Simulating the Ramsey/Cass-Koopmans Model using MATLAB and Simulink

By Sonia Bridge and Ken Deeley, MathWorks

Many economic and financial models involve systems of differential equations with no explicit analytical solutions. Solving these systems numerically to gain insight into market conditions is a key challenge for economists and other financial professionals.

The fundamental Ramsey/Cass-Koopmans (RCK) model aims to explain long-term economic growth in terms of capital accumulation and consumption growth. The core RCK model is two-dimensional, comprising two coupled ordinary differential equations for per-capita wealth (*k*) and per-capita consumption (*c*). The phase portrait of the model is shown in figure 1.

This article presents a complete workflow showing how MATLAB and Simulink can be used to create, simulate and visualize the RCK model.

Three sentences:

Simulink is a block diagram / system modelling / time-varying systems with feedback …. What is Simulink???

While visually appealing, it’s not typically used by financial professionals for numerical modelling … It’s not usually used in finance

In this case, when working with this type of equation Simulink is an appropriate modelling and presentation environment It should be used in finance where it’s the right tool

<<Figure 1, phase portrait of the system.>>

**Sections:**

* **Introduction/motivation**
  + This article shows how MATLAB and Simulink ….
* **Creating the model using MATLAB**
  + Should be familiar to the main audience
  + ode45
* **Creating the model using Simulink**
  + More details needed (intro Simulink, block diagrams)
  + More hand-holding
* **Parallelization**
  + Straightforward parfor for both cases
  + Visualization
* **Comparison between the MATLAB and Simulink approaches**
  + Discussion feedback loops clearly graphically represented in Simulink
  + 2x2 summary table (presentation styles/modelling styles – what types of models are suitable)? Different ways of thinking about the models.

%% MATLAB/Simulink plus/minus section.

% \* Simulink plus: work the equations "as is" - no transformation/rewriting

% of the equations required to work with Simulink, whereas with ODE45 you

% need to write the equations in standard form.

% \* Time-varying parameters: roughly the same for both approaches. Put in

% further improvements section.

% \* Simulink plus: can compute the derivatives of f(k) automatically,

% whereas in MATLAB you could use Symbolic Toolbox to do this, but another

% approach is to use two files, one for f(k) and one for f'(k).

* + <https://en.wikipedia.org/wiki/Ramsey%E2%80%93Cass%E2%80%93Koopmans_model>
  + More of an “academic” question, widely studied in university degrees/modules, but should be of interest to central banks and possibly others
  + Much easier to understand than DSGE (it’s a 2D system) – should highlight how easy it is to get started with Simulink and build something useful
  + Would like to make a nice visualisation of the phase plane (see wiki page for a simple visualisation of this)
  + Run the Simulink model in parallel to create many distinct trajectories for the phase plane.

------------------------------------------

Gentle introduction to Simulink, with parfor.

Simulink adds value for graphically setting

up the equations. Simulink use cases outside of

the usual engineering world. Applications

of Simulink.

Compare with a MATLAB implementation using

ODE45.