EC7025 - SFC

Antoine Godin

Autumn 2016

Welcome!

- Welcome and thanks for taking this module
- intended for students who have a good mathematical background and wish to expand their knowledge of advanced economic modelling, data calibration, and model simulation
- On completion of the module, the students will have a deep understanding of various modelling practices in economics and their relevance in policy making, as well as an in-depth knowledge of data calibration and model simulation

Aims

- introduce students to pluralistic modelling practices such as Stock-Flow-Consistent (SFC) modelling, Kaleckian growth models, Post-Keynesian and Marxian models of growth and debt cycles and Agent-Based Modeling (ABM)
- equip students with the necessary knowledge to build and simulate different small and medium scale models.
- equip students with an in-depth knowledge of data analysis and model simulation.
- introduce students to the basics of system dynamics approach

Logistics

- Module leader: Antoine Godin
  - Office: HH0009
  - Office hours: Tue, Wed, Thu: 11-12
  - Email: a.godin@kingston.ac.uk
- Lecturer: Devrim Yilmaz
  - Office: HH1011
  - Office hours: Mon: 4.00-5.30 & Tue: 2.30-4.00
  - Email: s.yilmaz@kingston.ac.uk

Outline

- Week 1-2: Understanding National Accounts
- Week 3-5: Stock-Flow Consistent Modelling
- Week 6-8: Heterodox Theories of Distribution and Growth
- Week 9-12: Growth and Debt Cycles
- Week 13-14: Modelling System Dynamics
- Week 15–16: Introduction to Agent - Based Modelling
- Week 17–22: Stock-Flow Consistent Agent Based Models

Assessment

- Three research projects (Nov. 16, Feb. 15 and Apr. 7), each 25%
- One class test (Jan. 11), 25%

Week 1 to 5: SFC modelling and national accounts

- Get you fluent in ESA2010 language: what does S11, P3, MIO_NAC or EL stand for?
- Know your way around Eurostat and ONS database
- Introduction to R and relevant packages (pdfetch, PKSFC)
- Build and calibrate medium scale SFC model

National Accounts and Stock-Flow Consistent Modelling

- SFC models are based on a set of different tables that are more or less connected to real data and national accounts.
- Balance Sheets
- Transaction Flow Matrix
- Full Integration Matrix
- References: Godley and Lavoie Ch. 2, Caverzasi and Godin (2015), Eurostat and ONS Blue book

Stock-Flow Accounting

- Started with Copeland (1949) and his Social Accounting for moneyflows, picked up by Denizet (1969) and many others…
- Highlights the importance to incorporate monetary and financial processes into national accounts such as NIPA.
- Very close to Keynes's idea to integrate financial and income accounting.
- Idea is to be able to answer Copeland questions:
  - when total purchases of our national product increase, where does the money come from to finance them?
  - when purchases of our national product decline, what becomes of the money that is not spent?

Balance Sheets

- Balance sheets display the assets, liabilities and the balancing item net worth.

- Most of you are familiar with basic balance sheets such as the households balance sheet for the households at the end of 2015 in the United Kingdom (source Eurostat).

As	sets Lia	bilities
Produced non-financial asset	5238911	0
Non-produced non financial assets	2507	0
Currency and deposits	1412172	0
Securities other than shares	91487	2226
Loans	18745	1563594
Shares and other equity	777956	0
Insurance technical reserves	3708339	69232
Other accounts receivable/payable	6847	2842
Net Worth	0	9738640

Non-financial corporations

	Assets	Liabilities
Produced non-financial asset	2097113	0
Non-produced non financial assets	0	0
Currency and deposits	546337	0
Securities other than shares	65083	384631
Loans	262326	965308
Shares and other equity	837888	2475658
Insurance technical reserves	4029	1056253
Other accounts receivable/payable	29976	50912
Net Worth	0	-1146126

- note negative net worth due to market value of equity
- capital stock are at market value (replacement cost) and not historical costs

Financial corporations

	Assets	Liabilities
Produced non-financial asset	162277	0
Non-produced non financial assets	0	0
Currency and deposits	3129884	5141459
Securities other than shares	3520842	1829957
Loans	3418831	1553526
Shares and other equity	2986514	2046996
Insurance technical reserves	1296602	3844915
Other accounts receivable/payable	6069641	6031487
Net Worth	0	64149

- Central Banks are in the financial corporations

Goverment

	Assets	Liabilities
Produced non-financial asset	2097113	0
Non-produced non financial assets	0	0
Currency and deposits	546337	0
Securities other than shares	65083	384631
Loans	262326	965308
Shares and other equity	837888	2475658
Insurance technical reserves	4029	1056253
Other accounts receivable/payable	29976	50912
Net Worth	0	-1146126

Balance Sheets in SFC

- When you are constructing the balance sheets of your model, you should first consider which assets you will include in your model.
  - Real assets: Capital stock, housing etc.
  - Financial assets/liabilities: cash, deposits, bills, bonds, loans, equities, derivatives, bank reserves, monetary gold, SDR etc.
- These assets will contain the economic wealth accumulated by economic agents. So your balance sheet matrix must contain the assets you decide to include in your model, and it should clearly identify which sectors in your economy hold which assets and which liabilities. As usual, the difference between assets and liabilities will yield net worth.

Example

	HHs	Firms	Gov.	Banks	C. B.	Sum
Capital	+Kh	+Kf				+K
Money	+Hh			+Hb	-H	0
Bills	+Bh		-Bs	+Bb	+Bcb	0
Loans	-Lh	-Lf	+L	0	0	0
Equities	+Ef	-Ef	0	0	0	0
Equities	+Eb	0	-Eb	0	0	0
Net worth	-NWh	-NWf	-NWb	-NWg	0	-K
Sum	0	0	0	0	0	0

Sectorial accounts (from Eurostat)

- Sector accounts record every transaction between sector and the change in financial assets and liabilities.
- Transactions are grouped in categories having a distinct economic meaning. Each non-financial transaction is recorded as an increase in the "resources" of a sector and an increase in the "uses" of another.
- Shown in a sequence of accounts, each of which covers a specific economic process.
- Two main categories: current accounts and accumulation accounts
  - *Current accounts* record transactions that do not involve the purchase or sale of financial or non-financial assets. Final balancing item is saving
  - *Accumulation accounts* record net acquisition of non-financial and financial assets, and the net incurrence of liabilities. Also show other changes in balance sheets, such as revaluations and write-offs of bad debts
  - The accumulation accounts explain all the changes in the (non-financial and financial) balance sheets

Example for households in the UK in 2014

	Households
Total Income	9851688
Taxes	-1442130
Social Contributions	-2425244
Social Benefits	2635931
Other transfers	109116
[Gross Disposable Income	8729361]

	Households
Consumption	-8012603
Adjustments in Pensions	201553
[Gross Savings	918310]
Gross Capital Formation	-716849
Capital Transfer	14213
Net Non-Produced NF Assets	6360
Net Lending Position	222034

Transaction flow matrix

1. The transactions flow matrix consists of three separate parts.
   - On the top rows of the matrix, you will have output expressed as expenditures, which by definition is given by $Y = C + I + G(+X - M)$
   - The matrix should clearly identify the consumption and investment by the sectors in your model.
2. The second part of the transactions flow matrix outlines output using an income approach.
   - Depending on how disaggregated your model is and how many assets you have included in your model, this part may include various sources of income for your sectors

Assume a closed economy,

- Households work for firms in exchange for wages, consume, invest in housing and hold cash, equities of firms and banks, government bills and deposits as financial assets. Government taxes households, firms and banks and spends, and issues bills to finance its deficit.
- Firms employ households to produce goods and invest in productive capital stock. They use undistributed profits to finance investment and borrow from banks/issue new equity to finance any shortfall. -Banks lend to households and firms, hold bills, accept deposits from households and distribute part of their profits to households. They do not invest in tangible capital.
- Central bank holds government bills and transfers its profits to the government

Example of Transaction Flow Matrix

Transaction flow matrix, part 2

- Once you have identified the first two parts of the transactions flow matrix, you have a complete picture of income sources and expenditures of each sector in your model.
- Naturally, the difference between income and expenditures yields the savings of each sector, which are then allocated to real and financial assets to accumulate wealth.
- The last part of the full flow matrix shows which assets and liabilities these savings/dis-savings have been channeled to.
- In order to ensure that each column adds up to zero, we have to record the changes in assets/liabilities in a non-intuitive way and record changes in assets with a (-) sign and change in liabilities with a (+) sign.
- Therefore, each column now shows

Income – Expenditures – Change in assets/liabilities =0

	Households	Production firms		Banks		Government	Central Bank		
	(1)	Current (2)	Capital (3)	Current (4)	Capital (5)	(6)	Current (7)	Capital (8)	Σ
Consumption	$-C$	$+C$							0
Investment	$-I_{\mathrm{h}}$	$+I$	$-I_{\mathrm{f}}$						0
Govt. exp.		$+G$				$-G$			0
Wages	$+WB$	$-WB$							0
Profits, firms	$+FD_{\mathrm{f}}$	$-F_{\mathrm{f}}$	$+FU_{\mathrm{f}}$						0
Profits, banks	$+FD_{\mathrm{b}}$			$-F_{\mathrm{b}}$	$+FU_{\mathrm{b}}$				0
Profit, central Bk						$+F_{\mathrm{cb}}$	$-F_{\mathrm{cb}}$		0
Loan interests	$-r_{\mathrm{l}(-1)} \cdot L_{\mathrm{h}(-1)}$	$-r_{\mathrm{l}(-1)} \cdot L_{\mathrm{f}(-1)}$		$+r_{\mathrm{l}(-1)} \cdot L_{(-1)}$					0
Deposit interests	$+r_{\mathrm{m}(-1)} \cdot M_{\mathrm{h}(-1)}$			$-r_{\mathrm{m}(-1)} \cdot M_{(-1)}$					0
Bill interests	$+r_{\mathrm{b}(-1)} \cdot B_{\mathrm{h}(-1)}$			$+r_{\mathrm{b}(-1)} \cdot B_{\mathrm{b}(-1)}$		$-r_{\mathrm{b}(-1)} \cdot B_{(-1)}$	$+r_{\mathrm{b}(-1)} \cdot B_{\mathrm{cb}(-1)}$		0
Taxes – transfers	$-T_{\mathrm{h}}$	$-T_{\mathrm{f}}$		$-T_{\mathrm{b}}$		$+T$			0
Change in loans	$+\Delta L_{\mathrm{h}}$		$+\Delta L_{\mathrm{f}}$		$-\Delta L$				0
Change in cash	$-\Delta H_{\mathrm{h}}$				$-\Delta H_{\mathrm{b}}$			$+\Delta H$	0
Change, deposits	$-\Delta M_{\mathrm{h}}$				$+\Delta M$				0
Change in bills	$-\Delta B_{\mathrm{h}}$				$-\Delta B_{\mathrm{h}}$	$+\Delta B$		$-\Delta B_{\mathrm{cb}}$	0
Change, equities	$-(\Delta e_{\mathrm{f}} \cdot p_{\mathrm{ef}} + \Delta e_{\mathrm{b}} \cdot p_{\mathrm{eb}})$		$+\Delta e_{\mathrm{f}} \cdot p_{\mathrm{ef}}$		$+\Delta e_{\mathrm{b}} \cdot p_{\mathrm{eb}}$				0
Σ	0	0	0	0	0	0	0	0	0

Figure 1: Transaction Flow Matrix, source: G&L 2007

Full integration matrix

- Once you have written down the transactions flow matrix, you can move to derive the full integration matrix, which simply shows the changes in net worth of your sectors between the beginning of the period and the end of the period.
- In order to do so, you use the bottom part of the transactions flow matrix with opposite signs in order to make sure increases in assets lead to an increase in net worth and increases in liabilities lead to a decrease in net worth. (Do not forget to add change in tangible capital)
- One further consideration is the change in the value of some stocks of assets between the beginning of the period and the end of the period.
- In order to capture this, you will need to add rows for the assets whose values are subject to such change.
- The last row now becomes the net worth of each sector at the end of the period.

