Robust Control Toolbox

Design robust controllers for uncertain plants

Robust Control Toolbox™ provides functions, algorithms, and blocks for analyzing and tuning control systems for performance and robustness. You can create uncertain models by combining nominal dynamics with uncertain elements, such as uncertain parameters or unmodeled dynamics. You can analyze the impact of plant model uncertainty on control system performance and identify worst-case combinations of uncertain elements. H-infinity and mu-synthesis techniques let you design controllers that maximize robust stability and performance.

The toolbox automatically tunes both SISO and MIMO controllers. These can include decentralized, fixed-structure controllers with multiple tunable blocks spanning multiple feedback loops. The toolbox lets you tune one controller against a set of plant models. You can also tune gain-scheduled controllers. You can specify multiple tuning objectives, such as reference tracking, disturbance rejection, stability margins, and closed-loop pole locations.

Key Features

- Modeling of systems with uncertain parameters or neglected dynamics
- Worst-case analysis of stability margins and sensitivity to disturbances
- Automatic tuning of centralized, decentralized, and multiloop controllers
- Automatic tuning of gain-scheduled controllers
- Robustness analysis and controller tuning in Simulink®
- H-infinity and mu-synthesis algorithms
- General-purpose LMI solvers
Model of an aircraft autopilot system (top), the algorithm used to tune it (middle), and a plot of the closed-loop response to a step setpoint and a step disturbance before and after tuning (bottom). You can use Robust Control Toolbox to automatically tune complex multivariable controllers consisting of basic Simulink blocks and then evaluate the improvement in the closed-loop response.

**Modeling and Quantifying Plant Uncertainty**

With Robust Control Toolbox, you can capture not only the typical, or nominal, behavior of your plant, but also the amount of uncertainty and variability. Plant model uncertainty can result from:

- Model parameters with approximately known or varying values
- Neglected or poorly known dynamics, such as high-frequency dynamics
- Changes in operating conditions
- Linear approximations of nonlinear behaviors
- Estimation errors in a model identified from measured data
Building and Manipulating Uncertain Models

Build uncertain state-space models and analyze the robustness of feedback control systems that have uncertain elements.

Plot (top) and corresponding MATLAB® code (bottom) of the worst-case gain of a system with an uncertain parameter. Robust Control Toolbox lets you create an uncertain model by adding uncertain elements to nominal plant models and then analyze the effect of uncertainty by calculating the worst-case system performance.

Robust Control Toolbox lets you build detailed uncertain models by combining nominal dynamics with uncertain elements, such as uncertain parameters or neglected dynamics. By quantifying the level of uncertainty in each element, you can capture the overall fidelity and variability of your plant model. You can then analyze how each uncertain element affects performance and identify worst-case combinations of uncertain element values.

Performing Robustness Analysis

Using Robust Control Toolbox, you can analyze the effect of plant model uncertainty on the closed-loop stability and performance of the control system. In particular, you can determine whether your control system will perform adequately over its entire operating range, and what source of uncertainty is most likely to jeopardize performance.
Robustness of Servo Controller for DC Motor

Model uncertainty in DC motor parameters and analyze the effect of this uncertainty on motor controller performance.

You can randomize the model uncertainty to perform Monte Carlo analysis. Alternatively, you can use more direct tools based on mu-analysis and linear matrix inequality (LMI) optimization; these tools identify worst-case scenarios without exhaustive simulation.

Robust Control Toolbox provides functions to assess worst-case values for:

- Gain and phase margins, one loop at a time
- Stability margins that take loop interactions into account
- Gain between any two points in a closed-loop system
- Sensitivity to external disturbances

These functions also provide sensitivity information to help you identify the uncertain elements that contribute most to performance degradation. With this information, you can determine whether a more accurate model, tighter manufacturing tolerances, or a more accurate sensor would most improve control system robustness.

Nominal and worst-case rejection of a step disturbance (top) and Bode diagram of a sensitivity function (bottom). Robust Control Toolbox lets you analyze the effect of plant model uncertainty on closed-loop stability and control system performance.
**Synthesizing Robust Controllers**

Robust Control Toolbox lets you automatically tune centralized and decentralized MIMO control systems. The controller synthesis algorithms include H-infinity and mu-synthesis techniques, nonsmooth optimization, and LMI optimization. These algorithms are applicable to SISO and MIMO control systems. MIMO controller synthesis does not require sequential loop closure and is therefore well-suited for multiloop control systems with significant loop interaction and cross-coupling.

**Automatic Tuning of Fixed-Structure Control Systems**

Most embedded control systems have a fixed, decentralized architecture with simple tunable elements such as gains, PID controllers, or low-order filters. Such architectures are easier to understand, implement, schedule, and retune than complex centralized controllers. Robust Control Toolbox provides tools for modeling and tuning these decentralized control architectures. You can:

- Specify tunable elements such as gains, PID controllers, fixed-order transfer functions, and fixed-order state-space models
- Combine tunable elements with ordinary linear time-invariant (LTI) models to create a tunable model of your control architecture
- Specify and visualize tuning requirements such as tracking performance, disturbance rejection, noise amplification, closed-loop pole locations, and stability margins
- Automatically tune the controller parameters to satisfy the must-have requirements (design constraints) and to best meet the remaining requirements (objectives)
- Validate controller performance in the time and frequency domains

In addition to tuning a fixed-structure controller for one plant model, Robust Control Toolbox lets you automatically tune a controller against a set of plant models. You can use this functionality to design a controller that will be robust to changes in plant dynamics due to plant parameter variations, changes in operating conditions, and sensor or actuator failures.

