Velocity sensorless control of uncertain load using RKF tuned with an evolutionary algorithm and mu-analysis
Sébastien Carrière, Stéphane Caux, Maurice Fadel, Francesco Alonge

To cite this version:
Sébastien Carrière, Stéphane Caux, Maurice Fadel, Francesco Alonge. Velocity sensorless control of uncertain load using RKF tuned with an evolutionary algorithm and mu-analysis. 4th IFAC Symposium on System, Structure and Control (IFAC SSC) 2010, Sep 2010, Ancône, Italy. pp.11-16, 10.3182/20100915-3-IT-2017.00019 : hal-03542119

HAL Id: hal-03542119
https://ut3-toulouseinp.hal.science/hal-03542119
Submitted on 25 Jan 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Velocity sensorless control of uncertain load using RKF tuned with an evolutionary algorithm and mu-analysis

Sébastien Carrière* Stéphane Caux* Maurice Fadel*
Francesco Alonge**

* Université de Toulouse; INP, UPS; LAPLACE (Laboratoire Plasma et Conversion d’Energie); ENSEEIHT, 2 rue Charles Camichel, BP 7122, F-31071 Toulouse cedex 7, France. (e-mail: stephane.caux@laplace.univ-tlse.fr).
** Università di Palermo; DIAS; viale delle scienze, I-90128 PALERMO ITALY., (e-mail: alonge@unipa.it)

Abstract: In case of a velocity control scheme for a load directly driven by an actuator, large variations of its parameters are problematic due to possible instability and large variations of the final performances. This performances are then decreasing if a sensorless control is implemented due to cost, reliability or application constraints. This paper proposes solutions to quickly and accurately tune an observer with a lower computer time consumption and lower conception time. A previous calculated state feedback is used as base for a Kalman filter with special noise matrices. An evolutionary algorithm optimizes the observers degrees of freedom all over the variations. The mu-analysis theory helps to cancel known unstable set of parameters before running iterations in the optimization algorithm. Experiments show that the stability and the performance are effectively maintained.

Keywords: Motion Control, Robustness, Optimization.

1. INTRODUCTION

Numerous systems working in hard surroundings as a crusher, a rolling mill or a driller have their reliability dramatically decreased by sensors. A system with a long shaft does not hold sensor on load side due to a cost or a complexity requirement so an observer should be considered. Due to large parameter variations of the axis’ stiffness, the inertia or the friction depending on the load, a robust specification should be introduced. Modeling these systems is possible by a well-known two-mass system. Due to the stiffness, the load and the motor inertias see different speeds. Therefore, removing the load speed sensor can lead to oscillations or to an unstable system if it is looped with the wrong motor speed or by an information furnished by a non accurate observer. However, implementing a reliable control algorithm should not only enable the system to remain stable but it should also ensures a desired performance.

Several methods exist to solve this problem : remaining the motor speed loop and slowing down the system response or implementing a robust observer is possible. Previous works in Carrière et al. (2008), or a robust synthesis exposed in Peter et al. (2001) prove that conserving the system’s performance under variations is feasible with a simple controller such as an optimized state feedback. Oscillations due to parameter variations should be avoided by a robust controller synthesis or adding an observer not only to obtain the states used in the controller, but to contribute with the command to the global closed loop performance.

Different structures allow a robust state reconstruction: An adaptive structure is used in Feiler et al. (2003), an extended Kalman filter in Kalman and Bucy (1961) or a sliding mode control in Fakham et al. (2008) provide a correct control but are complex and time consuming mainly in the conception phase.

Nevertheless, a Kalman filtering method appears as a good solution to this class of problem because its stochastic approach allows to deal optimally with uncertainties and to pre-calculate gains off-line. Furthermore, this method allows to check one objective, the low on-line computation time but the Kalman parameters tuning is still problematic.

First of all, the system is modeled by separating the mechanical and electrical parts in section 2. The stiffness and the inertia variations taken into account are bounded and it is shown how that they modify the system behavior. The following section 3 describes the control law implemented to reach as easily as possible the requirements. As a consequence, the heart of the work, the Kalman filter optimization is described in 4. Firstly, the Kalman filter is presented in its classical form, secondly a variable-model-based form is detailed, transforming the variations acting on dynamic’s system matrices into additional state noises. Consequently, a stability study is performed with the µ-analysis theory. To have optimized results, an evolutionary algorithm is used to find the coefficients of the observer (6). Finally, the complete sensorless control structure is
hence compared under the variations with experimental measurements in section 7.