Example of Full Integration Matrix

		Households (1)	Production firms (2)	Banks (3)	Government (4)	Central bank (5)	Σ
	Net worth, end of previous period	NW_{h-1}	NW_{f-1}	NW_{b-1}	NW_{g-1}	0	K_{-1}
Change in net assets arising from transactions	Change in loans	$-\Delta L_h$	$-\Delta L_f$	$+\Delta L$			0
	Change in cash	$+\Delta H_h$		$+\Delta H_b$		$-\Delta H$	0
	Change in deposits	$+\Delta M_h$		$-\Delta M$			0
	Change in bills	$+\Delta B_h$		$+\Delta B_h$	$-\Delta B$	$+\Delta B_{cb}$	0
	Change in equities	$+\Delta e_f \cdot p_{ef} + \Delta e_b \cdot p_{eb}$	$-\Delta e_f \cdot p_{ef}$	$-\Delta e_b \cdot p_{eb}$			0
	Change in tangible capital	$+\Delta k_h \cdot pk$	$+\Delta k_f \cdot pk$				$+\Delta k \cdot pk$
Change in net assets arising from revaluations	Capital gains in equities	$+\Delta p_{ef} \cdot e_{f-1}$ $+\Delta p_{eb} \cdot e_{b-1}$	$-\Delta p_{ef} \cdot e_{f-1}$	$-\Delta p_{eb} \cdot e_{b-1}$			0
	Capital gains in tangible capital	$+\Delta pk \cdot k_{h-1}$	$+\Delta pk \cdot k_{f-1}$				$\Delta pk \cdot (k_{h-1} + k_{f-1})$
	Net worth, end of period	NW_h	NW_f	NW_b	NW_g	0	K

Figure 2: Full Integration Matrix, source: G&L 2007

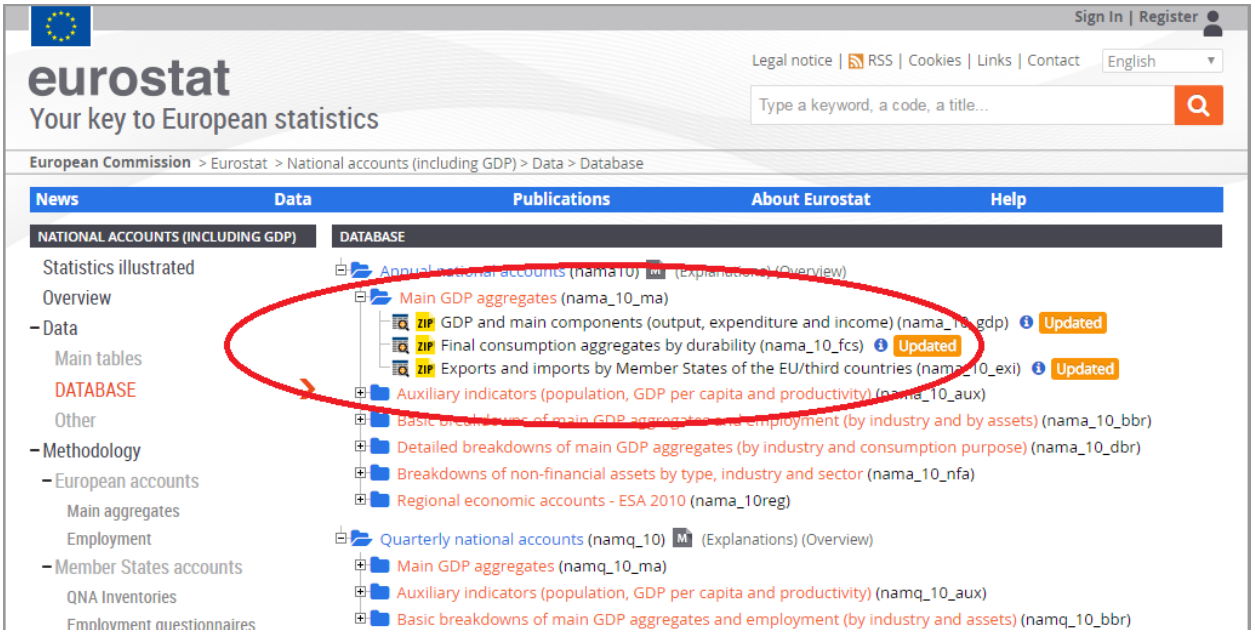


Figure 3: Data Search in Eurostat

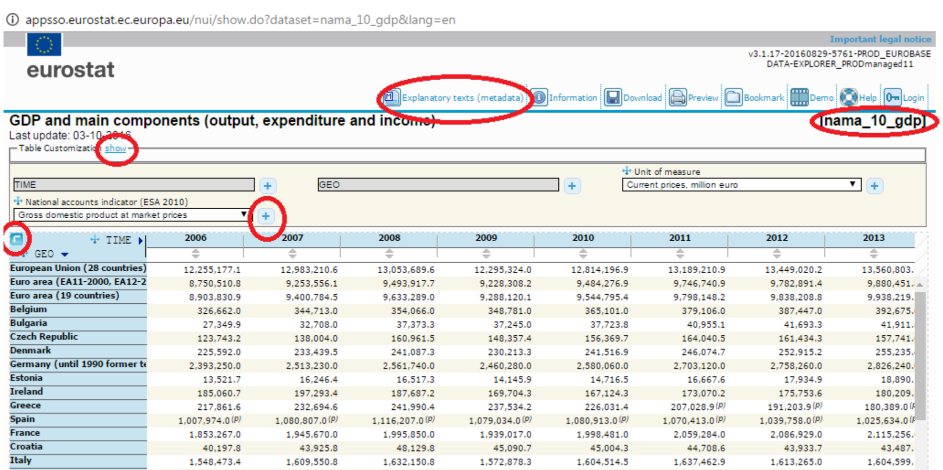


Figure 4: Data Table in Eurostat

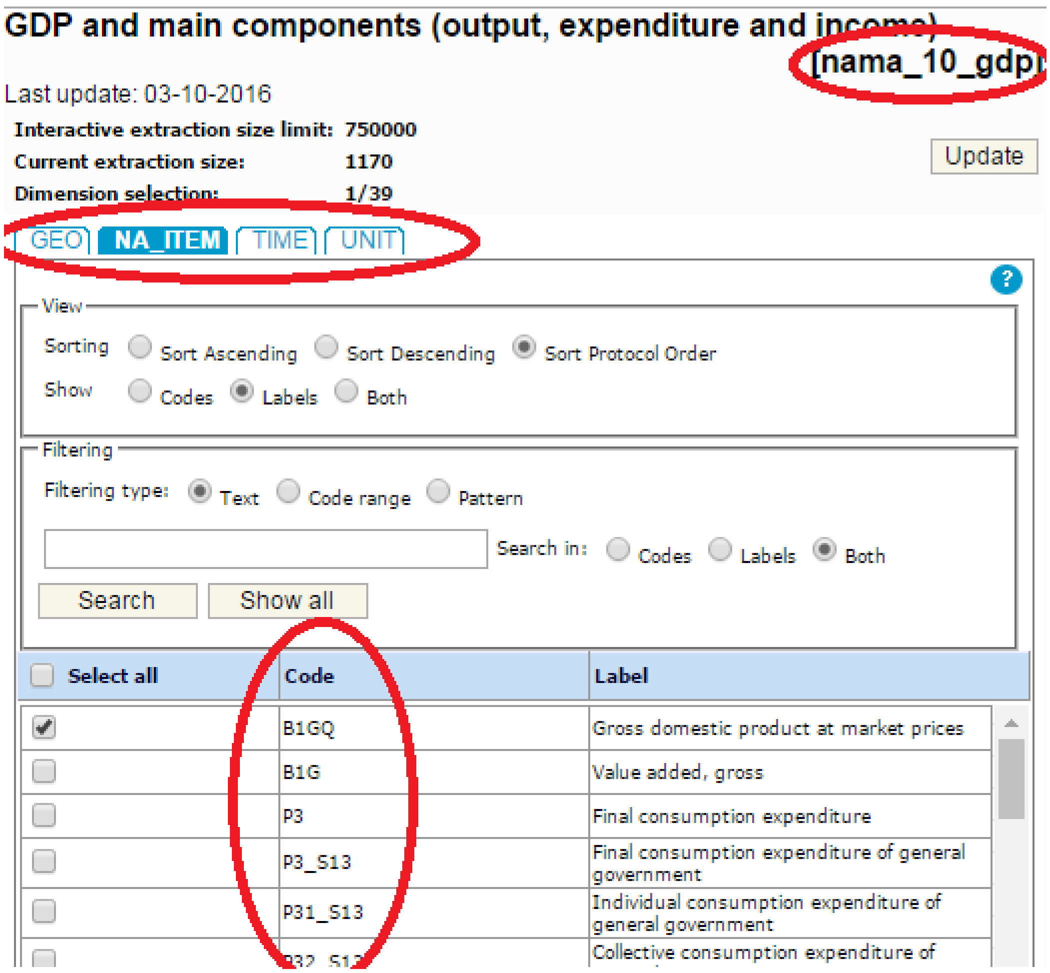


Figure 5: Data Details

Data

Eurostat

Looking at sectoral Accounts

url: http://ec.europa.eu/eurostat/web/sector-accounts

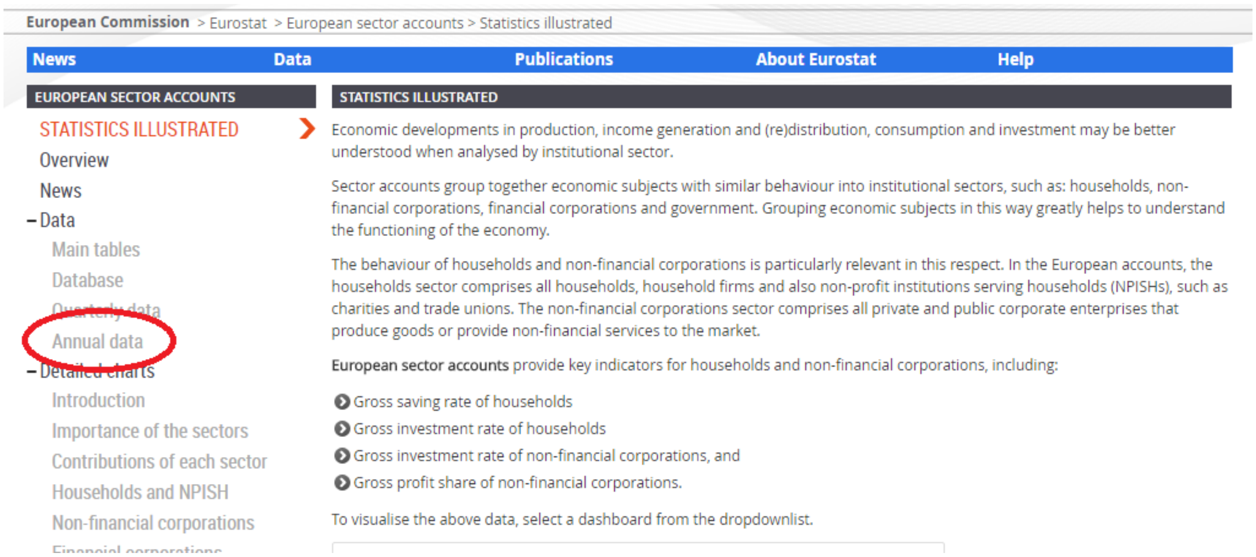


Figure 6: Sectoral Accounts webpage

pdfetch: getting data automatically

- Fetch Economic and Financial Time Series Data from Public Sources
- Package developed by Abiel Reinhart
- We will be using the `pdfetch_EUROSTAT` function

