Using GEMPACK Runge-Kutta solution methods with the GTAP model

GTAP conference, 17-19 June, 2026

Erwin Corong, Michael Jerie, Florian Schiffmann

Centre of Policy Studies
Victoria University
300 Queen Street
Melbourne Victoria 3000

Telephone: +61 3 9919 1437
E-mail: michael.jerie@vu.edu.au

Introduction

Runge-Kutta solution methods added to GEMPACK (release 12.2, Dec 2024)

Many good properties:

- Faster convergence than other available methods (Euler, Midpoint, Gragg)
- Computationally cheaper than Richardson extrapolation
- Improved automatic accuracy (adaptive step size for RK)
- Error estimates given for some of the RK methods (HAR file format)
- Better diagnostics for simulation problems

Introduction

Implementation of the Runge-Kutta in GEMPACK
by
Florian Schiffmann
CoPS, Victoria University, Melbourne.

The Runge-Kutta methods in GEMPACK are now thoroughly tested – used and refined in-house in CoPS for more than 2 years before release in GEMPACK 12.2 (December 2024).

The CGE problem as Initial Value Problem

$$\begin{pmatrix} F_1(\mathbf{X},\mathbf{Y}) \\ F_2(\mathbf{X},\mathbf{Y}) \\ \vdots \\ F_m(\mathbf{X},\mathbf{Y}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

The CGE problem in Levels
System of nonlinear Equations

X = vector of exogenous
Y = vector of endogenous

We can construct an initial solution (X^0, Y^0) and shocks X
Perform a trivial linearization of the shocks:

$$\mathbf{X} = \mathbf{X_0} + t\Delta\mathbf{X}$$

The CGE problem as IVP (II)

Derive F with respect to t (using chain rule):

$$\frac{d\mathbf{F}}{dt} = \frac{\partial \mathbf{F}}{\partial \mathbf{X}}\frac{\partial \mathbf{X}}{\partial t} + \frac{\partial \mathbf{F}}{\partial \mathbf{Y}}\frac{\partial \mathbf{Y}}{\partial t} = 0$$

Using the shock vector we know:

$$\frac{\partial \mathbf{X}}{\partial t} = \Delta \mathbf{X}$$

From equation system and closure:

$$\frac{\partial \mathbf{F}}{\partial \mathbf{X}} = \mathbf{J_F}(\mathbf{X}) = \begin{pmatrix} \frac{\partial \mathbf{F_1}}{\partial \mathbf{X_1}} & \frac{\partial \mathbf{F_1}}{\partial \mathbf{X_2}} & \cdots & \frac{\partial \mathbf{F_1}}{\partial \mathbf{X_n}} \\ \frac{\partial \mathbf{F_2}}{\partial \mathbf{X_1}} & \frac{\partial \mathbf{F_2}}{\partial \mathbf{X_2}} & \cdots & \frac{\partial \mathbf{F_2}}{\partial \mathbf{X_n}} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \mathbf{F_n}}{\partial \mathbf{X_1}} & \frac{\partial \mathbf{F_n}}{\partial \mathbf{X_2}} & \cdots & \frac{\partial \mathbf{F_n}}{\partial \mathbf{X_n}} \end{pmatrix} \qquad \frac{\partial \mathbf{F}}{\partial \mathbf{Y}} = \mathbf{J_F}(\mathbf{Y}) = \begin{pmatrix} \frac{\partial \mathbf{F_1}}{\partial \mathbf{Y_1}} & \frac{\partial \mathbf{F_1}}{\partial \mathbf{Y_2}} & \cdots & \frac{\partial \mathbf{F_1}}{\partial \mathbf{Y_m}} \\ \frac{\partial \mathbf{F_2}}{\partial \mathbf{Y_1}} & \frac{\partial \mathbf{F_2}}{\partial \mathbf{Y_2}} & \cdots & \frac{\partial \mathbf{F_2}}{\partial \mathbf{Y_m}} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \mathbf{F_n}}{\partial \mathbf{Y_1}} & \frac{\partial \mathbf{F_n}}{\partial \mathbf{Y_2}} & \cdots & \frac{\partial \mathbf{F_n}}{\partial \mathbf{Y_m}} \end{pmatrix}$$

The CGE problem as IVP (III)

Rewrite derivative:

$$\frac{\partial \mathbf{F}}{\partial t} = \mathbf{J_F}(\mathbf{Y})\frac{\partial \mathbf{Y}}{\partial t} + \mathbf{J_F}(\mathbf{X})\Delta\mathbf{X} = 0$$

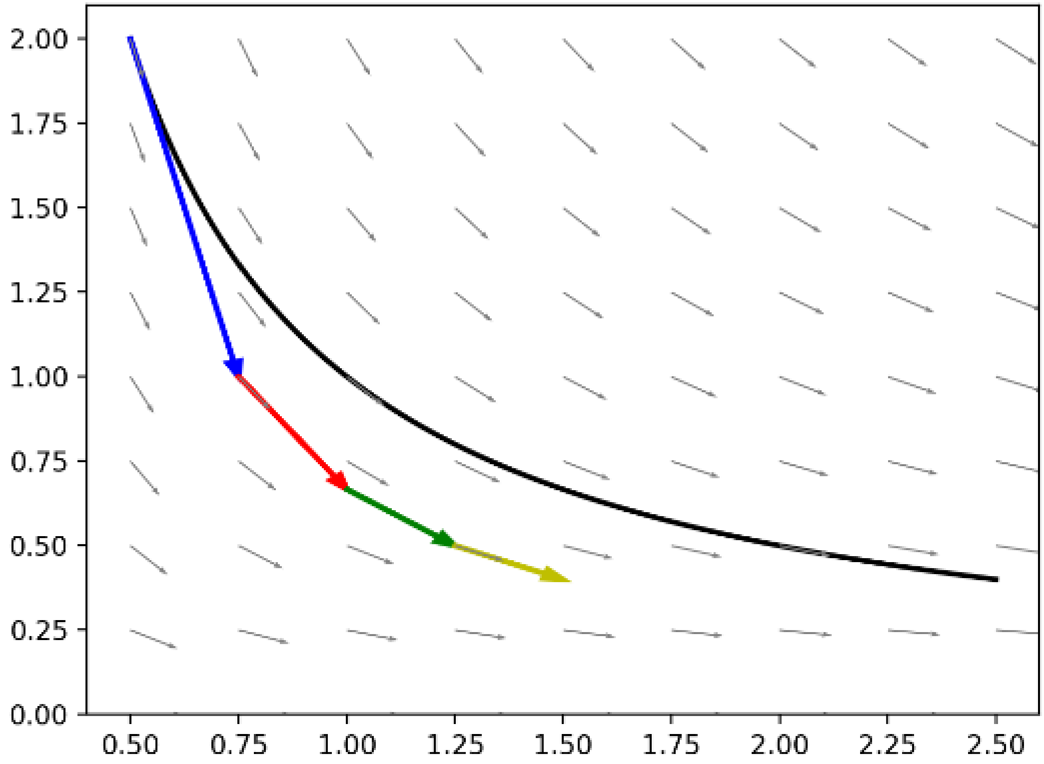
Rearrange we have ODE for solution as:

$$\frac{\partial \mathbf{Y}}{\partial t} = \underbrace{\mathbf{J_F}(\mathbf{Y})^{-1}}_{\text{LHS matrix}}\underbrace{\mathbf{J_F}(\mathbf{X})\Delta\mathbf{X}}_{\text{RHS vector}}$$

By providing initial solution (database) can be solved as IVP

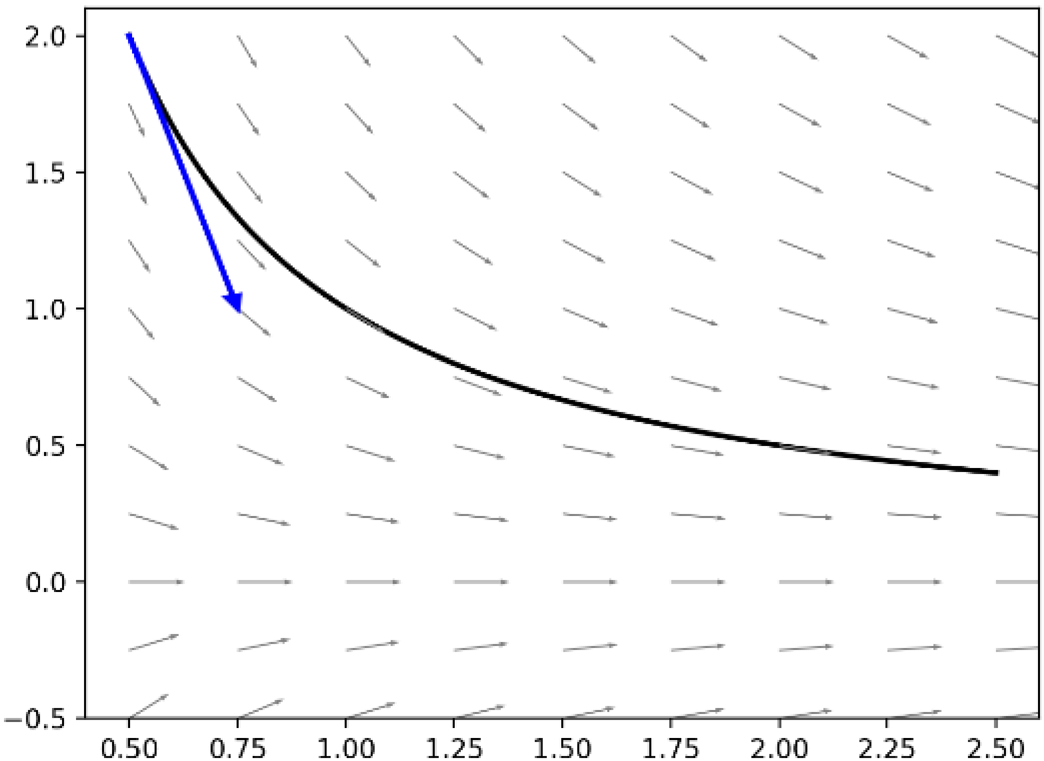
Illustration of Euler

Integrate: $$\frac{dy}{dx} = -\frac{y}{x}$$



RK4 (poking around)

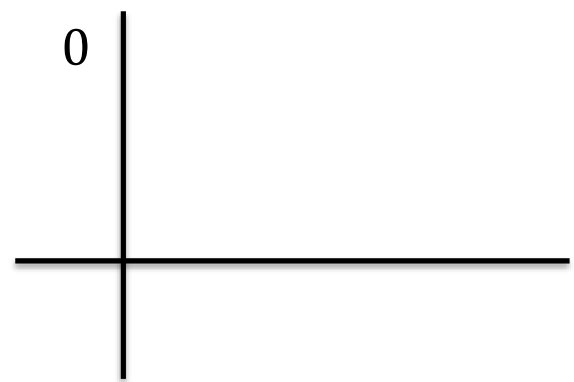
Integrate: $$\frac{dy}{dx} = -\frac{y}{x}$$



