$H_\infty$-Controller Design Methods Applied to One Joint of a Flexible Industrial Manipulator

Patrik Axelsson, Anders Helmersson and Mikael Norrlöf

Linköping University Post Print

N.B.: When citing this work, cite the original article.

Original Publication:
Series; World Congress, ISSN 1474-6670; Volume# 19, Part# 1

http://dx.doi.org/10.3182/20140824-6-ZA-1003.00143

International Federation of Automatic Control
http://www.ifac-papersonline.net/

Postprint available at: Linköping University Electronic Press
http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-104785
Abstract: Control of a flexible joint of an industrial manipulator using \( H_\infty \) design methods is presented. The considered design methods are i) mixed-\( H_\infty \) design, and ii) \( H_\infty \) loop shaping design. Two different controller configurations are examined: one uses only the actuator position, while the other uses the actuator position and the acceleration of the end-effector. The four resulting controllers are compared to a standard PID controller where only the actuator position is measured. The choices of the weighting functions are discussed in details. For the loop shaping design method, the acceleration measurement is required to improve the performance compared to the PID controller. For the mixed-\( H_\infty \) method it is enough to have only the actuator position to get an improved performance. Model order reduction of the controllers is briefly discussed, which is important for implementation of the controllers in the robot control system.

Keywords: Robotics, Flexible, H-infinity control, Accelerometers

1. INTRODUCTION

The requirements for controllers in modern industrial manipulators are that they should provide high performance, at the same time, robustness to model uncertainty. In the typical standard control configuration the actuator positions are the only measurements used in the higher level control loop. At a lower level, in the drive system, the currents and voltages in the motor are measured to provide torque control for the motors. In this contribution different \( H_\infty \)-controller design schemes are compared when using two different sensor configurations. First, the standard case where only the position of the actuator rotation is used, and second a configuration where, in addition, the acceleration of the tool tip is measured. Two different \( H_\infty \) methods are investigated: i) loop shaping [McFarlane and Glover, 1992], and ii) multi-\( H_\infty \) design [mixedHinfsyn, 2013, Zavari et al., 2012].

Motivated by the conclusions from Sage et al. [1999] regarding the area of robust control applied to industrial manipulators, this contribution includes:

- results presented using realistic models,
- a comparison with a standard PID control structure,
- model reduction of the controllers to get a result that more easily can be implemented in practice.

The model used in this contribution represents one joint of a typical modern industrial robot [Moberg et al., 2009]. It is a physical model consisting of four masses, which should be compared to the typical two-mass model used in many previous contributions, see Sage et al. [1999] and the references therein. The joint model represents the first joint of a serial 6-DOF industrial manipulator, where the remaining five axes have been configured to minimise the couplings to the first axis. To handle changes in the configuration of the remaining axes, gain scheduling techniques can be used based on the results in this paper.

An important part of the design is the choice of the weighting functions, which is an essential task to get a satisfactory performance. The work of choosing the weights is difficult, tedious and time consuming. This can be the reasons for why \( H_\infty \) methods are not used that often in practice even though the performance and robustness can be increased. In particular, the use of two measurements for control of one variable requires special treatment. The development of the weighting functions for the four controllers are discussed in details, and provides a significant part of the contributions in the paper.

Controller synthesis using \( H_\infty \) methods has been proposed in Song et al. [1992], Stout and Sawan [1992], where the complete non-linear robot model first is linearised using exact linearisation, second an \( H_\infty \) controller is designed using the linearised model. The remaining non-linearities due to model errors are seen as uncertainties and/or disturbances. In both papers, the model is rigid and the \( H_\infty \) controller, using only joint positions, is designed using the mixed-sensitivity method. In Sage et al. [1997] \( H_\infty \) loop shaping with measurements of the actuator positions is applied to a robot. The authors use a flexible joint model which has been linearised. The linearised model makes it possible to use decentralised control, hence \( H_\infty \) loop shaping is applied to n SISO-systems instead of the complete MIMO-system.
Explicit use of acceleration measurements for control in robotic applications has been reported in, for example, de Jager [1993], Dumetz et al. [2006], Kosuge et al. [1989], Readman and Bélanger [1991] and Xu and Han [2000]. In Dumetz et al. [2006], a control law using motor position and acceleration of the load in the feedback loop is proposed for a Cartesian robot. The robot is assumed to be flexible and modelled as a two-mass system, where the masses are connected by a linear spring-damper pair. Another control law of a Cartesian robot using acceleration measurements is presented in de Jager [1993]. The model is a rigid joint model and the evaluation is made both in simulation and experiments.