**H-Infinity and Mu-Synthesis Techniques**

Robust Control Toolbox provides several algorithms for synthesizing robust MIMO controllers directly from frequency-domain specifications of the closed-loop responses. For example, you can limit the peak gain of a sensitivity function to improve stability and reduce overshoot, or limit the gain from input disturbance to measured output to improve disturbance rejection. Using mu-synthesis algorithms, you can optimize controller performance in the presence of model uncertainty, ensuring effective performance under all realistic scenarios.
Robust Control of an Active Suspension

Approximate high-order plant models with simpler, lower-order models. Design a robust controller for an active suspension system using H-infinity and mu-synthesis methods.

Tuning Gain-Scheduled Controllers

Gain scheduling is a linear technique for controlling nonlinear or time-varying plants. It involves computing linear approximations of the plant at various operating conditions, tuning controller gains at the operating condition, and scheduling controller gains as the plant changes operating conditions. Robust Control Toolbox provides tools for automatically computing gain schedules for fixed-structure control systems. You can:

- Automatically trim and linearize Simulink models at multiple operating conditions (using Simulink Control Design)
- Parameterize controller gain surfaces as functions of scheduling variables
- Construct a closed-loop model representing the system throughout its operating range
- Specify tuning requirements such as tracking and disturbance rejection
- Automatically tune gain surface coefficients to satisfy tuning requirements at all operating conditions
- Update parameters of Simulink blocks implementing the controller with tuned gain values

Smooth surfaces (bottom) for scheduling four controller gains in a Simulink model of an autopilot (top) as functions of velocity (V) and angle of attack (alpha). These four surfaces were automatically tuned using a 9x5 grid of operating points.

Robust Control Toolbox eliminates the need to tune one controller at a time for each operating condition and instead provides globally tuned gain surfaces that generate smooth transitions from one operating condition to another.
Tuning of Gain-Scheduled Three-Loop Autopilot
Generate smooth gain schedules for a three-loop autopilot.

Analyzing and Tuning Controllers in Simulink
Robust Control Toolbox provides tools for performing robustness analysis and for tuning controllers modeled in Simulink.

Uncertainty Modeling and Robustness Analysis
The toolbox lets you model and analyze uncertainty in Simulink models. You can:

- Introduce uncertainty into a Simulink model by using an Uncertain State Space block or by specifying block linearization for any Simulink block
- Linearize a Simulink model to create an uncertain system that represents the whole Simulink model
- Analyze the resulting uncertain system for stability and performance

Automatic Controller Tuning
Robust Control Toolbox lets you automatically tune decentralized controllers modeled in Simulink. Because Robust Control Toolbox operates on linear models, you can use Simulink Control Design to automatically compute and store a linearization of your Simulink model. Simulink Control Design automatically creates a tunable model of control architecture specified in a Simulink model. You can:

- Specify Simulink model blocks that should be tuned
- Specify tuning requirements
- Automatically tune specified blocks to satisfy the must-have requirements (design constraints) and to best meet the remaining requirements (objectives)
- Validate your design by running nonlinear simulations

Using this approach, you can automatically tune complex multivariable controllers that are modeled using Simulink blocks. For example, you can automatically tune inner-loop and outer-loop PID controllers in a multiloop control system without changing the control system architecture.
**Reducing Plant and Controller Order**

Detailed first-principles or finite-element plant models often have a large number of states. Similarly, H-infinity and mu-synthesis algorithms tend to produce high-order controllers with superfluous states. Robust Control Toolbox provides algorithms that let you reduce the order (number of states) of a plant or controller model while preserving its essential dynamics. As you extract lower-order models, which are more cost-effective to implement, you can control the approximation error.

![Model Order Reduction](image)

*Bode plots comparing the magnitude and phase of the original and reduced-order models for the rigid body motion dynamics of a multistory building.*

The model reduction algorithms are based on Hankel singular values of the system, which measure the energy of the states. By retaining high-energy states and ignoring low-energy states, the reduced model preserves the essential features of the original model. You can use the absolute or relative approximation error to select the order, and use frequency-dependent weights to focus the model reduction algorithms on specific frequency ranges.

**Simplifying Higher-Order Plant Models**

Approximate higher-order plant models with simpler, lower-order models.
<table>
<thead>
<tr>
<th>Resources</th>
<th>Online User Community</th>
<th>Training Services</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trial Software</td>
<td><a href="http://www.mathworks.com/trialrequest">www.mathworks.com/trialrequest</a></td>
<td></td>
</tr>
<tr>
<td>Sales</td>
<td><a href="http://www.mathworks.com/contactsales">www.mathworks.com/contactsales</a></td>
<td></td>
</tr>
<tr>
<td>Technical Support</td>
<td><a href="http://www.mathworks.com/support">www.mathworks.com/support</a></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© 2014 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.