$$x_0 = 0.5$$

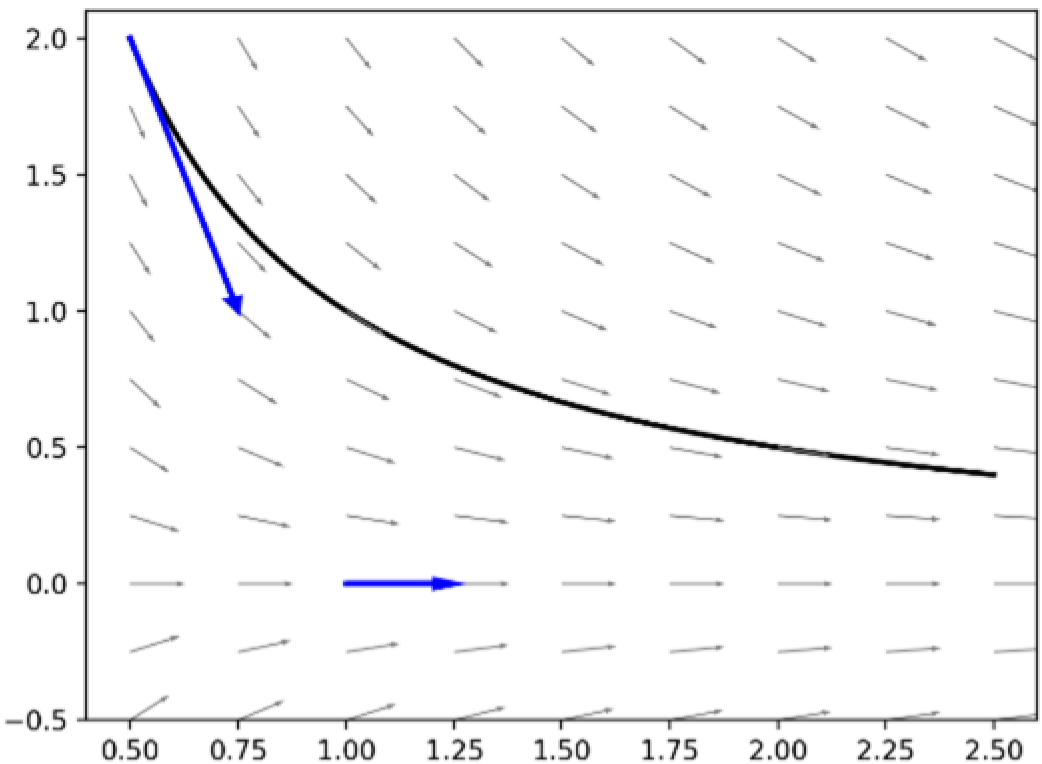
$$y_0 = 2.0$$

$$g_1 = -\frac{y_0}{x_0}$$



RK4 (poking around)

Integrate: $$\frac{dy}{dx} = -\frac{y}{x}$$



$$x_1 = x_0 + 0.5h$$

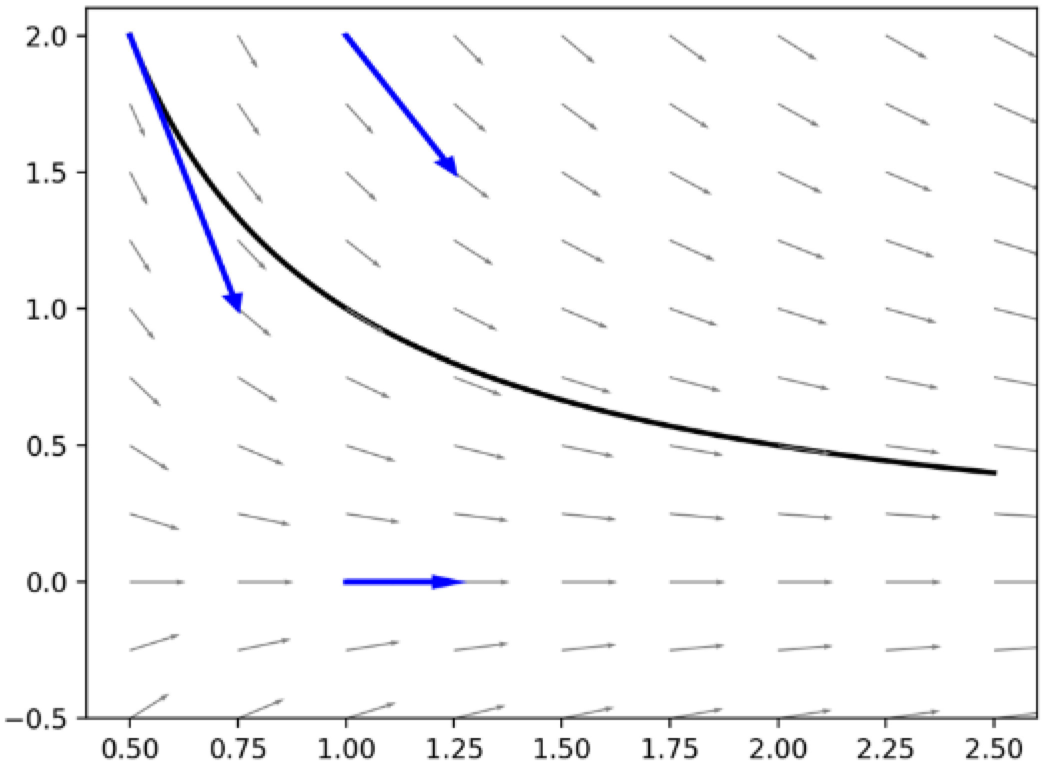
$$y_1 = y_0 + 0.5g_1$$

$$g_2 = -\frac{y_1}{x_1}$$

$$\begin{array}{c|c}
0 & \\
0.5 & 0.5 \\
 & \\
\hline
 &
\end{array}$$

RK4 (poking around)