In Kosuge et al. [1989] a 2-degree-of-freedom (DOF) manipulator is controlled using acceleration measurements of the end-effector. The model is assumed to be rigid and it is exactly linearised. The joint angular acceleration used in the non-linear feedback loop is calculated using the inverse kinematic acceleration model and the measured acceleration. The use of direct measurements of the angular acceleration in the feedback loop is presented in Readman and Bélanger [1991] for both rigid and flexible joint models. A more recent work is presented in Xu and Han [2000], where a 3-DOF manipulator is controlled using only measurements of the end-effector acceleration.

The theory for synthesis of $H_\infty$ controllers is presented in Section 2. The model describing the robot joint is explained in Section 3. In Section 4, the requirements of the system as well as the design of four controllers are described, and in Section 5 the simulation results are shown. Finally, Section 6 discuss low order controller synthesis and Section 7 concludes the work.

2. CONTROLLER DESIGN METHODS

In this section, a brief introduction to mixed-$H_\infty$ design [mixedHinfsyn, 2013, Zavari et al., 2012] and $H_\infty$ loop shaping [McFarlane and Glover, 1992] will be presented.

2.1 Mixed-$H_\infty$ Controller Design

A common design method is to construct the system $P(s)$ in

\[
\begin{pmatrix}
  z \\
  y
\end{pmatrix} =
\begin{pmatrix}
  P_{11}(s) & P_{12}(s) \\
  P_{21}(s) & P_{22}(s)
\end{pmatrix}
\begin{pmatrix}
  w \\
  u
\end{pmatrix}
= P(s)
\begin{pmatrix}
  w \\
  u
\end{pmatrix}
\]

by augmenting the original system $y = G(s)u$ with the weights $W_u(s)$, $W_S(s)$, and $W_T(s)$, such that the system $z = F(P, K)$, depicted in Figure 1(a), can be written as

\[
F(P, K) = \begin{pmatrix}
  W_u(s)G_{uu}(s) \\
  -W_T(s)T(s) \\
  W_S(s)S(s)
\end{pmatrix},
\]

where $S(s) = (I + G(s)K(s))^{-1}$ is the sensitivity function, $T(s) = I - S(s)$ is the complementary sensitivity function, and $G_{uu}(s) = -K(s)(I + G(s)K(s))^{-1}$ is the transfer function from $w$ to $u$. The $H_\infty$-controller is then obtained by minimising the $H_\infty$-norm of the system $F(P, K)$, i.e., minimise $\gamma$ such that $\|F(P, K)\|_\infty < \gamma$. Using (2) gives

\[
\begin{align*}
|W_u(i\omega)G_{uu}(i\omega)| < \gamma, & \forall \omega, \quad (3a) \\
|W_T(i\omega)T(i\omega)| < \gamma, & \forall \omega, \quad (3b) \\
|W_S(i\omega)S(i\omega)| < \gamma, & \forall \omega. \quad (3c)
\end{align*}
\]

The transfer functions $G_{uu}(s)$, $S(s)$, and $T(s)$ can now be shaped to satisfy the requirements by choosing the weights $W_u(s)$, $W_S(s)$, and $W_T(s)$. The aim is to get a value of $\gamma$ close to 1, which in general is a hard to achieve and it requires insight in the design method as well as the system dynamics. For more details about the design method, see e.g. Skogestad and Postlætawe [2005], Zhou et al. [1996].

The mixed-$H_\infty$ controller design [mixedHinfsyn, 2013, Zavari et al., 2012] is a modification of the standard $H_\infty$ design method. Instead of choosing the weights in (2) such that the norm of all weighted transfer functions satisfies (3), the modified method divides the problem into design constraints and design objectives. The controller can now be found as the solution to

\[
\begin{align*}
\text{Minimise} & \quad \gamma_K(s) \quad (4a) \\
\text{subject to} & \quad \|W_P(s)S(s)\|_\infty < \gamma, \quad (4b) \\
& \quad \|M_S(s)\|_\infty < 1, \quad (4c) \\
& \quad \|W_u(s)G_{uu}(s)\|_\infty < 1, \quad (4d) \\
& \quad \|W_T(s)T(s)\|_\infty < 1, \quad (4e)
\end{align*}
\]

where $\gamma$ not necessarily has to be close to 1. Here, the weight $W_S(s)$ has been replaced by the weights $M_S(s)$ and $W_P(s)$. The method can be generalised to other control structures and in its general form it is formulated as a multi-objective optimisation problem. More details about the general form and how to solve the optimisation problem are presented in mixedHinfsyn [2013], Zavari et al. [2012].

