. 6095–6100, Beijing (2010)
11. Dallali, H., Brown, M., and Vanderborght, B.: Using the torso to compensate for non-minimum phase behaviour in ZMP bipedal walking. In: *Advances in Robotics Research*, pp. 191–202, Springer, Heidelberg (2009)
12. NAO Software 1.14.5 documentation <http://doc.aldebaran.com/1-14/contents.html>