Integrate: $$\frac{dy}{dx} = -\frac{y}{x}$$



$$x_2 = x_0 + 0.5h$$

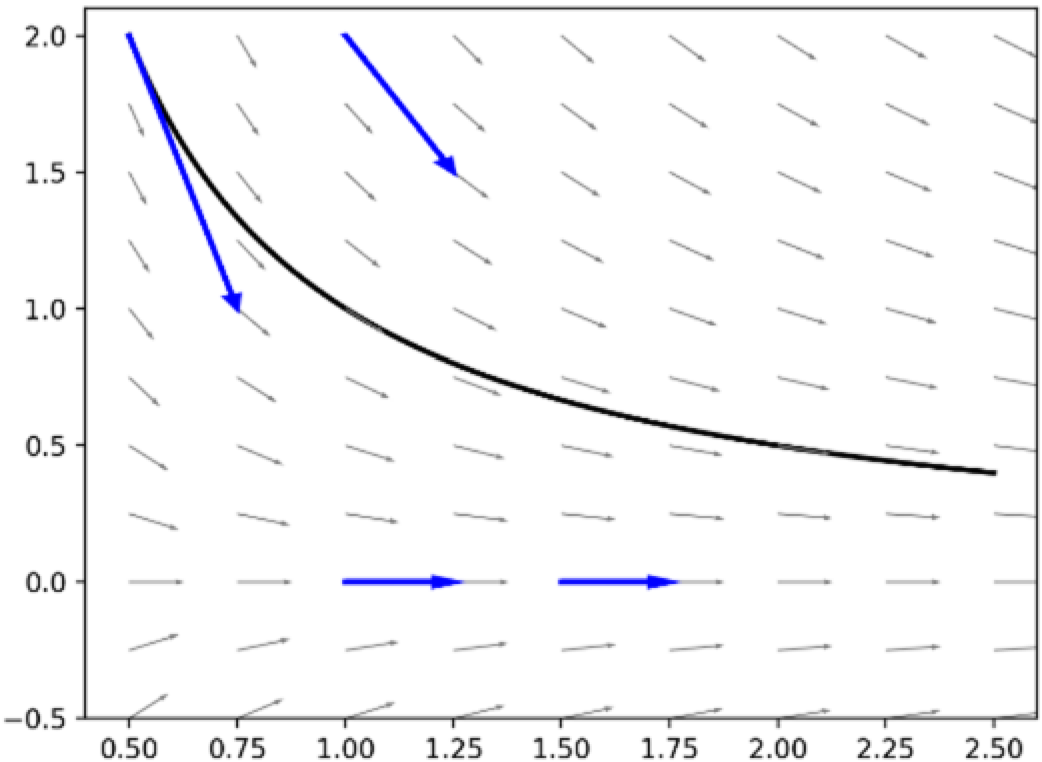
$$y_2 = y_0 + 0.5g_2$$

$$g_3 = -\frac{y_2}{x_2}$$

$$\begin{array}{c|ccc}
0 & & & \\
0.5 & 0.5 & & \\
0.5 & 0 & 0.5 & \\
\hline
 & & &
\end{array}$$

RK4 (poking around)

Integrate: $$\frac{dy}{dx} = -\frac{y}{x}$$



$$x_3 = x_0 + h$$

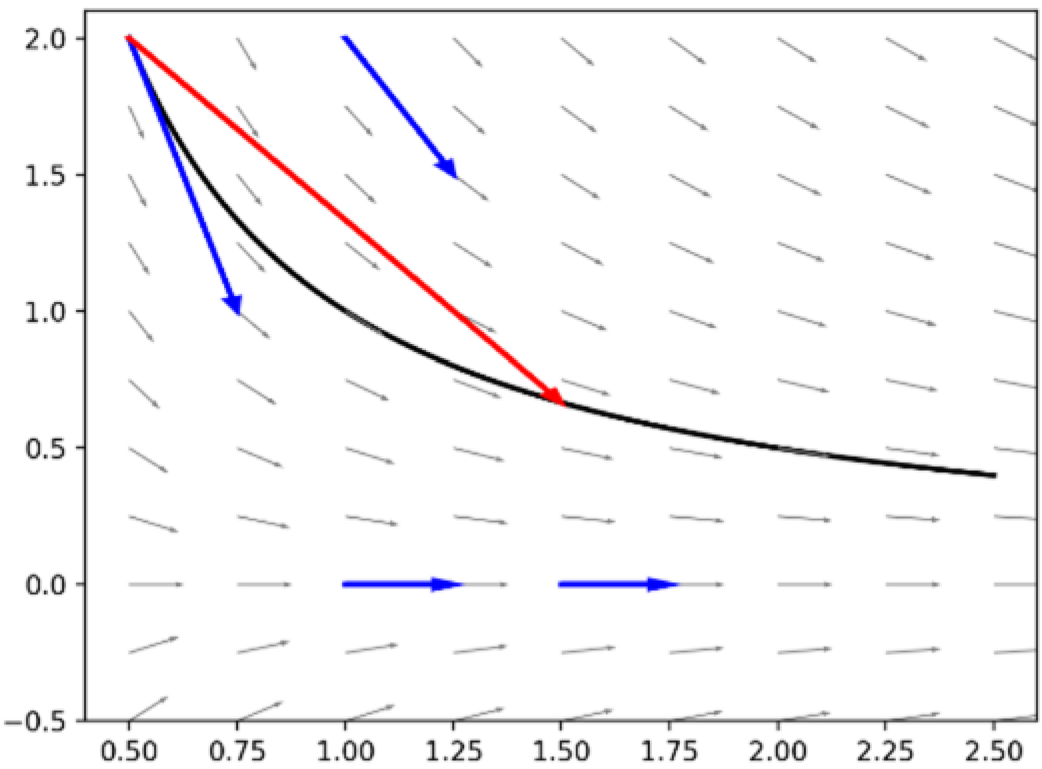
$$y_3 = y_0 + g_3$$

$$g_4 = -\frac{y_3}{x_3}$$

$$\begin{array}{c|cccc} 0 & & & & \\ 0.5 & 0.5 & & & \\ 0.5 & 0 & 0.5 & & \\ 1 & 0 & 0 & 1 & \\ \hline & & & & \end{array}$$

RK4 (using gathered informtion)