2. SYSTEM MODELING

2.1 Actuator Control Structure

A Permanent Magnet Synchronous Motor (PMSM) is the most used drive in machine tool servos and in modern speed control applications due to its desirable capabilities. Added to the previous advantages, this 3-phase motor is similar to a DC motor for AC power supply. The 2.5 kW motor is powered by a three-phase reversible inverter switching at 15 kHz. The influence of inverter in such control is commonly neglected if an accurate current control (described in section 3) and a torque control are effective to contribute to the accurate velocity on the load side. Keep in mind for the load control, this actuator imposes current and torque limitation in bandwidth and magnitude.

2.2 Mechanical specifications

The mechanical load driven without any reducer is here not only composed with the inertia and the friction but also includes a flexible transmission joint. Indeed, a spring can be inserted between the motor and the load as shown in Fig. 1. So, two first order systems (inertia and viscous friction) respectively subscripted by \( l \) and \( m \) for the load and the motor elements are connected through the spring. The load inertia and the spring stiffness can vary respectively by a factor of 2.5 and 26.6 (see Table 1). A load disturbance torque, created by a powder brake, can be applied. Despite the motor speed measurement, a varying stiffness imposes to recalculate the other state variables to not have an oscillating or unstable load control. To implement this item, the system has to be modeled by state space equation as shown in 1 where : \( T_m \) is the applied motor torque, \( T_l \) is the load torque, \( J_m \), \( f_m \) and \( J_l \), \( f_l \) are the inertia and the friction of respectively the motor and the load side and \( K_T \) represents the stiffness of the axis and joint.

The \( C \) matrix provides the motor speed as output until the load speed is controlled because this is the only sensor used in such application. Moreover, three state variables belong to the system, the motor and the load speed (\( \omega_m \) and \( \omega_l \)) and the position difference (\( \Delta \theta = \theta_l - \theta_m \)).

\[
X = \begin{bmatrix} \omega_m & \omega_l & \Delta \theta & T_l \end{bmatrix}^T
\]

\[
\dot{X} = \begin{bmatrix}
-\frac{f_m}{J_m} & 0 & -\frac{K_T}{J_m} & 0 \\
0 & -\frac{f_l}{J_l} & \frac{K_T}{J_l} & -\frac{1}{J_l} \\
1 & -1 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix} X + \begin{bmatrix}
\frac{1}{J_m} \\
0 \\
0 \\
0
\end{bmatrix} T_m
\]

(1)

\[
y = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} X
\]

The stiffness and inertia variation are created by replacing the spring and taking down the wheels.

![Fig. 1. The test bench](image1)

![Fig. 2. Complete control scheme](image2)

Table 1. Test bench parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Motor</th>
<th>Load</th>
</tr>
</thead>
<tbody>
<tr>
<td>Torque (Nm)</td>
<td>16</td>
<td>7 (brake)</td>
</tr>
<tr>
<td>Inertia (kg\cdot m^2)</td>
<td>2.6 \cdot 10^{-4}</td>
<td>19.2; 50.9 \cdot 10^{-4}</td>
</tr>
<tr>
<td>Friction (Nm\cdot s\cdot rd^{-1})</td>
<td>2.7 \cdot 10^{-4}</td>
<td>5.4 \cdot 10^{-4}</td>
</tr>
<tr>
<td>Stiffness (Nm\cdot rd^{-1})</td>
<td>[75; 2000] (springs)</td>
<td></td>
</tr>
</tbody>
</table>

The following sections describe the control laws implemented on the system and their performances including stiffness or inertia variations.

3. SENSORLESS CONTROL SCHEME

Two loops are needed to control this system as shown in Fig. 2. The current controllers \( (d \) and \( q \) are the Park frame axis) are classical PI calculated to have a 100 Hz bandwidth and no overshoot. The current controllers are effective and feed the Pulse Width Modulation (PWM) inverter and impose the duty cycles \( \alpha \) to each phase voltage alimentation and provide a dynamic and accurate torque control. Now, with the torque known, the speed load controller has to maintain the motor speed dynamics as constant as possible, in spite of mechanical parameter variations. Previous works Carrière et al. (2008) focused on a constant gain controller designed to minimize these effects using linear quadratic optimization coupled to a dominant pole placement including an integrator on the speed error, this is now completed for large variation control with the observer gains synthesis.