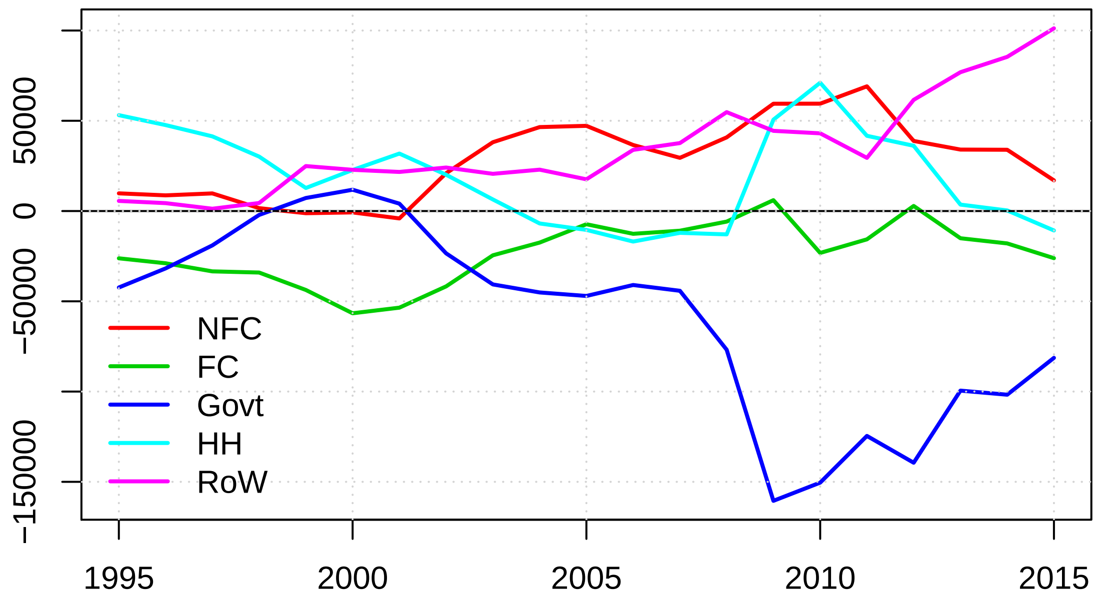
Example 1: Net lending per sector, UK

```
# Specifying the name of the flows of interests
names<-c("B9")
# Downloading the data by specifying the various filters
UKdata_raw = pdfetch_EUROSTAT(flowRef = "nasa_10_nf_tr",
        UNIT="CP_MNAC",NA_ITEM=names, GEO="UK", DIRECT="PAID",
        SECTOR=c("S11","S12","S13","S14_S15","S2"))
# Transforming the obtained data into a data.frame
UKdata<-as.data.frame(UKdata_raw)
# Setting readable names
colnames(UKdata)<-c("NFC","FC","Govt","HH","RoW")

# Matricial plot
matplot(1995:2015,UKdata,lwd=2,type="l",lty=1,
        ylab="",xlab="",col=2:6)
```

Figure 7: Sectoral Accounts Countries Data

```r
# Adding the horizontal line
abline(h=0,col=1)
# Adding a grid
grid()
# Adding a legend
legend("bottomleft",col=c(2:6),lwd=2,lty=1,legend=c("NFC",
        "FC","Govt","HH","RoW"),bty='n')
```



Example 2: Household income statement from 2014

```
# Selecting the flows
names<-c("B5G","D5","D61","D62","D7","D8","B6G","P3","B8G"
            ,"P5G","D9","NP","B9")
# Obtaining the data
EZdata_raw = pdfetch_EUROSTAT("nasa_10_nf_tr",
            UNIT="CP_MNAC",NA_ITEM=names, GEO="EU28",
            SECTOR=c("S14_S15"), TIME="2014")
# Transforming the data into a data.frame
EZdata<-as.data.frame(EZdata_raw)

# Automatic procedure to remove the non-interesting bit
# of the colnames
coln<-colnames(EZdata)
newcoln<-c()
HHdata<-c()
for(i in 1:length(coln)){
  name<-coln[i]
  tname<-strsplit(name,"\\.")[[1]]
  newname<-paste(tname[3:4],collapse=".")
# If the column contains only NA, remove it from dataset
  if(!is.na(EZdata[16,i])){
    newcoln<-c(newcoln,newname)
```

```r
    HHdata<-c(HHdata,EZdata[16,i])
  }
}

# Creating a new dataset with only values 2014
HHdata<-as.data.frame(t(HHdata))
colnames(HHdata)<-newcoln
# Creating the aggregates
HHdata_1<-as.data.frame(c(HHdata$PAID.B5G,-HHdata$PAID.D5,
            -HHdata$PAID.D61+HHdata$RECV.D61,
            +HHdata$RECV.D62-HHdata$PAID.D62,
            -HHdata$PAID.D7+HHdata$RECV.D7,HHdata$PAID.B6G))
colnames(HHdata_1)<-"Households"
rownames(HHdata_1)<-c("Total Income","Taxes"
        ,"Social Contributions","Social Benefits",
        "Other transfers","Gross Disposable Income")
kable(HHdata_1)
```

	Households
Total Income	9851688
Taxes	-1442130
Social Contributions	-2425244
Social Benefits	2635931
Other transfers	109116
Gross Disposable Income	8729361

```r
HHdata_2<-as.data.frame(c(HHdata$PAID.B6G,-HHdata$PAID.P3,
        -HHdata$PAID.D8+HHdata$RECV.D8,HHdata$PAID.B8G))
colnames(HHdata_2)<-"2014"
rownames(HHdata_2)<-c("Gross Disposable Income",
    "Consumption","Adjustments in Pensions","Gross Savings")
kable(HHdata_2)
```

	2014
Gross Disposable Income	8729361
Consumption	-8012603
Adjustments in Pensions	201553
Gross Savings	918310

```r
HHdata_3<-as.data.frame(c(HHdata$PAID.B8G,-HHdata$PAID.P5G
        ,-HHdata$PAID.D9+HHdata$RECV.D9,-HHdata$PAID.NP,
        HHdata$PAID.B9))

colnames(HHdata_3)<-"2014"
rownames(HHdata_3)<-c("Gross Savings",
        "Gross Capital Formation","Capital Transfer",
        "Net Non-Produced NF Assets","Net Lending Position")
kable(HHdata_3)
```

	2014
Gross Savings	918310
Gross Capital Formation	-716849
Capital Transfer	14213
Net Non-Produced NF Assets	6360
Net Lending Position	222034

PKSFC package

- Allowing to simulate SFC models in an open source environment}
- Still preliminary
- Only one numerical solver: Gauss-Seidel algorithm (Kinsella and O'Shea 2010)
- github.com/s120/pksfc
- Technical aspect
- R package
- EViews translator
- Read equation files
- Build model from console
- Visualization/Design tools and helpers

Installing the dependent libraries and the package

You need to install all the required libraries This is for traditional libraries

```
install.packages("expm")
install.packages("igraph")
```

For non-conventional libraries, such as the one need to visualize Direct Acyclical Graphs (DAG), you need to do the following

```
source("http://bioconductor.org/biocLite.R")
biocLite("Rgraphviz")
```

Finally you can then download the PKSFC package from github and install it locally

```
install.packages("path/PKSFC_1.5.tar.gz", repos = NULL,
                                  type="source")
```

Testing

Now we're ready to load the package:

```
library(PKSFC)
```

```
## Loading required package: expm

## Warning: package 'expm' was built under R version 3.2.5

## Loading required package: Matrix

## Warning: package 'Matrix' was built under R version 3.2.5

##
## Attaching package: 'expm'
```

```
## The following object is masked from 'package:Matrix':
##
##     expm
## Loading required package: igraph
## Warning: package 'igraph' was built under R version 3.2.5
##
## Attaching package: 'igraph'
## The following objects are masked from 'package:stats':
##
##     decompose, spectrum
## The following object is masked from 'package:base':
##
##     union
```

1. Load SIM (download SIM.txt from Github)

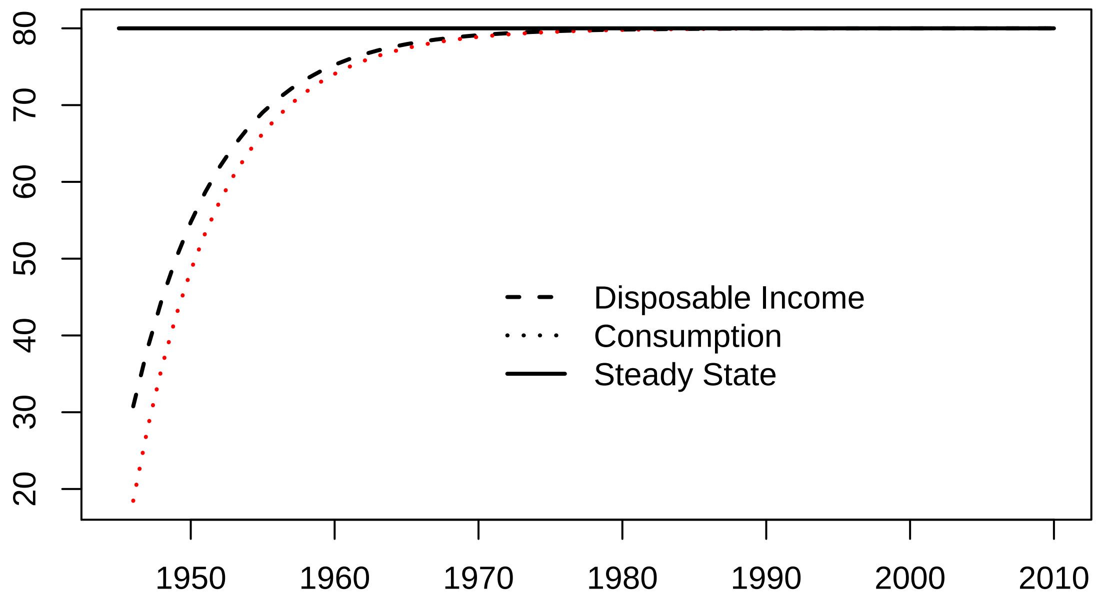
```
sim<-sfc.model("SIM.txt",modelName="SIMplest model
        from chapter 3 of Godley and Lavoie (2007)")
```

2. Simulate the model

```
datasim<-simulate(sim)
```

3. replicate figure 3.2 of page 73.

```
matplot(sim$time,datasim$baseline[,c("Yd","C")],type="l",
    xlab="", ylab="",lwd=2,lty=c(2,3))
lines(sim$time,vector(length=length(sim$time))
    +datasim$baseline["2010","C"], lwd=2)
legend(x=1970,y=50,lwd=2, legend=c("Disposable Income",
    "Consumption","Steady State"),lty=c(2,3,1), bty="n")
```



How does it work?

- The package parses a text file containing the equations
- It generates an internal representation of the model
- It checks the internal consistency of the model, the calibration
- Allows to simulate the model using a linear solver, the Gauss-Seidel Algorithm

Source code of SIM

```
#1. EQUATIONS
C = C
G = G
T = T
N = N
Yd = W*N - T
T = theta*W*N
C = alpha1*Yd + alpha2*H_h(-1)
H = H(-1) + G - T
H_h = H_h(-1) + Yd - C
Y = C + G
N = Y/W
#2. PARAMETERS
alpha1=0.6
```

```
alpha2=0.4
theta=0.2
#EXOGENOUS
G=20
W=1
#INITIAL VALUES
H=0
H_h=0
#3. Timeline
timeline 1945 2010
```

A few important points regarding the model source code:

- The first line of the code should be a comment line (starting with `#`)
- The file should not contain any empty lines.
- You should avoid naming your variables with reserved names in R such as 'in' or 'max'.
- There should be only one equation per line.
- There should be only one variable on the left hand side of the equation.
- You can use R functions such as min, max, or logical operators such as > or <=. In the case the logical operator returns true, the numeric value will be one. Thus `(100>10)` will return `1`.
- The lag operator is represented by `(-x)` where `x` is the lag.
- You can add as many comments, using the `#` character at the beginning of the line. Each comment exactly above an equation will be considered as the description of the equation and will be stored in the internal representation of the sfc model object.