2.2 Loop Shaping using $H_\infty$ Synthesis

For loop shaping [McFarlane and Glover, 1992], the system $G(s)$ is pre- and post-multiplied with weights $W_1(s)$ and $W_2(s)$, see Figure 1(b), such that the shaped system $G_s(s) = W_2(s)G(s)W_1(s)$ has the desired properties. The controller $K_s(s)$ is then obtained using the method described in Glover and McFarlane [1989] applied on the system $G_s(s)$, giving the controller $K_s(s)$. Finally, the controller $K(s)$ is given by

\[
K(s) = W_1(s)K_s(s)W_2(s). \quad (5)
\]

Note that the structure in Figure 1(b) for loop shaping can be rewritten as a standard $H_\infty$ problem according to Figure 1(a), see Zhou et al. [1996] for details. It will be used in Section 6 for synthesis of low order controllers.

The MATLAB function ncfsyn, included in the Robust Control Toolbox, is used in this paper for synthesis of $H_\infty$ controllers using loop shaping.
In (6), \( \eta \) is the gear ratio and \( l_1, l_2, \) and \( l_3 \) are the respective link lengths. Using Lagrange’s equation, the linearised flexible joint model can be described by a set of four ODEs, which can be formulated as a linear state space model according to
\[
\dot{x} = Ax + Bu + B_w w, \quad (7a)
\]
\[
y = Cx + Du + D_w w. \quad (7b)
\]
where the state vector and disturbance vector are given by
\[
x = (q_m \quad q_{a1} \quad q_{a2} \quad q_{a3} \quad \dot{q}_m \quad \dot{q}_{a1} \quad \dot{q}_{a2} \quad \dot{q}_{a3})^T. \quad (8a)
\]
\[
w = (w_m \quad w_P)^T. \quad (8b)
\]
The linear state space model is used for design of the H\(_\infty\) controllers. Note that the matrices \( C, D, \) and \( D_w \) are different depending on which sensor configuration is used, whereas the matrices \( A, B, \) and \( B_w \) stay the same.

### 3. Flexible Joint Model

The model considered in this paper is a four-mass benchmark model of a single flexible joint, see Figure 2, presented in Moberg et al. [2009]. The model corresponds to joint 1 of a serial 6-DOF industrial manipulator, where the five remaining axes are configured such that the couplings to joint 1 is minimised, see Moberg et al. [2009] for more details about the operating point where the model has been linearised.

Input to the system is the motor torque \( u \), the motor disturbance \( w_m \) and the end-effector disturbance \( w_P \). The four masses are connected by spring-damper pairs, where the first mass corresponds to the motor. The other masses represents distributed masses placed along the arm. The first spring-damper pair is modelled by a linear damper and non-linear spring, whereas the other spring-damper pairs are modelled as linear springs and dampers. The non-linear spring is characterised by a low stiffness for low deflections and a high stiffness for high deflections. This behaviour is typical for compact gear boxes, such as harmonic drive [Ruderman and Bertram, 2012]. For design of the H\(_\infty\) controllers, the non-linear model is linearised in one operating point in the high stiffness region, motivated by that a constant torque, e.g. gravity, is acting on the joint. Moreover, the friction torques are assumed to be linear and the input torque \( u \) is limited to \( \pm 20 \) Nm. The output of the system is the motor position \( q_m \) and the end-effector acceleration \( \ddot{P} \), where
\[
\ddot{P} = l_1 q_{a1} + l_2 q_{a2} + l_3 q_{a3}. \quad (6)
\]
the magnitude of the peaks a modified elliptic filter\(^2\)

\[
H(s) = \frac{0.5227s^2 + 3.266s + 1406}{s^2 + 5.808s + 2324}
\]  

(9) is introduced in \(W_2\). The weights are finally given as

\[
W_1(s) = 1, \quad W_2(s) = \frac{100s^2 + 10s}{s} H(s).
\]  

(10)

Using \texttt{nffsyn} a controller of order 13 is obtained.