To reach such system requirements, an observer is implemented the motor speed is provided by the system and is given as input to the observer. All the states are rebuilt and the load torque is also included in the observer for convergence. The load torque information can be helpful to accelerate the response in case of a disturbed torque by...
directly reacting without any integration on control input
(torque reference \(T_{m\text{ref}}\) as explained in Choi et al. (2005). So, observed torque is directly added to the computed control input as shown in the complete control scheme Fig. 2.

The load parameter variations imply a modification of the matrix \(A\) and removes the possibility of using a Luenberger-type observer. However, a Kalman filter first exposed in Stevens and Lewis (2003) is a quite interesting method by considering the parameter variations as an additional state variable noise. Note that goal of this studied observer is not only to rebuild the states with a sufficient accuracy but also to do it even if variations occur in the system’s set of parameters. The observer has to provide values requested by the controller to impose a defined response characteristic whatever the mechanical parameters. The two following sections present the method to compute the Kalman filter gain and an off-line optimization algorithm for tuning these gains helped with a theoretical mu-analysis.

4. THE CONTINUOUS STANDARD KALMAN FILTER (KS)

The continuous Kalman filter is an optimal observer in noise rejection point of view. From the motor speed \(\omega_m\) and the motor torque \(T_m\), the state should be rebuilt with the dynamic model established 2.

\[
\begin{align*}
\dot{X}(t) &= AX(t) + BT_m(t) + w(t) \\
y &= CX(t) + v(t)
\end{align*}
\]

(2)

\(X\) is the state vector, \(T_m\) is the input torque, \(\omega_m\) is the output and \(w\) and \(v\) are respectively the state and the measurement output noises. However, to ensure the filter’s convergence, some critical assumption should be effective, mainly, the noises have to be white Gaussian noises with a null mean value.

The minimization of the quadratic error between the estimation and the real values provides the algebraic Riccati equation which allows to calculate the instantaneous Kalman coefficients. It can be proven that this equation tends to a steady state. Consequently, only the steady state coefficients should be implemented. Finally, to find the coefficients, 3 has to be solved using \textit{are} algorithm under \textit{matlab}(C)

\[
P(t)A + ATP(t) + P(t)CTR^{-1}CP(t) + Q = 0
\]

(3)

Unfortunately, the matrices \(Q\) and \(R\) change widely the output estimation behavior and conduce the system to reacts well, to be roughly stable or to be unstable. The first method is to choose the matrices with white uncorrelated noises with a null mean, so the matrices are diagonals.

The parameter variations imply that matrices does not match with classical noises notion. They have to describe a state variable shape modification which is not gradable as a white gaussian noise.

4.1 Model variation tuning (KR)

The second method is inspired from Szabat and Orlowska-Kowalska (2008) where the system is modified following 4. A matrix \(B_r\) is premultiplied with the state noises to give them a specific direction.

\[
\begin{align*}
\dot{X}(t) &= AX(t) + BT_m(t) + B_rw(t) \\
y &= CX(t) + v(t)
\end{align*}
\]

In the first method, the parametric variations are minimized by scaling each state noise covariance. On this system, these variations modify directly the dynamic matrix \(A\). As a consequence, the \(B_r\) matrix represents the variation of this dynamic matrix as written in 5.

\[
B_r = A_{\text{min}} - A_{\text{max}} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & x \\ \end{bmatrix}
\]

(5)

However, the systems are always subject to exogenous perturbations, the load torque \(T_l\) allows the observer to be stable under this kind of perturbations. Therefore, a diagonal matrix is added to the dynamic matrix variation with only exogenous state variable terms which are non null.

This matrix normalizes the state noises \(w(t)\) and modify the Algebraic Riccati Equation, replacing covariance \(Q\) by the form 6. The \(\gamma\) coefficient allows to rate the \(B_r\) matrix compare to \(R\) matrix.

\[
Q = \frac{B_r \ast B_r^T}{\gamma}
\]

(6)

Here, only 3 coefficients have to be tuned \((R, x\) and \(\gamma\)) compare to the five of the standard structure. The high number of parameters avoids a human tuning to optimize the filter behavior, then an evolutionary algorithm, presented in section 6, is used to tune optimally the matrices coefficients. This optimization is based on exploring the search-space and a criterion evaluation on each element tested. An unstable set of parameters may first be avoided to not have a long running simulation time and help the quick convergence.