Integrate: $$\frac{dy}{dx} = -\frac{y}{x}$$



$$x = x_0 + h$$

$$g = \frac{1}{6}g_1 + \frac{2}{6}g_2 + \frac{2}{6}g_3 \frac{1}{6}g_4$$

$$y = y_0 + g$$

$$\begin{array}{c|cccc}
0 & & & & \\
0.5 & 0.5 & & & \\
0.5 & 0 & 0.5 & & \\
1 & 0 & 0 & 1 & \\
\hline
 & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6}
\end{array}$$

Order of integrator

Local integration error as Taylor series of step size h

$$\Delta = \sum_{r=1}^{\infty} \frac{1}{r!} C_r h^r = C_1 h + \frac{1}{2} C_2 h^2 + \frac{1}{6} C_3 h^3 + \cdots$$

n^th order integrator has C_1 to $C_n = 0$

Truncating Taylor series at r=n+1

$$\Delta \sim h^{n+1}$$

Euler 1^st order, local error $\sim h^2$, multistep error $\sim h$
RK4 4^th order, local error $\sim h^5$, multistep error $\sim h^4$

Good integrators try minimizing higher order C

Explicit Runge-Kutta methods

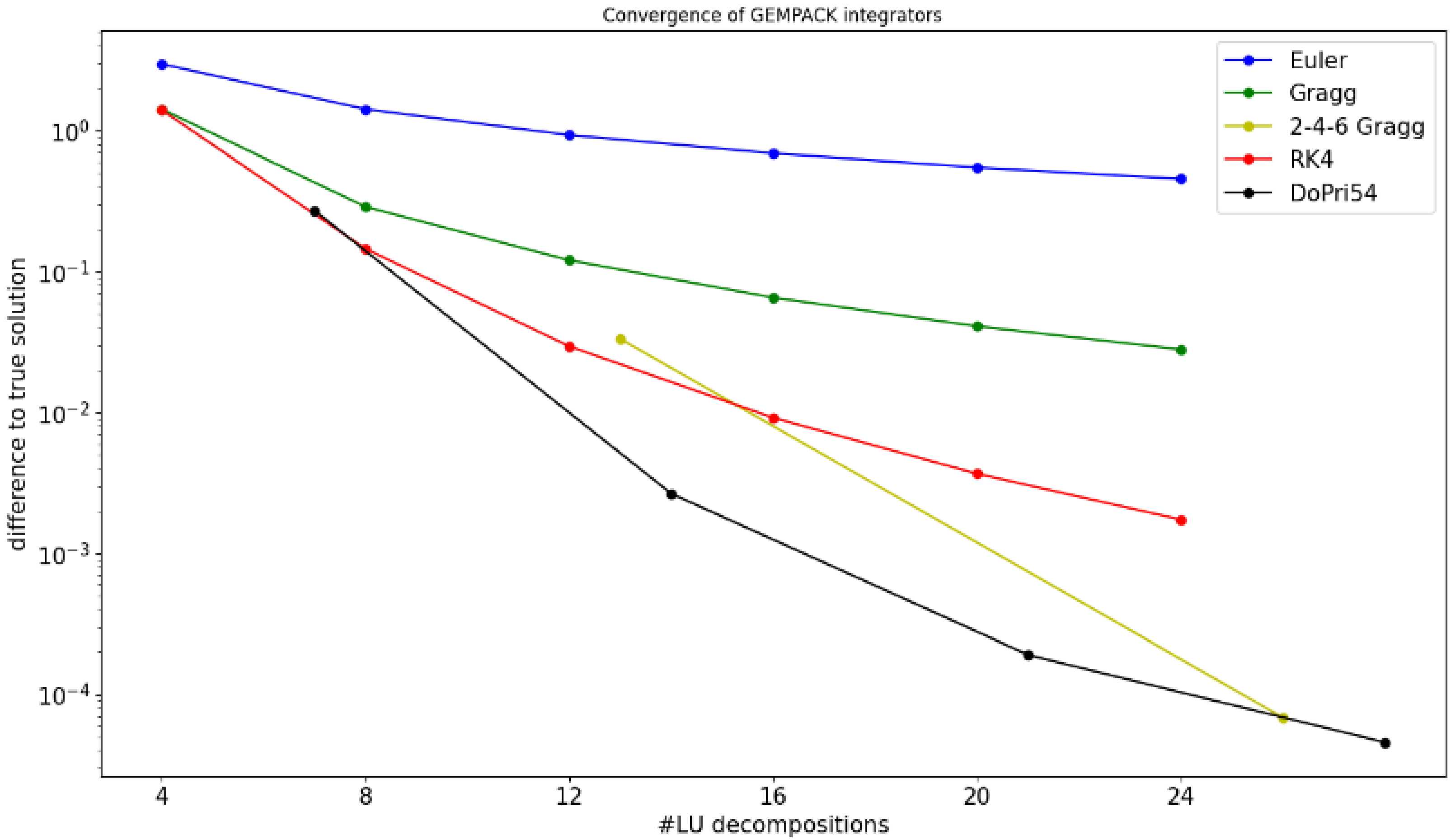
RK2: 2^{nd} order, 2 stages

RK4: 4^{th} order, 4 stages

How fast converging

How much work

Convergence



Embedded methods

RK4
(4th order)

$$
\begin{array}{c|cccc}
0 & & & & \\
0.5 & 0.5 & & & \\
0.5 & 0 & 0.5 & & \\
1 & 0 & 0 & 1 & \\
\hline
 & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6}
\end{array}
$$

Euler
(1st order)

$$
\begin{array}{c|c}
0 & \\
\hline
 & 1
\end{array}
$$

Compute higher order
Get lower order for free

Midpoint
(2nd order)

$$
\begin{array}{c|cc}
0 & & \\
0.5 & 0.5 & \\
\hline
 & 0 & 1
\end{array}
$$

Embedded Runge-Kutta methods

Construction of useful embedded methods might require additional trial point.