4.2.2. \(H_{\infty}^m(q_m, \hat{P})\): Adding an extra measurement signal in terms of the acceleration of the end-effector gives a system with one input and two outputs. For stability reasons, it is not possible to have an integrator in both control channels. Therefore, the integrator is placed in the channel for \(q_m\), since the accelerometer measurement has low frequency noise, such as drift. For the same reason as for the other controller, a zero in \(s = -3\) is introduced. The transfer function from input torque to acceleration of the end-effector has a high gain in the frequency range of interest. To decrease the gain such that it is comparable with the motor angle measurement, a low pass filter is added in the acceleration channel. The final weights are

\[
W_1(s) = 50, \quad W_2(s) = \text{diag} \left( \frac{s^2 + 3s + 0.2}{s} \left( \frac{s + 5}{2} \right)^2 \right),
\]  

(11)
giving a controller of order 13. Introducing an elliptic filter to attenuate the peaks in the open loop did not give the same results as for the \(H_{\infty}^m(q_m)\)-controller. Instead of improving the loop gain, the elliptic filter made it worse.

4.2.3. \(H_{\infty}^\infty(q_m, \hat{P})\): For this controller, four different weights have to be chosen, recall (4). The weight \(M_S\) should limit the peak of \(S\) and is therefore chosen to be a constant\(^3\). The peak of \(G_{wu}\) is also important to reduce in order to keep the control signal bounded, especially for high frequencies. A constant value of \(W_u\) is therefore also chosen. In the spirit of try simplest thing first, the weight \(W_T\) is also chosen to be a constant.

In order to attenuate the disturbances it is, as was mentioned above, necessary to have at least one integrator in the controller. Forcing \(S\) to 0 is the same as letting \(W_P\) approach \(\infty\) when \(\omega \to 0\). To get a proper inverse, a zero is also included in the weight. Since a pure integrator is not used, the slope of the weight has to be higher than 20 dB per decade frequency, in order to force \(S\) to be low enough. This was accomplished by taking the weight to the power of 3 (2 was not enough). The numerical values of the weights are chosen as

\[
W_u = 10^{-50/20}, \quad W_T = 10^{-10/20}, \quad  
M_S = 10^{-10/20}, \quad W_P = \left( \frac{s + 100}{s + 0.1} \right)^3.
\]  

(12a, 12b)

The constant weights in the form \(10^{-\alpha/20}\) can be interpreted as a maximum value, for the corresponding transfer function, of \(\alpha\) dB. The resulting controller is of order 10.

4.2.4. \(H_{\infty}^m(q_m, \hat{P})\): Like for the controller \(H_{\infty}^m(q_m, \hat{P})\), designing the weights for the mixed-\(H_{\infty}\) method becomes somewhat more involved with two measurements and one control signal. The aim is to attenuate the disturbances influence on the end-effector position. A variant is to find a rough estimate of the end-effector position and then choosing the weights from that. A straightforward estimate of \(P\) using \(\hat{P}\) is

\[
\hat{P} = \frac{1}{s} \hat{P}.
\]  

(13)

Due to low frequency drift and bias in an accelerometer, this estimate is only useful for high frequencies. A high pass filter is therefore used according to

\[
\hat{P}_{\text{high}} = c_2 \frac{s^2}{(p + s)^2} \frac{1}{s^2} \hat{P} = c_2 \frac{1}{p + s} \hat{P}
\]  

(14)

where \(c_2\) and \(p\) are constants to choose. Another straightforward estimate of \(P\) is to use the motor angle \(q_m\) according to \(\hat{P} = l q_m\), where \(l\) is the length of the arm. Compared to the estimated position using the acceleration, this new estimate is only valid for low frequencies. Using a low pass filter gives an estimate for low frequencies. It is important that the two estimates do not overlap each other, hence the low pass filter is chosen as the complementarity to the previous used high pass filter. The low frequency estimate is now given by

\[
\hat{P}_{\text{low}} = c_1 \left( 1 - \frac{s^2}{(p + s)^2} \right) l q_m = c_1 \frac{2s + p}{(p + s)^2} pl q_m.
\]  