5. \(\mu\)-ANALYSIS

5.1 Theory

In order to prove mathematically stability otherwise by simulating a long time or having a fastidious scheme, \(\mu\)-analysis technic proved and explained in Skogestad and Postlethwaite (2005) allows to check the system stability under variations. This analysis is an extension of the small gain theorem. Separately, the system modeled by LFT \(M(s)\) shown in Fig. 3 and the variation matrix are stable. The variation matrix is normalized, thus its norm is inferior to 1. Then, the only component which is able to destabilize the system is the feedback introduced by the loop. Consequently, only the poles and the upper singular value of the matrix \(M_{wz}(s)\) is important to define stability.
Fig. 3. System under LFT

\[ M_{wz}(s) \] is the submatrix corresponding to the input \( w \) et output \( z \) of the \( M(s) \) matrix.

In the small gain theorem, while the upper system \( M_{zw}(s) \) singular value is less than 1, the system is stable for all uncertainties matrix \( \Delta(s) \). Nevertheless, a lot of \( \Delta(s) \) matrix cannot correspond to with the system definition and decrease the space of stability. The uncertainties are classified in two categories, see Skogestad and Postlethwaite (2005):

- Structured: These uncertainties can be totally modeled (gain and phase) with the system state variables and lead to a real scalar or a group of diagonal scalars in \( \Delta(s) \) matrix. By definition, on the described system, this type represents the parametric variations.
- Unstructured: By adding transfer function, only the gain is modeled by using the system state variables for these uncertainties then the phase is given by a complex value inside the \( \Delta(s) \) matrix. This type allows to model all the neglected dynamics, delays, and noises.

Resulting from the previous assumptions, the \( \Delta \) matrix is a bloc diagonal where \( \alpha_i(s) \) represents the \( q_{th} \) unstructured uncertainties and \( \delta_i \), the \( t_{th} \) structured uncertainties. The \( \Delta(s) \) matrix is a bloc diagonal matrix because each uncertainties can only have an effect on the part of the system where they are located.

\[
\Delta(s) = \text{diag}\{ \alpha_1(s), ..., \alpha_q(s), \delta_jI_j, ..., \delta_kI_k \} \\
\Delta(s) \in CHC; \delta \in \mathbb{R}
\]

Now, the space of all the admissible \( \Delta(s) \) matrices is defined, for all frequency and all the \( \Delta(s) \) respecting 7, the value of \( \mu(\Delta(s)) \) described in 8 is estimated by the algorithm programmed inside the robust toolbox of Matlab software. By solving 8, the algorithm looks for the value of \( k_m \) with poles on imaginary axis. Consequently, if \( \mu(\Delta(s)) < 1, \forall \omega, k_m > 1 \) and shifts the poles toward the right part of the complex plane, it implies that the system is stable for the uncertainties taken into account.

\[
\mu(\Delta(M(s))) = (\min\{k_m, \text{det}(I - k_m \Delta(s)M(s)) = 0\})^{-1}
\]

The stability can now be proved before simulation.

5.2 Specifications for the control structure

Finally considering this system, the uncertainties are:

- the load inertia variation,
- the stiffness variation,
- the delay of measurement,
- the neglected dynamics of current loop,
- the measurement noise.

Each uncertainties is modeled by an LFT considering multiplicative effect. Scalars or transfer functions are involved depending on the uncertainties. The \( \mu \)-analysis is working with symmetric variation space thereof the system nominal point is chosen at the center of the variations to have the smallest space.