I.e., increased computational cost for error estimate

Bogacki-Shampine: 3^{rd} order, 2^{nd} order embedded, 4 stages
Dormand-Prince: 5^{th} order, 4^{th} order embedded, 7 stages

How to adjust steps

New step size

tolerance t

$$h_t = \left(\frac{t}{\Delta_{estimate}}\right)^{\frac{1}{n+1}} h_0$$

current step h_0

error estimate

$\Delta_{estimate} < t$ (within t)

- Perform step
- Assume error for next step similar to current
- Increase step size

$\Delta_{estimate} > t$ (exceeds t)

- Discard step
- Use current error to compute new step size
- Redo step with reduced step size

Other reasons for redoing steps

Assertion failure / Range check error

Assertions and Range checks indicate Values leaving valid range

If values are not valid, results for this stage will be inconsistent or lead to arithmetic errors.

In both cases step size adjustment and step re-computation is inevitable at the last stage (with mispredicted adjustment)

Percent changes below -100%

To avoid unnecessary work and avoid bad adjustments GEMPACK stops current step and immediately recomputes with half the step size.

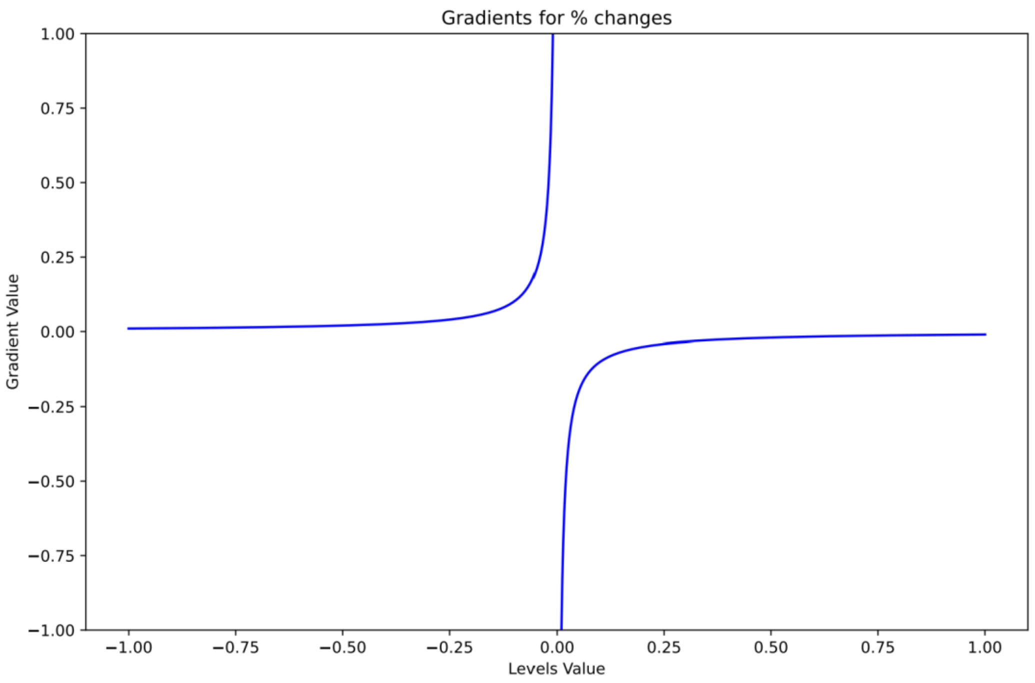
No more getting past -100% !!!

Singularity in gradients at -100%

Failure for higher order integrators

Adaptive step size automatically rescales step

Non adaptive terminates with error



Using from CMF

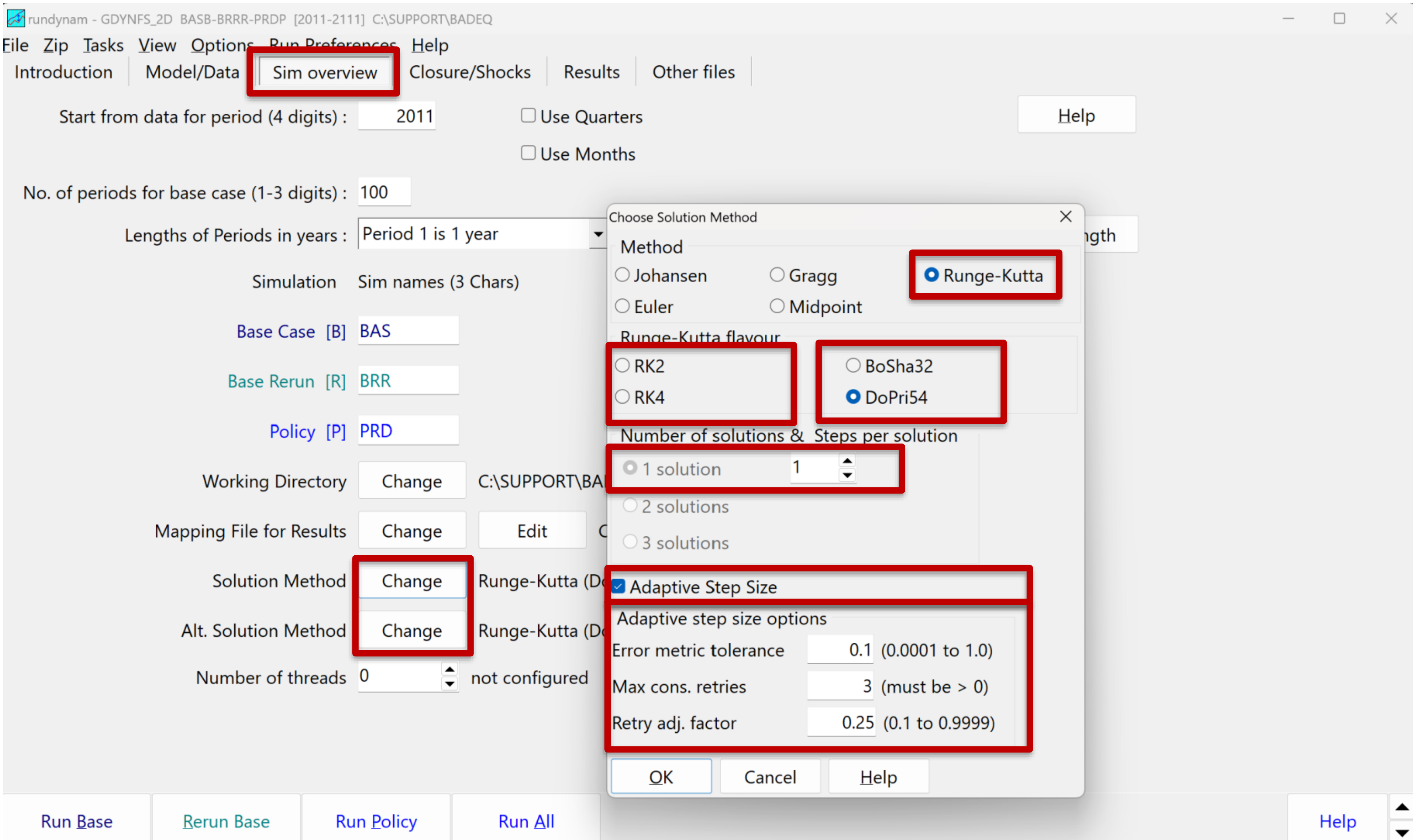
Methods without error estimate (non adaptive)