Internal representation

```
print(sim)
```

```
## $name
## [1] "SIMplest model \n\t\tfrom chapter 3 of Godley and Lavoie (2007)"
##
## $simulated
## [1] FALSE
##
## $variables
##        name     initial value description
##  [1,] "Yd"     NA            "1. EQUATIONS"
##  [2,] "T"      NA            ""
##  [3,] "C"      NA            ""
##  [4,] "H"      "0"           ""
##  [5,] "H_h"    "0"           ""
##  [6,] "Y"      NA            ""
##  [7,] "N"      NA            ""
##  [8,] "alpha1" "0.6"         "2. PARAMETERS"
##  [9,] "alpha2" "0.4"         ""
## [10,] "theta"  "0.2"         ""
## [11,] "G"      "20"          "EXOGENOUS"
## [12,] "W"      "1"           ""
##
## $endogenous
```

```
##      name  initial value lag description
## [1,] "Yd"  NA            "0" "1. EQUATIONS"
## [2,] "T"   NA            "0" ""
## [3,] "C"   NA            "0" ""
## [4,] "H"   "0"           "1" ""
## [5,] "H_h" "0"           "1" ""
## [6,] "Y"   NA            "0" ""
## [7,] "N"   NA            "0" ""
##
## $equations
##      endogenous value equation                  description
## [1,] "Yd"             "W*N-T"                   "1. EQUATIONS"
## [2,] "T"              "theta*W*N"               ""
## [3,] "C"              "alpha1*Yd+alpha2*H_h_1" ""
## [4,] "H"              "H_1+G-T"                 ""
## [5,] "H_h"            "H_h_1+Yd-C"              ""
## [6,] "Y"              "C+G"                     ""
## [7,] "N"              "Y/W"                     ""
##
## $time
##  [1] 1945 1946 1947 1948 1949 1950 1951 1952 1953 1954 1955 1956 1957 1958
## [15] 1959 1960 1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972
## [29] 1973 1974 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986
## [43] 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000
## [57] 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010
##
## $matrix
##     Yd T C H H_h Y N
## Yd   0 1 0 0   0 0 1
## T    0 0 0 0   0 0 1
## C    1 0 0 0   0 0 0
## H    0 1 0 0   0 0 0
## H_h  1 0 1 0   0 0 0
## Y    0 0 1 0   0 0 0
## N    0 0 0 0   0 1 0
##
## $blocks
## $blocks[[1]]
## [1] 2 3 4 6 7 1 5
##
##
## attr(,"class")
## [1] "sfc"
```

Output data structure

- Output is a list of matrix where each element of the list are a scenario
  - baseline
  - scenario_i

Y	d	T	C	H H_	h	Y	N alpha1	alpha2	theta	G W	it	er bl
1945	NA	NA	NA	0.00000	0.00000	NA	NA	0.6	0.4	0.2	20	1
1946	30.76923	7.692308	18.46154	12.30769	12.30769	38.46154	38.46154	0.6	0.4	0.2	20	1

Y	d	T	C	H H_	h	Y	N alpha1	alpha2	theta	G W	it	er bl
1947	38.34320	9.585799	27.92899	22.72189	22.72189	47.92899	47.92899	0.6	0.4	0.2	20	1
1948	44.75193	11.187984	35.93992	31.53391	31.53391	55.93992	55.93992	0.6	0.4	0.2	20	1
1949	50.17471	12.543678	42.71839	38.99023	38.99023	62.71839	62.71839	0.6	0.4	0.2	20	1
1950	54.76322	13.690805	48.45402	45.29943	45.29943	68.45402	68.45402	0.6	0.4	0.2	20	1
1951	58.64580	14.661450	53.30725	50.63798	50.63798	73.30725	73.30725	0.6	0.4	0.2	20	1
1952	61.93106	15.482766	57.41383	55.15521	55.15521	77.41383	77.41383	0.6	0.4	0.2	20	1
1953	64.71090	16.177725	60.88862	58.97749	58.97749	80.88862	80.88862	0.6	0.4	0.2	20	1
1954	67.06307	16.765767	63.82884	62.21172	62.21172	83.82884	83.82884	0.6	0.4	0.2	20	1
1955	69.05337	17.263341	66.31671	64.94838	64.94838	86.31671	86.31671	0.6	0.4	0.2	20	1
1956	70.73746	17.684366	68.42183	67.26401	67.26401	88.42183	88.42183	0.6	0.4	0.2	20	1
1957	72.16247	18.040617	70.20309	69.22339	69.22339	90.20309	90.20309	0.6	0.4	0.2	20	1
1958	73.36824	18.342061	71.71030	70.88133	70.88133	91.71030	91.71030	0.6	0.4	0.2	20	1
1959	74.38851	18.597128	72.98564	72.28421	72.28421	92.98564	92.98564	0.6	0.4	0.2	20	1
1960	75.25182	18.812955	74.06477	73.47125	73.47125	94.06477	94.06477	0.6	0.4	0.2	20	1
1961	75.98231	18.995577	74.97789	74.47567	74.47567	94.97789	94.97789	0.6	0.4	0.2	20	1
1962	76.60041	19.150104	75.75052	75.32557	75.32557	95.75052	95.75052	0.6	0.4	0.2	20	1
1963	77.12343	19.280857	76.40428	76.04471	76.04471	96.40428	96.40428	0.6	0.4	0.2	20	1
1964	77.56598	19.391494	76.95747	76.65322	76.65322	96.95747	96.95747	0.6	0.4	0.2	20	1
1965	77.94044	19.485111	77.42555	77.16811	77.16811	97.42555	97.42555	0.6	0.4	0.2	20	1
1966	78.25730	19.564324	77.82162	77.60378	77.60378	97.82162	97.82162	0.6	0.4	0.2	20	1
1967	78.52541	19.631351	78.15676	77.97243	77.97243	98.15676	98.15676	0.6	0.4	0.2	20	1
1968	78.75227	19.688067	78.44033	78.28437	78.28437	98.44033	98.44033	0.6	0.4	0.2	20	1
1969	78.94423	19.736056	78.68028	78.54831	78.54831	98.68028	98.68028	0.6	0.4	0.2	20	1
1970	79.10665	19.776663	78.88332	78.77165	78.77165	98.88332	98.88332	0.6	0.4	0.2	20	1
1971	79.24409	19.811023	79.05511	78.96062	78.96062	99.05511	99.05511	0.6	0.4	0.2	20	1
1972	79.36038	19.840096	79.20048	79.12053	79.12053	99.20048	99.20048	0.6	0.4	0.2	20	1
1973	79.45879	19.864697	79.32348	79.25583	79.25583	99.32348	99.32348	0.6	0.4	0.2	20	1
1974	79.54205	19.885513	79.42756	79.37032	79.37032	99.42756	99.42756	0.6	0.4	0.2	20	1
1975	79.61250	19.903126	79.51563	79.46719	79.46719	99.51563	99.51563	0.6	0.4	0.2	20	1
1976	79.67212	19.918030	79.59015	79.54916	79.54916	99.59015	99.59015	0.6	0.4	0.2	20	1
1977	79.72256	19.930640	79.65320	79.61852	79.61852	99.65320	99.65320	0.6	0.4	0.2	20	1
1978	79.76524	19.941311	79.70656	79.67721	79.67721	99.70656	99.70656	0.6	0.4	0.2	20	1
1979	79.80136	19.950340	79.75170	79.72687	79.72687	99.75170	99.75170	0.6	0.4	0.2	20	1
1980	79.83192	19.957980	79.78990	79.76889	79.76889	99.78990	99.78990	0.6	0.4	0.2	20	1
1981	79.85778	19.964445	79.82222	79.80445	79.80445	99.82222	99.82222	0.6	0.4	0.2	20	1
1982	79.87966	19.969915	79.84957	79.83453	79.83453	99.84957	99.84957	0.6	0.4	0.2	20	1
1983	79.89817	19.974543	79.87272	79.85999	79.85999	99.87272	99.87272	0.6	0.4	0.2	20	1
1984	79.91384	19.978460	79.89230	79.88153	79.88153	99.89230	99.89230	0.6	0.4	0.2	20	1
1985	79.92709	19.981774	79.90887	79.89975	79.89975	99.90887	99.90887	0.6	0.4	0.2	20	1
1986	79.93831	19.984578	79.92289	79.91518	79.91518	99.92289	99.92289	0.6	0.4	0.2	20	1
1987	79.94780	19.986950	79.93475	79.92823	79.92823	99.93475	99.93475	0.6	0.4	0.2	20	1
1988	79.95583	19.988958	79.94479	79.93927	79.93927	99.94479	99.94479	0.6	0.4	0.2	20	1
1989	79.96263	19.990657	79.95328	79.94861	79.94861	99.95328	99.95328	0.6	0.4	0.2	20	1
1990	79.96838	19.992094	79.96047	79.95652	79.95652	99.96047	99.96047	0.6	0.4	0.2	20	1
1991	79.97324	19.993310	79.96655	79.96321	79.96321	99.96655	99.96655	0.6	0.4	0.2	20	1
1992	79.97736	19.994340	79.97170	79.96887	79.96887	99.97170	99.97170	0.6	0.4	0.2	20	1
1993	79.98084	19.995210	79.97605	79.97366	79.97366	99.97605	99.97605	0.6	0.4	0.2	20	1
1994	79.98379	19.995947	79.97974	79.97771	79.97771	99.97974	99.97974	0.6	0.4	0.2	20	1
1995	79.98628	19.996571	79.98285	79.98114	79.98114	99.98285	99.98285	0.6	0.4	0.2	20	1
1996	79.98839	19.997098	79.98549	79.98404	79.98404	99.98549	99.98549	0.6	0.4	0.2	20	1
1997	79.99018	19.997545	79.98772	79.98650	79.98650	99.98772	99.98772	0.6	0.4	0.2	20	1
1998	79.99169	19.997923	79.98961	79.98857	79.98857	99.98961	99.98961	0.6	0.4	0.2	20	1

Y	d	T	C	H H_	h	Y	N alpha1	alpha2	theta	G W	it	er bl
1999	79.99297	19.998242	79.99121	79.99033	79.99033	99.99121	99.99121	0.6	0.4	0.2	20	1
2000	79.99405	19.998513	79.99256	79.99182	79.99182	99.99256	99.99256	0.6	0.4	0.2	20	1
2001	79.99497	19.998741	79.99371	79.99308	79.99308	99.99371	99.99371	0.6	0.4	0.2	20	1
2002	79.99574	19.998935	79.99468	79.99414	79.99414	99.99468	99.99468	0.6	0.4	0.2	20	1
2003	79.99640	19.999099	79.99549	79.99504	79.99504	99.99549	99.99549	0.6	0.4	0.2	20	1
2004	79.99695	19.999237	79.99619	79.99581	79.99581	99.99619	99.99619	0.6	0.4	0.2	20	1
2005	79.99742	19.999355	79.99677	79.99645	79.99645	99.99677	99.99677	0.6	0.4	0.2	20	1
2006	79.99782	19.999454	79.99727	79.99700	79.99700	99.99727	99.99727	0.6	0.4	0.2	20	1
2007	79.99815	19.999538	79.99769	79.99746	79.99746	99.99769	99.99769	0.6	0.4	0.2	20	1
2008	79.99844	19.999609	79.99805	79.99785	79.99785	99.99805	99.99805	0.6	0.4	0.2	20	1
2009	79.99868	19.999669	79.99835	79.99818	79.99818	99.99835	99.99835	0.6	0.4	0.2	20	1
2010	79.99888	19.999720	79.99860	79.99846	79.99846	99.99860	99.99860	0.6	0.4	0.2	20	1

The Gauss Seidel Algorithm

- Principle: Solving $Ax = b$, $A \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$ via an iterative algorithm, where each iteration can be represented by $Lx^{k+1} = b - Ux^k$, $A = L + U$. Where L is lower triangular and U is upper triangular.
- Pseudo-code:

1. Select initial values x^0
2. While $k < maxIter$ & $\delta < tolValue$

a. For each $i = 1, ..., n$:

$$x_i^{k+1} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=i+1}^{n} a_{ij}x_j^k\right)$$

b. Compute δ:

$$\delta = \frac{x^{k+1} - x^k}{x^k}$$

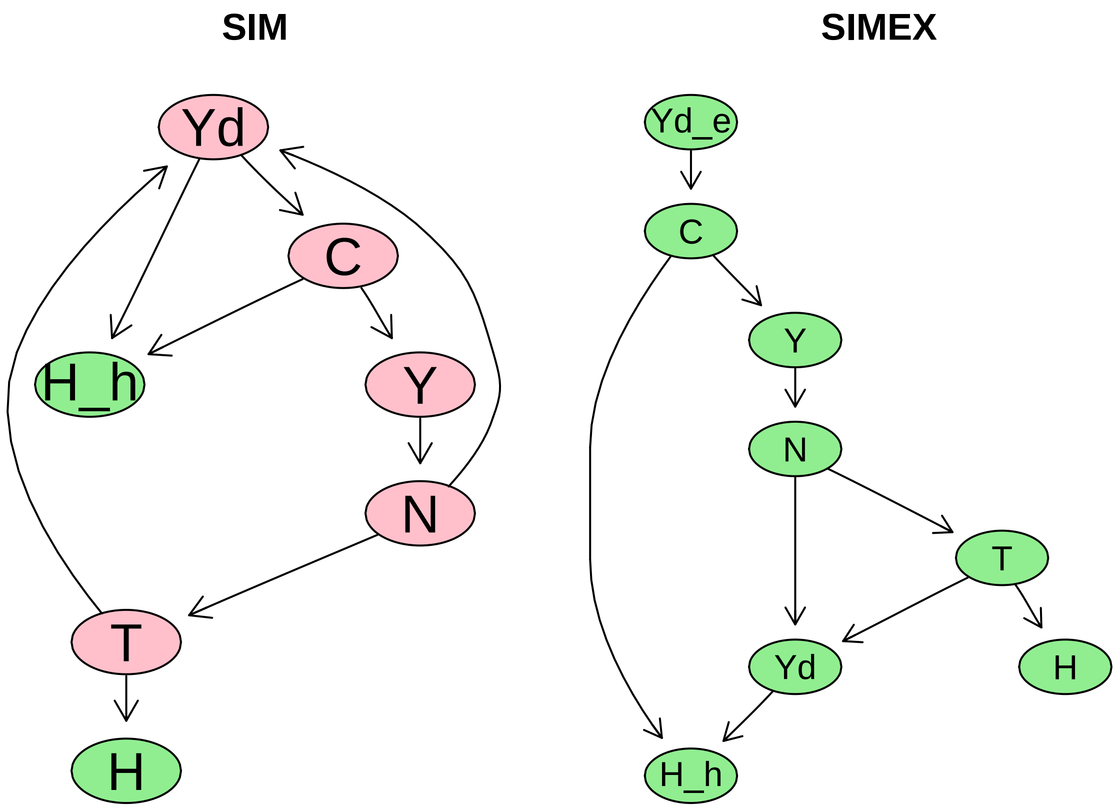
System of (in)dependent equations

See Fenell et. al (2016)

- The Gauss-Seidel has to be used only in the case of system of dependent equations
- In other case, we only need to find order in which each variable is computed and simply compute the new value in each period
- This order is the “logical causal order” of the model and can be visualized using Direct (A)Cyclical Graphs

Direct Acyclic graphs

```
simex<-sfc.model("SIMEX.txt",modelName="SIMplest model with expectation")
layout(matrix(c(1,2),1,2))
plot.dag(sim,main="SIM" )
plot.dag(simex,main="SIMEX" )
```

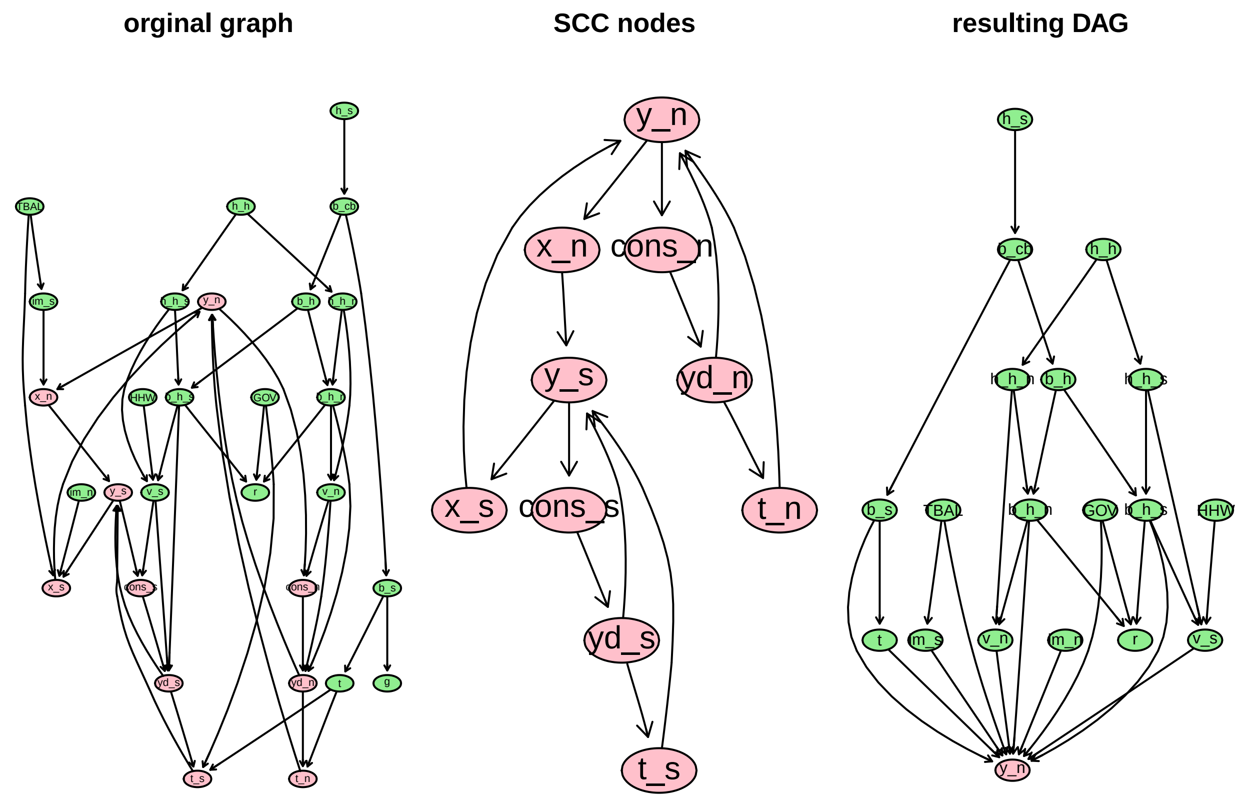


- Aside from the mathematical implication that a system of equation represent, it also has an economic meaning:
- the economy represented by SIM will adjust in one period to any shock applied to government spending.
- for SIMEX it is not the case because consumption depend on expected disposable income which is equal to previous period disposable income.
- in this case, the economy represented by SIMEX will adjust slowly to a shock applied to government spending, via the stocks (and particularly the buffer stock)

In the case of a more complex model - Chapter 6

- We can generate the plots that allow us to delve into the actual structure of the system.
- Nodes that do not form a cycle are green while nodes that form a cycle in the system are pink.
- The Gauss-Seidel needs to be applied only for the cycles

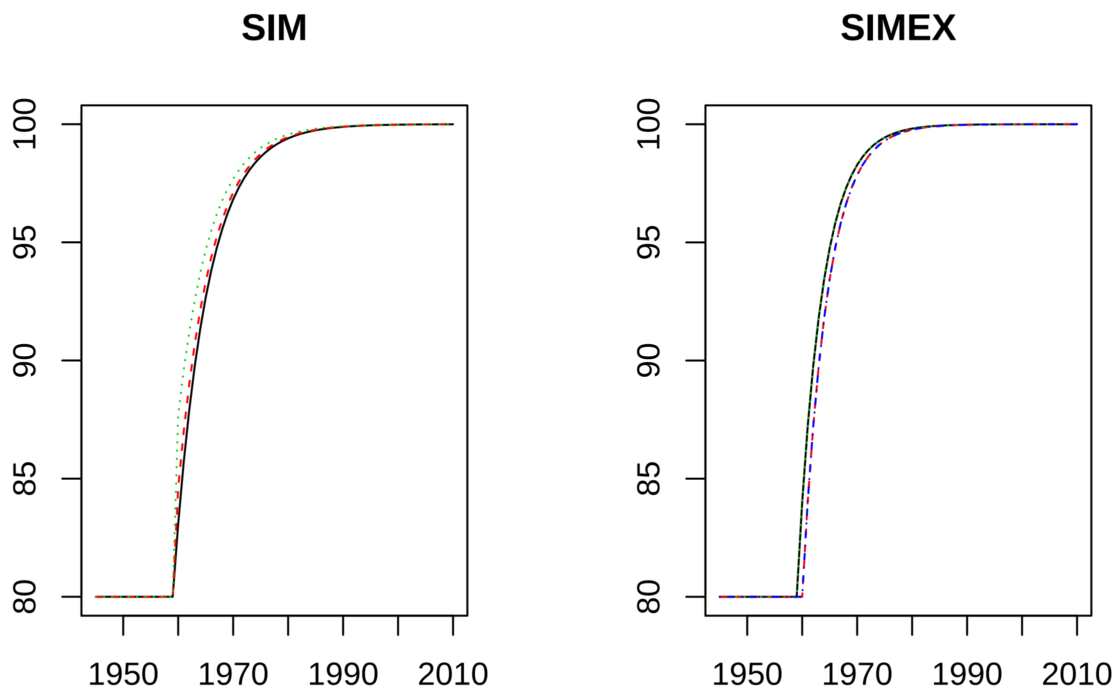
```
ch6 <- sfc.model("ch6.txt",modelName="Chapter6_openmodel")
graphs = generate.DAG.collapse( adjacency = ch6$matrix )
par(mfrow = c(1,3))
# first plot the orgianl grpah
plot_graph_hierarchy( graphs$orginal_graph, main = "orginal graph" )
# plot hte nodes that form the strongly connected compoent
plot_graph_hierarchy( graphs$SCC_graph, main = "SCC nodes" )
# plot the result ing DAG when we take the condensation of the graph
plot_graph_hierarchy( graphs$DAG, main = "resulting DAG" )
```



Systems of dependent vs independent equations

```
#Doing the simulations
datasimex<-simulate(simex)
init = datasimex$baseline[66,]
simex<-sfc.addScenario(simex,"G",25,1960,2010,init)
datasimex<-simulate(simex)
datasim<-simulate(sim)
init = datasim$baseline[66,]
sim<-sfc.addScenario(sim,"G",25,1960,2010,init)
datasim<-simulate(sim)

#Plotting it all
layout(matrix(c(1,2),1,2))
matplot(sim$time,datasim$scenario_1[,c("H","C","Yd")],
    lty=1:3,type="l",xlab="",ylab="",main="SIM")
legend(x=1944,y=130,legend=c("Wealth","Consumption",
    "Disposable Income"), lty=1:3,bty="n")
matplot(simex$time,datasimex$scenario_1[,c("H","C","Yd",
    "Yd_e")],lty=1:4,type="l",xlab="",ylab="",main="SIMEX")
legend(x=1944,y=130,legend=c("Wealth","Consumption","Disposable Income",
   "Expecetd Disposable Income"),lty=1:4,bty="n")
```



Computational implications

Let's see how much time it takes to run sim:

```
ptm <- proc.time()
data1<-simulate(sim)
print(paste("Elapsed time is ",proc.time()[3]-ptm[3],
                        "seconds"))
```

```
## [1] "Elapsed time is  3.76 seconds"
```

Now lets play with some of the parameters of the simulate function:

1. tolValue

```
ptm <- proc.time()
data2<-simulate(sim,tolValue = 1e-3)
print(paste("Elapsed time is ",proc.time()[3]-ptm[3],"seconds"))
```

```
## [1] "Elapsed time is  0.189999999999998 seconds"
```

2. maxIter

```
ptm <- proc.time()
data3<-simulate(sim, maxIter=10)
print(paste("Elapsed time is ",proc.time()[3]-ptm[3],"seconds"))
```

```
## [1] "Elapsed time is  0.18 seconds"
```