The 2 first are structured and leads each to one scalar \( \delta_i \) representing the normalized variation from the nominal point. The 3 last are unstructured and are gathered inside a \( 2 \times 3 \) complex matrix \( \Delta_u(s) \) to be treated as described into Skogestad and Postlethwaite (2005). This matrix creates all the phase variations possible due to the unstructured uncertainties. As a consequence, the matrix \( \Delta(s) \) has the form described in 9. \( M(s) \) is split into four submatrix 10:

- \( M_{zw}(s) \) is the feedback created by modeling with LFT. It gives the system stability under variation. Its dimensions are \( 5 \times 4 \) to be connected with the \( \Delta(s) \) matrix.
- \( M_{zu} \) and \( M_{yw} \) create the link between the variation and the nominal system.
- \( A_c \) is the dynamic matrix of the tested closed loop system.

\[
\Delta(s) = \begin{pmatrix}
\delta_1 & 0 & 0 \\
0 & \delta_2 & 0 \\
0 & 0 & \Delta_u(s)
\end{pmatrix}
\]

\[
M(s) = \begin{pmatrix}
M_{zw}(s) & M_{zu}(s) \\
M_{yw}(s) & A_c
\end{pmatrix}
\]

The following evolutionary algorithm simulates and computes the time response of the controlled system. But inside the allowed space of observer’s parameters, some combinations have very slow instability which appear after the end of the time response simulation or due to nonlinear effect as time delays, noise or neglected dynamics. The \( \mu \)-analysis integrated inside the criterion calculation allows to check such effect on stability prior to simulation and save time during algorithm execution. Furthermore, the neglected dynamics are also neglected inside the time simulation process (to avoid very long time simulation) and unstable system can be detected prior experimental test and allow algorithm to always give an effective solution.

6. Evolutionary Algorithm

Given the human abilities, the both presented structures require to deal with a great number of variables. To overcome this difficulty and to scan faster a large space of possible solutions, an evolutionary algorithm has been implemented. Extract from Bäck (1996), this algorithm searches the best pattern by evolving item parameters as made in nature. It means that only a combination and a mutation are allowed.

The principle of this algorithm is presented in Fig. 4. Firstly, classifying items from the most competitive to the less interesting one is a priority, then an evaluation criterion is calculated (detailed hereafter 11). A close regards on criterion reveals two summed items. The first item is
the integral quadratic error between the tested response and the reference, this error is relevant when the difference is high thus for the transient time. Nevertheless, the algorithm takes high coefficients to minimize the transient deviation and this increases noise effects which creates small speed variations avoided by speed regulation. Consequently, the second item is the integral time absolute error which amplifies the error for steady state because the error is multiplied by time and avoid torque ripple at steady state.

\[
f = \sum_{i=0}^{n} \left( \int_{0}^{T_{end}} (\omega_f(t, \Theta_{ref}) - \omega_i(t, \Theta_i))^2 dt + 2 \int_{0}^{T_{end}} abs((T_m(t, \Theta_{ref}) - T_m(t, \Theta_i)) \cdot t \cdot dt) \right)
\]  

In fact, this optimization ensures that the system has the minimal output deviation for all the system variations (inertia or stiffness) compare to a reference behavior which is system with sensor and with this set of parameter: \( \Theta_{ref} = (J_{imax}, f_{imax}, K_{T_{max}}) \).

First, the system has to be stable, thus, the \( \mu \)-analysis of the closed loop system is computed. If system is unstable, a very high criterion is given to this item and evaluation is over, otherwise, a second step is executed providing a numerical criterion value by simulating the system. So, a set of parameter variations \( \Theta \) is defined. \( \Theta \) represents all the parametric variation bounds to be taken into account. All \( \Theta \) time responses of the system driven with the chosen observer are computed. After all experiments, the criterion 11 is computed to evaluate the performance of the structure. The criterion is computed for each variations of the set \( \Theta \) and all effects are summed.

Secondly, each item has a vector of parameters : the three coefficients listed in section 4 representing the observers : the output of algorithm, plus one standard deviation for observer’s parameter to achieve the mutation.

Thirdly, the optimization needs the following steps to be effective :

1. The parent’s creation allows the algorithm to start.
2. The children’s birth is operated with two calculations, the first is a recombination. Two parents are randomly chosen and each parameter is modified by a weighted mean of both parents parameters. The weighting is randomly defined by an uniform distribution.
3. The second operation is a mutation. The parameters used as standard deviation mutate following a decreasing exponential law.
4. Finally, the algorithm is ended when the criterion value is quite similar between two consecutive step (0.5%) after a minimal number of steps (10).

All methods and constants to be defined are deeply explained into the standard reference found in Bäck (1996). Tuning the filter’s gains is now possible in two or three hours with a Pentium 4 2.4 Ghz. The last section shows and compares the experimental results of the proposed filters.