(15)

where \(c_1\) is a design variable. The final estimate of \(P\) is the sum of the two estimates above, hence

\[
\hat{P} = \frac{c_1 (2s + p)}{(p + s)^2} \hat{P}_{\text{low}} \frac{1}{p + s} \hat{P}_{\text{high}}.
\]  

(16)

Using the weights

\[
M_S = M_{\hat{S}} W, \quad W_P = \hat{W}_P W, \quad W_T = \hat{W}_T W,
\]  

(17)

where \(M_{\hat{S}}, \hat{W}_P, \) and \(\hat{W}_T\) can be designed in a similar way as in Section 4.2.3, makes it possible to use more than one output together with one input. The last weight \(W_u\) can be chosen similar as in Section 4.2.3. The numerical values of the weights are

\[
W_u = 10^{-40/20}, \quad W = \left( \frac{30s + 75}{(s + 5)^2} \right) \left( \frac{0.1}{(s + 5)^2} \right),
\]  

(18a)

\[
\hat{M}_S = 10^{-2/20}, \quad \hat{W}_P = \left( \frac{s + 80}{s + 0.15} \right)^3.
\]  

(18b)
magnitude of the five controllers are evaluated using a simulation model. The simulation model consists of the flexible joint model described in Section 3, a measurement system, and a controller. The robot joint model is implemented in continuous time whereas the controllers operate in discrete time. The continuous-time controllers developed in Section 4, are therefore discretized using Tustin’s formula. The measurements are affected by a time delay of one sample as well as zero mean normal distributed measurement noise. The sample time is $T_s = 0.5$ ms.

The system is excited by a disturbance signal $w$ containing steps and chirp signals on both the motor and end-effector. The performance is evaluated using a performance index, which is a weighted sum of peak-to-peak errors and settling times in the simulated end-effector position and the maximum torque and the torque noise in the simulated motor torque. The reader is referred to Moberg et al. [2009] for complete details about the disturbance signals and the performance index.

Figure 5 shows how the motor torque differs between the five controllers. In the upper diagram it can be see that $H_{\infty}^w(q_m, \ddot{P})$ gives higher torques than the PID and the $H_{\infty}^m(q_m)$ controllers. The PID gives higher torque noise during steady state due to the gain of the controller for high frequencies, recall Figure 4. In the lower diagram in Figure 5 it is shown that the controllers $H_{\infty}^q(q_m, \ddot{P})$ and $H_{\infty}^m(q_m, \ddot{P})$ give similar torque signals, and lower compared to the PID controller. A low torque signal is preferred to reduce the energy consumption and to decrease the wear in the motor and gear.

The end-effector position is presented in Figure 6. In the top graph it is seen that $H_{\infty}^w(q_m)$ gives, compared to the PID, higher oscillations during the time intervals 10-15 s and 37-42 s, which corresponds to a chirp disturbance at the end-effector. For step disturbances and chirp disturbances on the motor (time intervals 16-21 s and 43-58 s) $H_{\infty}^w(q_m)$ and the PID are more similar. The controller $H_{\infty}^m(q_m)$ is better than the other two controllers in the simulation. The bottom graph shows that $H_{\infty}^m(q_m, \ddot{P})$ can handle the chirp disturbances on the motor (time intervals 16-21 s and 43-58 s) and step disturbances very good. For a chirp disturbance on the end-effector, the two $H_{\infty}$ controllers give similar results. For step disturbances, the controller $H_{\infty}^w(q_m, \ddot{P})$ gives lower peaks than the PID controller, however the settling time is longer. The steady state error of approximately 2 mm after 25 s is a result of a constant torque disturbance on the end-effector. The size of the error will depend on the size of the disturbance and the stiffness of the joint. The motor position, which is measured, is controlled to zero for all five controllers.

The performance index for the five controllers is presented in Table 1. It shows, as discussed above, that $H_{\infty}^w(q_m)$ and the PID controller behave similar and that $H_{\infty}^q(q_m, \ddot{P})$ and $H_{\infty}^m(q_m)$ give similar behaviour. The $H_{\infty}^m(q_m, \ddot{P})$-controller gives the best result.

6. LOW ORDER SYNTHESIS

For implementation of the controller in the robot control system it is important to have a low order controller. A
controller in state space form requires $O(n_x^3)$ operations to calculate the control signal, where $n_x$ is the dimension of the state vector in the controller.