Observing the results of the three simulations

```
kable(round(t(cbind(data1$baseline[c(1,2,20,40,66),c("Y")],
        data2$baseline[c(1,2,20,40,66),c("Y")],
        data3$baseline[c(1,2,20,40,66),c("Y")])),digits=3))
```

1945	1946	1964	1984	2010
NA	38.462	96.957	99.892	99.999
NA	38.464	96.874	99.577	99.911
NA	34.006	94.965	99.672	99.991

Block Gauss-Seidel

The order of equations matters, if first compute variables that do not depend on current period, this speeds the process. Define blocks of equation independent from the others.

```
print(simex$blocks)
```

```
## [[1]]
## [1] 7
##
## [[2]]
## [1] 3
##
## [[3]]
## [1] 6
##
## [[4]]
## [1] 8
##
## [[5]]
## [1] 2
##
## [[6]]
## [1] 1 4
##
## [[7]]
## [1] 5
```

Simulation of SIMEX

```
ptm <- proc.time()
dataex<-simulate(simex,tolValue = 1e-10)
print(paste("Elapsed time is ",proc.time()[3]-ptm[3]
                        ,"seconds"))
```

```
## [1] "Elapsed time is  0.110000000000003 seconds"
```

Results for SIMEX

```
kable(round(t(dataex$baseline[c(1,2,20,40,66),c("G","Y",
    "T","Yd","Yd_e","C","H","H_h")]),digits=3))
```

	1945	1946	1964	1984	2010
G	20	20	20.000	20.000	20
Y	NA	20	98.559	99.983	100
T	NA	4	19.712	19.997	20
Yd	0	16	78.847	79.987	80
Yd_e	NA	0	78.559	79.983	80
C	NA	0	78.559	79.983	80
H	0	16	78.847	79.987	80
H_h	0	16	78.847	79.987	80

Output data structure

- Output is a list of matrix where each element of the list are a scenario
- baseline
- scenario_i
- In the result matrix, there is a column indicating the number of iteration in the Gauss-Seidel algorithm per block of equations per period

```
kable(dataex$baseline)
```

	Yd	T	C	H	H_h	Y	Yd_e	N	alpha1	alpha2	theta
1945	0.00000	NA	NA	0.00000	0.00000	NA	NA	NA	0.6	0.4	0.2
1946	16.00000	4.00000	0.00000	16.00000	16.00000	20.00000	0.00000	20.00000	0.6	0.4	0.2
1947	28.80000	7.20000	16.00000	28.80000	28.80000	36.00000	16.00000	36.00000	0.6	0.4	0.2
1948	39.04000	9.76000	28.80000	39.04000	39.04000	48.80000	28.80000	48.80000	0.6	0.4	0.2
1949	47.23200	11.80800	39.04000	47.23200	47.23200	59.04000	39.04000	59.04000	0.6	0.4	0.2
1950	53.78560	13.44640	47.23200	53.78560	53.78560	67.23200	47.23200	67.23200	0.6	0.4	0.2
1951	59.02848	14.75712	53.78560	59.02848	59.02848	73.78560	53.78560	73.78560	0.6	0.4	0.2
1952	63.22278	15.80570	59.02848	63.22278	63.22278	79.02848	59.02848	79.02848	0.6	0.4	0.2
1953	66.57823	16.64456	63.22278	66.57823	66.57823	83.22278	63.22278	83.22278	0.6	0.4	0.2
1954	69.26258	17.31565	66.57823	69.26258	69.26258	86.57823	66.57823	86.57823	0.6	0.4	0.2
1955	71.41007	17.85252	69.26258	71.41007	71.41007	89.26258	69.26258	89.26258	0.6	0.4	0.2
1956	73.12805	18.28201	71.41007	73.12805	73.12805	91.41007	71.41007	91.41007	0.6	0.4	0.2
1957	74.50244	18.62561	73.12805	74.50244	74.50244	93.12805	73.12805	93.12805	0.6	0.4	0.2
1958	75.60195	18.90049	74.50244	75.60195	75.60195	94.50244	74.50244	94.50244	0.6	0.4	0.2
1959	76.48156	19.12039	75.60195	76.48156	76.48156	95.60195	75.60195	95.60195	0.6	0.4	0.2
1960	77.18525	19.29631	76.48156	77.18525	77.18525	96.48156	76.48156	96.48156	0.6	0.4	0.2
1961	77.74820	19.43705	77.18525	77.74820	77.74820	97.18525	77.18525	97.18525	0.6	0.4	0.2
1962	78.19856	19.54964	77.74820	78.19856	78.19856	97.74820	77.74820	97.74820	0.6	0.4	0.2
1963	78.55885	19.63971	78.19856	78.55885	78.55885	98.19856	78.19856	98.19856	0.6	0.4	0.2
1964	78.84708	19.71177	78.55885	78.84708	78.84708	98.55885	78.55885	98.55885	0.6	0.4	0.2
1965	79.07766	19.76942	78.84708	79.07766	79.07766	98.84708	78.84708	98.84708	0.6	0.4	0.2
1966	79.26213	19.81553	79.07766	79.26213	79.26213	99.07766	79.07766	99.07766	0.6	0.4	0.2
1967	79.40970	19.85243	79.26213	79.40970	79.40970	99.26213	79.26213	99.26213	0.6	0.4	0.2
1968	79.52776	19.88194	79.40970	79.52776	79.52776	99.40970	79.40970	99.40970	0.6	0.4	0.2
1969	79.62221	19.90555	79.52776	79.62221	79.62221	99.52776	79.52776	99.52776	0.6	0.4	0.2

	Yd	T	C	H	H_h	Y	Yd_e	N	alpha1	alpha2	theta
1970	79.69777	19.92444	79.62221	79.69777	79.69777	99.62221	79.62221	99.62221	0.6	0.4	0.2
1971	79.75821	19.93955	79.69777	79.75821	79.75821	99.69777	79.69777	99.69777	0.6	0.4	0.2
1972	79.80657	19.95164	79.75821	79.80657	79.80657	99.75821	79.75821	99.75821	0.6	0.4	0.2
1973	79.84526	19.96131	79.80657	79.84526	79.84526	99.80657	79.80657	99.80657	0.6	0.4	0.2
1974	79.87621	19.96905	79.84526	79.87621	79.87621	99.84526	79.84526	99.84526	0.6	0.4	0.2
1975	79.90096	19.97524	79.87621	79.90096	79.90096	99.87621	79.87621	99.87621	0.6	0.4	0.2
1976	79.92077	19.98019	79.90096	79.92077	79.92077	99.90096	79.90096	99.90096	0.6	0.4	0.2
1977	79.93662	19.98415	79.92077	79.93662	79.93662	99.92077	79.92077	99.92077	0.6	0.4	0.2
1978	79.94929	19.98732	79.93662	79.94929	79.94929	99.93662	79.93662	99.93662	0.6	0.4	0.2
1979	79.95944	19.98986	79.94929	79.95944	79.95944	99.94929	79.94929	99.94929	0.6	0.4	0.2
1980	79.96755	19.99189	79.95944	79.96755	79.96755	99.95944	79.95944	99.95944	0.6	0.4	0.2
1981	79.97404	19.99351	79.96755	79.97404	79.97404	99.96755	79.96755	99.96755	0.6	0.4	0.2
1982	79.97923	19.99481	79.97404	79.97923	79.97923	99.97404	79.97404	99.97404	0.6	0.4	0.2
1983	79.98338	19.99585	79.97923	79.98338	79.98338	99.97923	79.97923	99.97923	0.6	0.4	0.2
1984	79.98671	19.99668	79.98338	79.98671	79.98671	99.98338	79.98338	99.98338	0.6	0.4	0.2
1985	79.98937	19.99734	79.98671	79.98937	79.98937	99.98671	79.98671	99.98671	0.6	0.4	0.2
1986	79.99149	19.99787	79.98937	79.99149	79.99149	99.98937	79.98937	99.98937	0.6	0.4	0.2
1987	79.99319	19.99830	79.99149	79.99319	79.99319	99.99149	79.99149	99.99149	0.6	0.4	0.2
1988	79.99456	19.99864	79.99319	79.99456	79.99456	99.99319	79.99319	99.99319	0.6	0.4	0.2
1989	79.99564	19.99891	79.99456	79.99564	79.99564	99.99456	79.99456	99.99456	0.6	0.4	0.2
1990	79.99652	19.99913	79.99564	79.99652	79.99652	99.99564	79.99564	99.99564	0.6	0.4	0.2
1991	79.99721	19.99930	79.99652	79.99721	79.99721	99.99652	79.99652	99.99652	0.6	0.4	0.2
1992	79.99777	19.99944	79.99721	79.99777	79.99777	99.99721	79.99721	99.99721	0.6	0.4	0.2
1993	79.99822	19.99955	79.99777	79.99822	79.99822	99.99777	79.99777	99.99777	0.6	0.4	0.2
1994	79.99857	19.99964	79.99822	79.99857	79.99857	99.99822	79.99822	99.99822	0.6	0.4	0.2
1995	79.99886	19.99971	79.99857	79.99886	79.99886	99.99857	79.99857	99.99857	0.6	0.4	0.2
1996	79.99909	19.99977	79.99886	79.99909	79.99909	99.99886	79.99886	99.99886	0.6	0.4	0.2
1997	79.99927	19.99982	79.99909	79.99927	79.99927	99.99909	79.99909	99.99909	0.6	0.4	0.2
1998	79.99942	19.99985	79.99927	79.99942	79.99942	99.99927	79.99927	99.99927	0.6	0.4	0.2
1999	79.99953	19.99988	79.99942	79.99953	79.99953	99.99942	79.99942	99.99942	0.6	0.4	0.2
2000	79.99963	19.99991	79.99953	79.99963	79.99963	99.99953	79.99953	99.99953	0.6	0.4	0.2
2001	79.99970	19.99993	79.99963	79.99970	79.99970	99.99963	79.99963	99.99963	0.6	0.4	0.2
2002	79.99976	19.99994	79.99970	79.99976	79.99976	99.99970	79.99970	99.99970	0.6	0.4	0.2
2003	79.99981	19.99995	79.99976	79.99981	79.99981	99.99976	79.99976	99.99976	0.6	0.4	0.2
2004	79.99985	19.99996	79.99981	79.99985	79.99985	99.99981	79.99981	99.99981	0.6	0.4	0.2
2005	79.99988	19.99997	79.99985	79.99988	79.99988	99.99985	79.99985	99.99985	0.6	0.4	0.2
2006	79.99990	19.99998	79.99988	79.99990	79.99990	99.99988	79.99988	99.99988	0.6	0.4	0.2
2007	79.99992	19.99998	79.99990	79.99992	79.99992	99.99990	79.99990	99.99990	0.6	0.4	0.2
2008	79.99994	19.99998	79.99992	79.99994	79.99994	99.99992	79.99992	99.99992	0.6	0.4	0.2
2009	79.99995	19.99999	79.99994	79.99995	79.99995	99.99994	79.99994	99.99994	0.6	0.4	0.2
2010	79.99996	19.99999	79.99995	79.99996	79.99996	99.99995	79.99995	99.99995	0.6	0.4	0.2

Checking the number of iteractions for SIM

```
kable(round(t(cbind(
    data1$baseline[c(1,2,20,40,66),c("iter block 1")],
    data2$baseline[c(1,2,20,40,66),c("iter block 1")],
    data3$baseline[c(1,2,20,40,66),c("iter block 1")]
    )),digits=3))
```

1945	1946	1964	1984	2010
0	293	218	175	119
0	89	15	2	1
0	10	10	10	10

Checking the number of iteractions for simex, no simulate parameters

```
kable(round(t(dataex$baseline[c(1,2,20,40,66),
    c("iter block 1","iter block 2","iter block 3",
    "iter block 4","iter block 5","iter block 6",
    "iter block 7")]),digits=3))
```