Method = RungeKutta;
RK flavor = RK2 | RK4;
Steps = n;

Methods with error estimate (adaptive)

Method = RungeKutta;
RK flavor = BoSha32 | DoPri54;
Steps = n;
Adaptive step size = yes | no;
epsTolerance = 0.1;

Using Rundynam



Error metric

Using relative error

$$E^{rel} = \frac{\Delta}{V_{res}}$$

Avoids problem of different scale as values are normalized
Problem V_{res} close to zero produces large relative errors

$$E^{GP} = \frac{\Delta}{\max(1, V_{res})}$$

GEMPACK error metric uses relative error for $V_{res} > 1$
and absolute error for $V_{res} < 1$

Adpative step size based on error metric

Investigate .acc file

error.acc in C:\Users\e5107552\Documents

File Contents Export History Search Programs Help

	Header	Type	Dimension	Coeff	Total	Max ▼	Name
2	0001	RE	COM*SRC*IND	x1	3.12	0.113	Cumulative Error Metric Estimate for variable x1
156	0155	RE	IND	finv3	0.118	0.106	Cumulative Error Metric Estimate for variable finv3
85	0084	RE	COM	x0imp	0.149	0.102	Cumulative Error Metric Estimate for variable x0imp
84	0083	RE*	COM*SRC*DEST	delSale	0.209	0.066	Cumulative Error Metric Estimate for variable delSale
5	0004	RE	COM	x4	0.086	0.062	Cumulative Error Metric Estimate for variable x4
12	0011	RE	COM*MAR	x4mar	0.342	0.062	Cumulative Error Metric Estimate for variable x4mar
190	0189	RE	COM*REG	regx4	0.685	0.062	Cumulative Error Metric Estimate for variable regx4
52	0051	RE	COM*IND	q1	1.39	0.056	Cumulative Error Metric Estimate for variable q1
3	0002	RE	COM*SRC*IND	x2	2.53	0.053	Cumulative Error Metric Estimate for variable x2
18	0017	RE	IND*OCC	x1lab	0.460	0.040	Cumulative Error Metric Estimate for variable x1lab
33	0032	RE	IND	x1lab_o	0.057	0.040	Cumulative Error Metric Estimate for variable x1lab_o

ErrorMetric is absolute value.

Use Viewhar sorting on Max on title page

Check for badly converged components`

x1	1 dom	2 imp
1 BroadAcre	0.001	0.080
2 OtherAgric	0.003	0.019
3 ForestFish	0.000	0.009
4 Mining	0.003	0.021
5 MeatDairy	0.022	0.014
6 OthFoodProds	0.001	0.006
7 DrinksSmokes	0.003	0.024
8 Textiles	0.000	0.016
9 ClothingFtw	0.002	0.007
10 WoodProds	0.000	0.009
11 PaperPrint	0.001	0.110
12 Petrol_CoalP	0.001	0.018

Reasons for low accuracy

Large shocks

- Model produces highly nonlinear response
- Sizable error for most variables in .acc file
- Reduce Tolerance

Model error

- Bug in Equations or updates
- Sizable error for all components in few connected variables
- Inspect equations/updates relating to these variable

Database error

- Imbalance in database (not a valid initial state)
- Sizable error for few components in few connected variables
- Check database consistency

Investigate .log file

Embedded RK methods print worst component and accuracy

```
~2941,2,4:  RK | Variable accuracy "face" number for this simulation is 4 (out of 10).
~2942,2,4:  RK |
~2943,2,4:  RK | worst error estimtate of 0.12000 has been found at
~2944,2,4:  RK |    x0dom("LNG")
```

Embedded RK methods print summary of adjustments

```
~9077,2,4:  RK |              SUMMARY OF THE STEPS TAKEN IN ADAPTIVE STEP SIZE MODE
~9078,2,4:  RK |
~9079,2,4:  RK |        % of Sim complete |  StepSize (in %)  |  Status  |  Reason
~9080,2,4:  RK |        -------------------------------------------------------------------
~9081,2,4:  RK |             0.0000      |         25.0000   |   PASS   |
~9082,2,4:  RK |            25.0000      |         42.5000   |   FAIL   | assertion
~9083,2,4:  RK |            25.0000      |         21.2500   |   PASS   |
~9084,2,4:  RK |            46.2500      |         27.3582   |   FAIL   | assertion
~9085,2,4:  RK |            46.2500      |         13.6791   |   FAIL   | assertion
~9086,2,4:  RK |            46.2500      |          6.8395   |   PASS   |
~9087,2,4:  RK |            53.0895      |         10.1366   |   FAIL   | assertion
~9088,2,4:  RK |            53.0895      |          5.0683   |   FAIL   | assertion
~9089,2,4:  RK |            53.0895      |          2.5341   |   PASS   |
~9090,2,4:  RK |            55.6237      |          3.1016   |   PASS   |
~9091,2,4:  RK |            58.7253      |          2.6068   |   PASS   |
~9092,2,4:  RK |            61.3321      |          4.4315   |   PASS   |
~9093,2,4:  RK |            65.7636      |          7.5336   |   PASS   |
~9094,2,4:  RK |            73.2972      |         12.8071   |   PASS   |
~9095,2,4:  RK |            86.1043      |         13.8957   |   PASS   |
```