The low order controllers are here obtained using the MATLAB-function hinfstruct, which is included in Robust Control Toolbox and it is based on techniques from Apkarian and Noll [2006].

To find controllers with low orders using hinfstruct requires a model description, including the weights, in the form of (1). This structure is already used for the controllers $H_{\infty}(q_m)$ and $H_{\infty}(q_m, \hat{P})$, hence it is straightforward to synthesis low order controllers using the weights presented in Sections 4.2.3 and 4.2.4. For the loop shaping design method, the structure in Figure 1(b) can be rewritten in the form of (1) including the weights $W_1(s)$ and $W_2(s)$, explained in e.g. Zhou et al. [1996]. Using the rewritten structure, the low order controllers based on $H_{\infty}(q_m)$ and $H_{\infty}(q_m, \hat{P})$ can be obtained using the weights from Sections 4.2.1 and 4.2.2.

Table 2 shows the lowest order for the respective controllers, that can be achieved without changing the closed-loop performance too much. The table also shows the performance index obtained when the controllers are used in the simulation environment. The orders can be reduced by a factor of two to three but the performance of the reduced order controllers is worse than the full order controllers. Since the controller based on loop shaping with only $q_m$ as measurement has the same performance for the full order controller as the PID controller, the low order controller gives a worse performance than the PID controller. The other full order controllers are much better than the PID controller and afford to get a reduced performance for the low order controllers without getting worse than the PID controller.

Finally, note that the controllers only represents local minima solutions, hence rerunning hinfstruct with other initial values can give a better, or worse, controller. To handle this, several initial values have been used in hinfstruct.

7. CONCLUSIONS AND FUTURE WORK

Four different $H_{\infty}$ controllers for a flexible joint of an industrial manipulator are designed using mixed-$H_{\infty}$ controller design and the loop shaping method. The model, on which the controllers are based, is a four-mass model.
As input, the controllers use either the motor angle only or both the motor angle and the acceleration of the end-effector. Tuning of the controllers requires understanding of both the synthesis method and how the system behaves. For example, the measurements for the mixed-$H_\infty$ controller are first pre-filtered to give an estimate of the tool position. The weighting functions for the resulting SISO system, from input torque to the estimated tool position, are then chosen similar to the case where only the motor position is used.

The controllers are compared to a PID controller and it is shown that if only the motor angle is measured it is much better to use the mixed-$H_\infty$ design method compared to loop shaping. If instead the end-effector acceleration is added then the performance is improved significantly for both methods. The steady state error for the end-effector position is unaffected since the accelerometer does not provide low frequency measurements. Using a low order controller synthesis method, it is possible to reduce the order of the controllers by a factor of two to three but at the same time a decrease in the performance index of 10–30% can be observed.

Investigation of robustness for stability with respect to model errors is one of several future directions of research. The mixed-$H_\infty$ method has an advantage compared to the loop shaping method since a model of the error is possible to incorporate in the augmented plant $P(s)$.

Another continuation is to investigate the improvement for other types of sensors. One possibility is to have an encoder measuring the position directly after the gearbox, i.e., $q_m$. This will improve the stiffness of the system, although it will not eliminate the stationary error for the end-effector position. The ultimate solution is to measure the end-effector position, but for practical reasons this is in general not possible, instead the end-effector position can be estimated, as described in Axelsson [2012], Axelsson et al. [2012], and used in the feedback loop.

Extending the system to several joints giving a non-linear model, which has to be linearised in several points, is also a future problem to investigate. A controller, using the results from this paper, is designed in each point and for example gain scheduling can be used when the robot moves between different points.

### Table 2. Lowest order of the controllers obtained using hinfsynstruct, with the order before reduction in brackets. The corresponding performance index from simulations is also shown.

<table>
<thead>
<tr>
<th>Order</th>
<th>$H_{\infty}( q_m )$</th>
<th>$H_{\infty}( q_m, P )$</th>
<th>$H_{\infty}( q_m )$</th>
<th>$H_{\infty}( q_m, P )$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perf. ind.</td>
<td>60.8</td>
<td>49.4</td>
<td>48.3</td>
<td>40.8</td>
</tr>
</tbody>
</table>

REFERENCES