7. THE TEST BENCH

The experimentation is controlled via a DSpace® platform in a friendly environment. The programmes are made with MATLAB/Simulink®. Two control loops have to be programmed from the scheme drawn in figure 2. The loops have quite different dynamics to allow a simpler tuning procedure than a coupled system. The kalman filter computes all state from the noisy output of motor position derivation.

All elements confirm the necessity to deal with Robust observer synthesis as explained in previous theoretical parts. To underline this fact, some results are shown. Firstly, two tables compare the performance of each observer:

- Table. 2 writes the performance of each observer in all the variations cases
- Table. 3 compares the criterion and the number of generation needed by the algorithm to converge to the final solution

secondly, two curves shows the behavior of the system on extreme conditions, the minimal inertia and maximal stiffness in Fig. 5, the maximal inertia and minimal stiffness for Fig. 6.

However, KR4 is more dynamic in the reference transient Fig. 5 and have a slightly smaller overshoot in Fig. 6. Contrarily, the KS4 observer deals better with the load disturbances, less deviation and a better recovery time on the both Fig. 5 and Fig. 6 curves. Nevertheless, the difference are thin and the both observer can be considered as as performing for the two-mass system. All this minor differences are underlined by figure written in Table. 2. On the contrary, Table. 3 lights out something very interesting for an industrial. By reducing the number of parameter for a given structure, the criterion is not too much bad but,
8. CONCLUSION

A strategy has been developed to guarantee stability of the system. The observer gains are tuned with an evolutionary algorithm comparing the simulation results and making a $\mu$-analysis of the structure for stability purpose under parameter variations. The KR4 form has a slighter better results. But the main difference is on the parameter’s number to be tuned. The robust KR4 form has only 3 dof compared to 5 dof for standard KS4 method. In fact, the noise state matrix is defined by taking into account the dynamic matrix variation and allows a direct tuning. It is possible to compute gains off-line, that makes them easy to implement with a lighter execution time as usually required. The experimental results prove that implementation is feasible with good performance and allow to finally obtain “robust” optimized observer on a direct drive system with load parameter variations.

REFERENCES


Table 2. Performance table

<table>
<thead>
<tr>
<th>Overshoot (%)</th>
<th>Time response (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_{T_{1}} J_{l_{1}}$</td>
<td>KR4</td>
</tr>
<tr>
<td>0.7</td>
<td>0.7</td>
</tr>
<tr>
<td>$K_{T_{0}} J_{l_{0}}$</td>
<td>14</td>
</tr>
<tr>
<td>$K_{T_{0}} J_{l_{0}}$</td>
<td>0</td>
</tr>
<tr>
<td>$K_{T_{1}} J_{l_{1}}$</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 3. Optimization results

<table>
<thead>
<tr>
<th>parametric variation</th>
<th>Criterion</th>
<th>Number of generation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta K_{T}$ and $\Delta J_{l}$</td>
<td>13.39</td>
<td>13.49</td>
</tr>
<tr>
<td>$\Delta K_{T}$</td>
<td>9.75</td>
<td>9.77</td>
</tr>
<tr>
<td>$\Delta J_{l}$</td>
<td>10.68</td>
<td>10.53</td>
</tr>
</tbody>
</table>

Table 2. Performance table

<table>
<thead>
<tr>
<th>Load speed deviation (%)</th>
<th>Recovery time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_{T_{1}} J_{l_{1}}$</td>
<td>KR4</td>
</tr>
<tr>
<td>8.3</td>
<td>6.7</td>
</tr>
<tr>
<td>$K_{T_{0}} J_{l_{0}}$</td>
<td>7.1</td>
</tr>
<tr>
<td>$K_{T_{0}} J_{l_{0}}$</td>
<td>13</td>
</tr>
<tr>
<td>$K_{T_{1}} J_{l_{1}}$</td>
<td>14.2</td>
</tr>
</tbody>
</table>

Fig. 5. System behaviour with $K_{T_{1}}$ and $J_{l_{1}}$.

Fig. 6. System behaviour with $K_{T_{0}}$ and $J_{l_{0}}$.

the time synthesis is greatly improved, much of time by 40%.

In this case, the variations are taken into account so the instability is rejected.