	1945	1946	1964	1984	2010
iter block 1	0	2	2	2	2
iter block 2	0	2	2	2	2
iter block 3	0	2	2	2	2
iter block 4	0	2	2	2	2
iter block 5	0	2	2	2	2
iter block 6	0	2	2	2	2
iter block 7	0	2	2	2	2

Financial Imbalances

Godley's seven unsustainable processes Wynne Godley (1999)

1. Written in 1999 when everything was fine for the US economy. Clinton: "There are no limits to the world we can create, together, in the century to come."
2. Calling for political intervention and expansionary fiscal policies: "*The view taken here, which is built into the Keynesian model later deployed, is that the government's fiscal operations, through their impact on disposable income and expenditure, play a crucial role in determining the level and growth rate of total demand and output.*"
3. Highlighting seven unsustainable processes: "*(1) the fall in private saving into ever deeper negative territory, (2) the rise in the flow of net lending to the private sector, (3) the rise in the growth rate of the real money stock, (4) the rise in asset prices at a rate that far exceeds the growth of profits (or of GDP), (5) the rise in the budget surplus, (6) the rise in the current account deficit, (7) the increase in the United States's net foreign indebtedness relative to GDP.*"

Stock-flow norms

Turnovsky (1977) p.3 and W. Godley and Lavoie (2007), p.13

There are *intrinsic dynamics*, that reflect 'the dynamic behavior stemming from certain logical relationships which constrain the system; specifically the relationships between stocks and flows'

- *Standards*: (private or public) debt to GDP, capacity utilization, unemployment rate, etc...
- *Adjusted fiscal ratio*: $\theta = \frac{T}{Y}$ is the average tax rate, then the fiscal ratio $\frac{G}{\theta}$ is equal to Y when $G = T$, adjusted for inflation.
- *Adjusted Trade Ratio*: $\mu = \frac{M}{Y}$ is the average propensity to import, then the trade ratio $\frac{X}{\mu}$ is equal to GDP when $X = M$, adjusted for inflation.

- *Combined Fiscal and Trade Ratio*: $\frac{G+X}{\theta+\mu}$ is equal to GDP when balanced budget and trade balance.

Unsustainable processes Wynne Godley (1999)

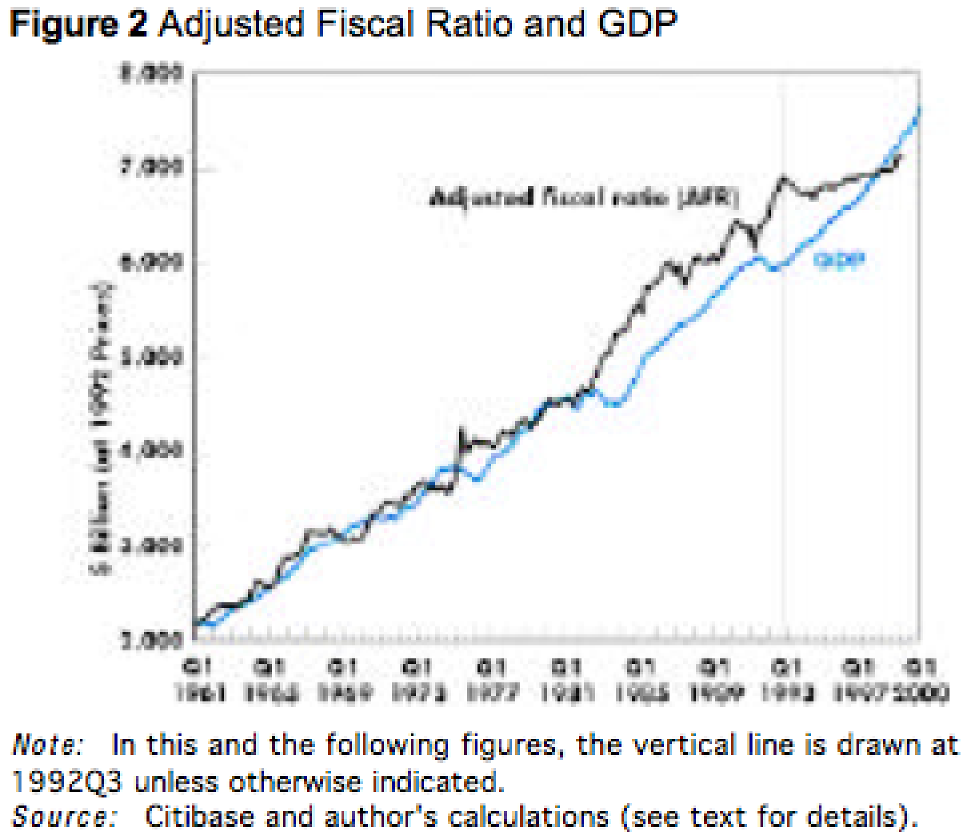


Figure 8: Adjusted Fiscal Ratio and GDP, source: Wynne Godley (1999)

- "Given unchanged fiscal policy and accepting the consensus forecast for growth in the rest of the world, continued expansion of the U.S. economy requires that private expenditure continues to rise relative to income. [...] The growth in net lending to the private sector and the growth in the growth rate of the real money supply cannot continue for an extended period." (p. 5)
- "It will become necessary both to relax the fiscal stance and to increase exports relative to imports [but] it will be difficult to get the timing right." (p. 9)
- He then simulates "whatever fiscal expansion plus (effective) dollar devaluation is necessary to generate the growth of output assumed in the CBO projections (growth just enough to keep unemployment close to its present low level) and an improving balance of payments. Specifically, it was necessary to raise total general government outlays [...] in stages by about 16 percent - corresponding to about $400 billion per annum at current prices - compared with what the CBO is at present projecting." (p. 10)

Model with Portfolio choice

Balance Sheet

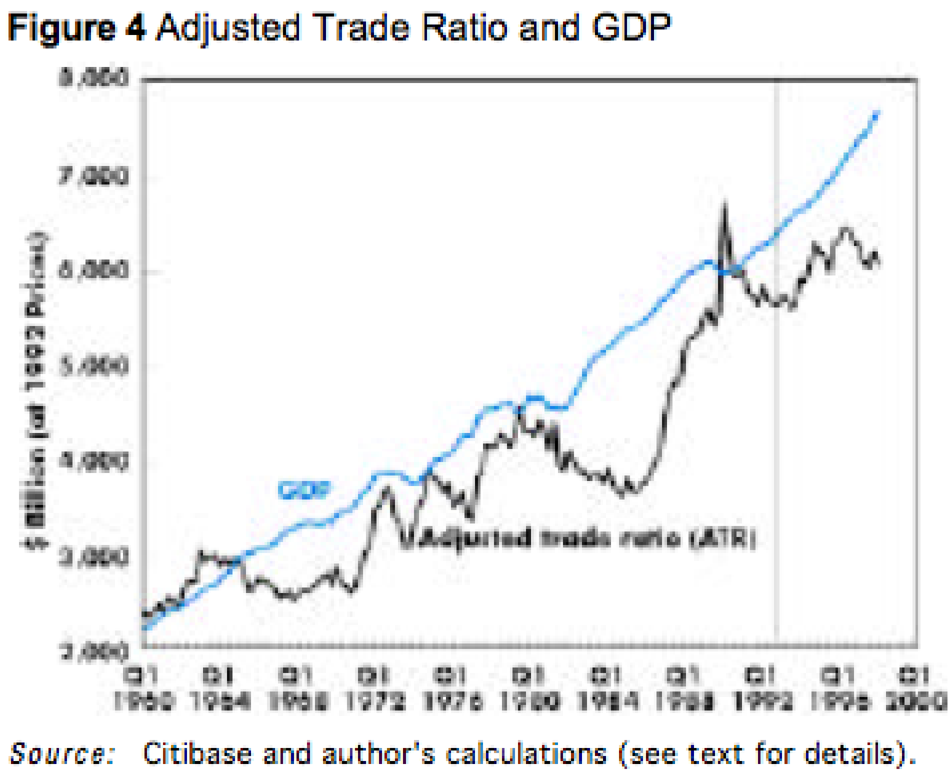


Figure 9: Adjusted Trade Ratio and GDP, source: Wynne Godley (1999)

	Households	Production	Government	Central Bank	Sum
Money	+H			-H	0
Bills	+Bh		-B	+Bcb	0
Net worth	-V		+V		0
Sum	0	0	0	0	0

Transaction Flow Matrix

	Households	Production	Government	Central Bank	Sum
Consumption	-C	+C			0
Gov. Exp.		+G	-G		0
Income = GDP	+Y	-Y			0
Interests	+r(-1)*Bh(-1)		-r(1)*B(-1)	+r(-1)*Bcb(-1)	0
CB profits			+r(-1)*Bcb(-1)	-r(1)*Bcb(-1)	0
Taxes	-T		+T		0
Change in Money	-Δ H			+Δ H	0
Change in Bills	-Δ Bh		+Δ B	-Δ Bcb	0
Sum	0	0	0	0	0

Behavioural Equations (from PC.txt file)

```
#######Determination of output - eq. 4.1
y = cons + g
#######Disposable income - eq. 4.2
yd = y - t + r(-1)*b_h(-1)
#######Tax payments - eq. 4.3
t = theta*(y + r(-1)*b_h(-1))
#######Wealth accumulation - eq. 4.4
```

2. Liquidity preferences: money balances some proportion of total wealth

Merging the two into Tobin's portfolio equations:

$$\frac{H_h}{V} = (1 - \lambda_0) - \lambda_1 \cdot r + \lambda_2 \cdot \left(\frac{YD}{V}\right) \tag{4.6A}$$

$$\frac{B_h}{V} = \lambda_0 + \lambda_1 \cdot r - \lambda_2 \cdot \left(\frac{YD}{V}\right) \tag{4.7}$$

$$H_h = V - B_h \tag{4.6}$$

Setting up the environment

Before doing any modelling, we need to load the package in the R environment.

```
library(PKSFC)
```

Then, you need to download the attached 'PC.txt' file and save it in the folder of your choice. Make sure to set the working directory where you saved the downloaded file. In command line this looks like this but if you use Rstudio, you can use the graphical interface as well (Session>Set Working Directory>Choose Directory)

```
setwd("pathToYourDirectory")
```

Loading the model

The first thing to do is to load the model and check for completeness.

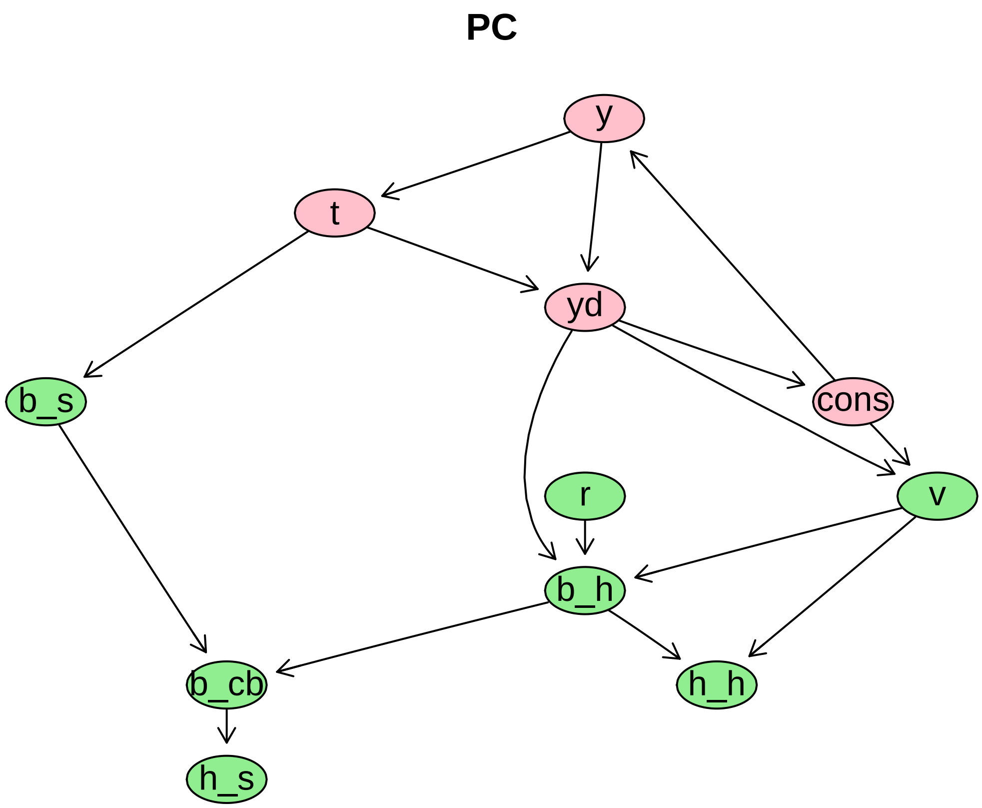
```
pc<-sfc.model("PC.txt",modelName="Portfolio Choice Model")
pc<-sfc.check(pc,fill=FALSE)
```

Let's have a look at the graphical representation of model PC. In order to do so, we need to load the `Rgraphviz` library:

```
library("Rgraphviz")
```

We can now look at the graph of model PC:

```
plot.dag(pc,main="PC" )
```



You can see that there is a cycle in the graph, implying that GDP, taxes, disposable income and consumption are determined all together (and that they fully adapt to any shock applied to the economy). While this is a mathematical property of the system of equations representing the economy we wish to model, it has an economic meaning and you want to be sure that this is what you believe to be the best representation of what you have in mind.