Read the log or search for “RK |”

Tracing the RK | convergence info lets us identify the problems

```
~4648,2,4:  RK | ACCURACY OK. Changing stepsize 0.25000000 -> 0.42500001
~4649,2,4:  RK |                 25.000% of the simulation have been completed.
~4871,2,4:  RK | A Range check or Assertion failed.
~4872,2,4:  RK | Changing stepsize from 0.42500001 -> 0.21250001
~5212,2,4:  RK | ACCURACY OK. Changing stepsize 0.21250001 -> 0.27358173
~5213,2,4:  RK |                 46.250% of the simulation have been completed.
~5435,2,4:  RK | A Range check or Assertion failed.
~5436,2,4:  RK | Changing stepsize from 0.27358173 -> 0.13679087
~5654,2,4:  RK | A Range check or Assertion failed.
~5655,2,4:  RK | Changing stepsize from 0.13679087 -> 0.06839543
~5995,2,4:  RK | ACCURACY OK. Changing stepsize 0.06839543 -> 0.10136558
~5996,2,4:  RK |                 53.090% of the simulation have been completed.
~6214,2,4:  RK | A Range check or Assertion failed.
~6215,2,4:  RK | Changing stepsize from 0.10136558 -> 0.05068279
~6435,2,4:  RK | A Range check or Assertion failed.
~6436,2,4:  RK | Changing stepsize from 0.05068279 -> 0.02534139
~6778,2,4:  RK | ACCURACY OK. Changing stepsize 0.02534139 -> 0.03101602
~6779,2,4:  RK |                 55.624% of the simulation have been completed.
~7123,2,4:  RK | ACCURACY OK. Changing stepsize 0.03101602 -> 0.02606782
~7124,2,4:  RK |                 2 variables below but close to the threshold were ignored.
~7125,2,4:  RK |                 58.725% of the simulation have been completed.
~7469,2,4:  RK | ACCURACY OK. Changing stepsize 0.02606782 -> 0.04431529
~7470,2,4:  RK |                 61.332% of the simulation have been completed.
~7812,2,4:  RK | ACCURACY OK. Changing stepsize 0.04431529 -> 0.07533600
~7813,2,4:  RK |                 65.764% of the simulation have been completed.
~8157,2,4:  RK | ACCURACY OK. Changing stepsize 0.07533600 -> 0.12807120
~8158,2,4:  RK |                 73.297% of the simulation have been completed.
~8500,2,4:  RK | ACCURACY OK. Changing stepsize 0.12807120 -> 0.13895685
~8501,2,4:  RK |                 86.104% of the simulation have been completed.
~8845,2,4:  RK | ACCURACY OK. Successfully completed Simulation
```

We find in the log file near the failed RK step

```
4857   %% Assertion "No negatives in USE matrix" does not hold
4858    when quantifiers are ("Hogs","imp","INV","SENE").
```

Read the log or search for “RK |”

RK methods write offending components to log

Adaptive step size will immediately redo step
with reduced step size to obtain more accurate solution

```
~3959,2,4:  RK | The percent change variable
~3960,2,4:  RK |     x1lab("LNG","Manager")
~3961,2,4:  RK |     x1lab("OthNonFeOre","Manager")
~3962,2,4:  RK |     x1lab("MiningSrv","Manager")
~3963,2,4:  RK |     x1lab("ForTourism","Manager")
~3964,2,4:  RK |     x1lab("ForStudent","Manager")
~3965,2,4:  RK |     x1lab("LNG","Professional")
~3966,2,4:  RK |     x1lab("OthNonFeOre","Professional")
~3967,2,4:  RK |     x1lab("MiningSrv","Professional")
~3968,2,4:  RK |     x1lab("ForTourism","Professional")
~3969,2,4:  RK |     x1lab("ForStudent","Professional")
~3970,2,4:  RK |     and others not reported.
~3971,2,4:  RK | went below -100%. For Runge Kuttan methods,
~3972,2,4:  RK | this means that the solution will not be reliable.
~3973,2,4:  RK | Redoing step with reduced step size
~3974,2,4:  RK | Preparing coefficients values for redoing step.
~3976,2,4:  RK | Changing stepsize from 0.20062359 -> 0.10031180 due to
~3977,2,4:  RK | sign change in a percent change variable.
```

Conclusions

Things you will immediately appreciate

- RK integrators outperform existing integrators in GEMPACK
- RK avoids numerical instabilities known to exist for Richardson extrapolation
- Adaptive step size more stable, faster and more expressive than automatic accuracy
- Improved log file output to pinpoint problems

Things that might take a bit longer

- RK unforgiving of model/database errors, e.g.
  'Tried and trusted' models might fail with RK where Euler 'worked'
- Understanding how log and acc file help to find problems and fix them

The new CoPS standard: DoPri54 + adaptive step size

Next steps

Develop a set of worked examples with a GTAP model addressing common problems.

- GTAP
  - Tuning the solution method for accuracy and solution time
  - Using RK error estimates to identify a coding or data error
- GTAP-RD Using RK with adaptive step sizes and modifying default control parameters where necessary
- GTAP-E Industry output falls by -100%
- GTAP-Power Identifying solution problems related to the energy nest