We are now ready to simulate the model

```
datapc<-simulate(pc)
```

Expectations and random shocks

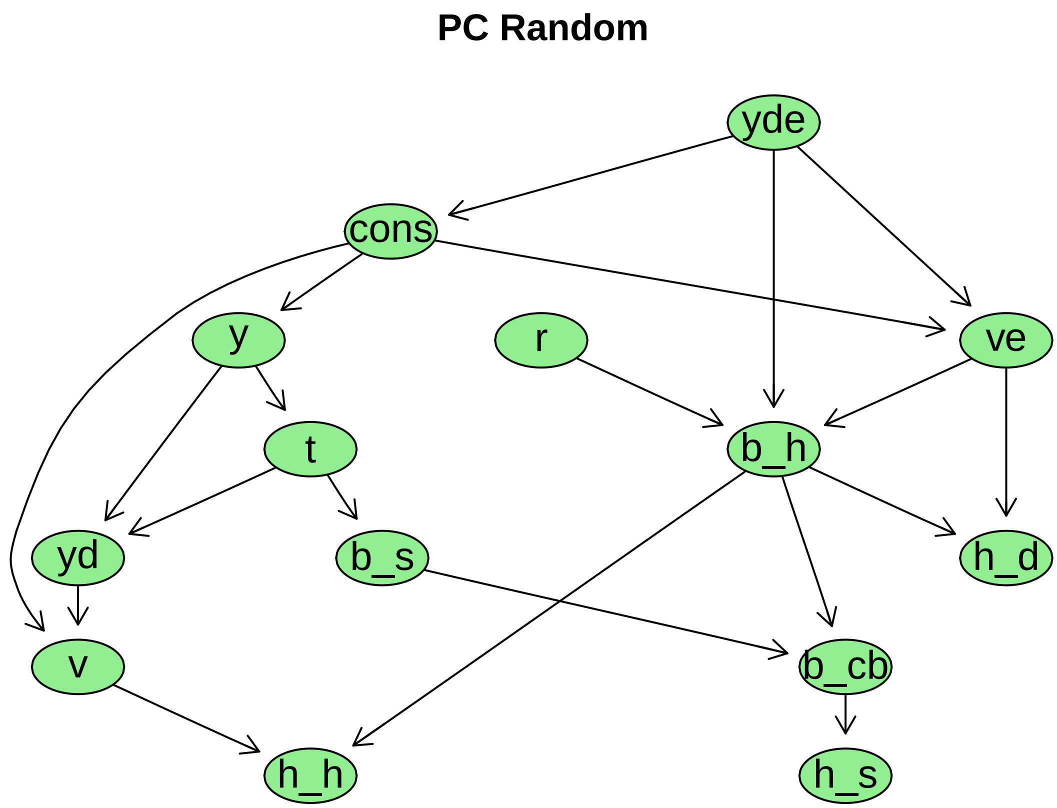
The first of experiment is meant to observe the buffer role of money, in case of random shocks applied to expected disposable income. To do these, we need to modify slightly model pc and change the equations determining consumption, demand for bonds and money, and the expectations on income and wealth. We will call the new model pcRand.

```
pcRand<-sfc.editEqus(pc,list(list(var="cons",eq="alpha1*yde+alpha2*v(-1)"),
      list(var="b_h",eq="ve*(lambda0 + lambda1*r - lambda2*(yde/ve))")))
pcRand<-sfc.addEqus(pcRand,list(
  list(var="yde",eq="yd(-1)*(1+rnorm(1,sd=0.1))",
         desc="Expected disposable income depending on random shocks"),
  list(var="h_d",eq="ve-b_h",desc="Money demand"),
  list(var="ve",eq="v(-1)+yde-cons",desc="Expected disposable income")))
pcRand<-sfc.editVar(pcRand,var="yd",init=86.48648)
pcRand<-sfc.check(pcRand)
```

You probably have noticed that the yde equation contains the R function rnorm which allows you to extract a random number from a normal distribution centered around 0, with standard variation of 0.1. The package indeed allows you to use any R function such as min, max or rnorm. However, because of the way the Gauss-Seidel algorithm works, we are facing a problem. Indeed, as the rnorm function extract a number in each iteration, this will mean that the algorithm cannot converge since the difference between each iteration of the algorithm depends of the difference between each extraction. This is why we need to restrict the number of iteration of the Gauss-Seidel.

Before simulating the model, let's have a look at how the graph of the model has changed:

```
plot.dag(pcRand,main="PC Random" )
```

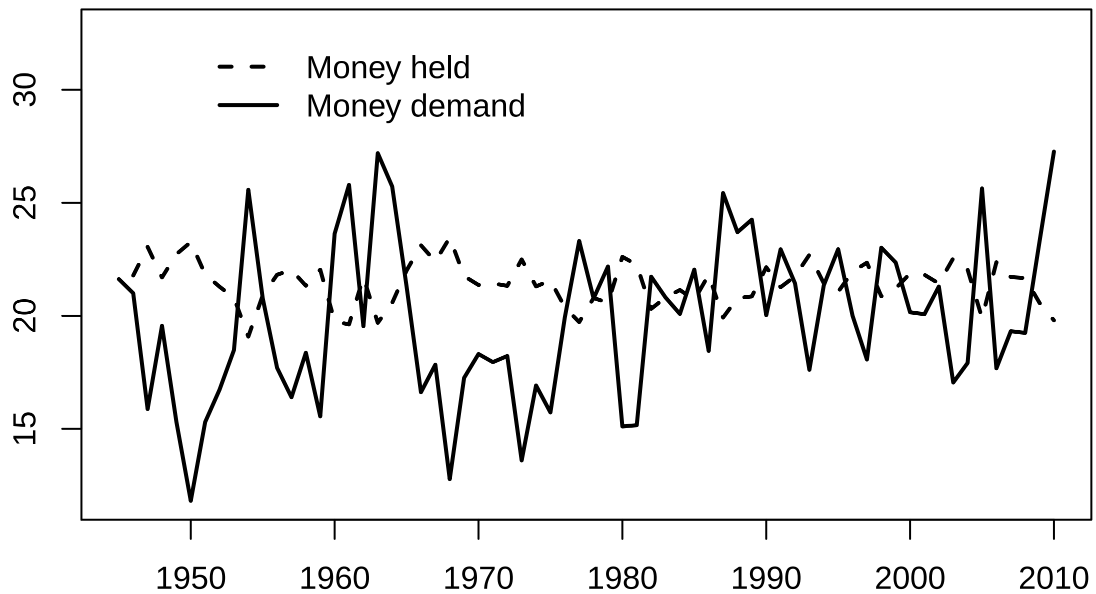


You can now see that the cycle observed in the original model has disappeared. Indeed, consumption doesn't depend on disposable income anymore but on expected disposable income. Let's now simulate the model.

```
datapcRand<-simulate(pcRand,maxIter=2)
```

This replicates figure 4.1, page 110

```
plot(pcRand$time,datapcRand$baseline[,"h_h"],type="l",xlab="",ylab="",lty=1,lwd=2,
     ylim=c(1*min(datapcRand$baseline[,c("h_h","h_d")],na.rm=T),
            1.2*max(datapcRand$baseline[,c("h_h","h_d")],na.rm=T)))
lines(pcRand$time,datapcRand$baseline[,"h_d"],lty=2,lwd=2)
legend(x=1950,y=1.2*max(datapcRand$baseline[,c("h_h","h_d")],na.rm=T),
       legend=c("Money held","Money demand"),lty=c(2,1),lwd=2,bty="n")
```



This graph highlights the buffer role of certain stocks in PK-SFC models. Indeed, because expectations are incorrect or because demand might not be equal to supply in any market, at least one stock will not be equal to the targeted level. As highlighted by Foley (1975), in a model without perfect foresight you need a buffer stock in order to obtain equilibrium between demand and supply. The role of buffer stocks in PK-SFC model is thus fundamental and is at the hart of the approach used by Wynne Godley (1999) in his seven unsustainable processes. It is by observing the dynamics of certain stock-flow norms that you are able to observe the unsustainable processes evolving in an economy, because stocks precisely absorb disequilibrium.

Interest rates impacts

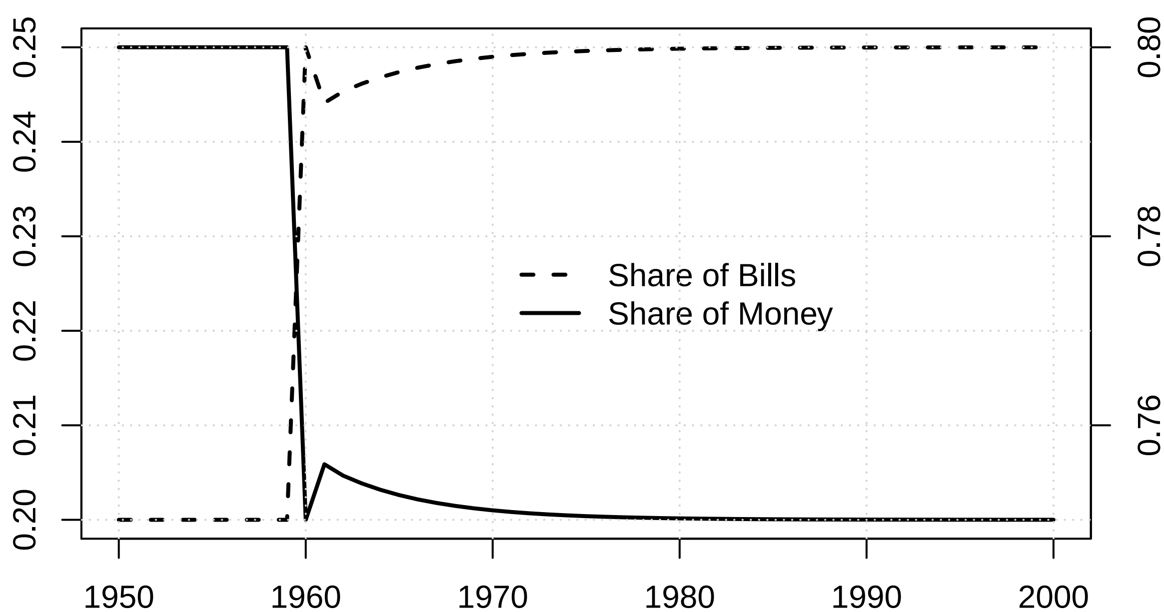
The graphs presented on the paper are based on the model PCEX, which includes expectation on disposable income and wealth. We will first create the model before simulating it.The shock represents an 100 basis point increase in interest rates.

```
pcex<-sfc.editEqus(pc,list(
  list(var="cons",eq="alpha1*yde+alpha2*v(-1)"),
  list(var="b_h",eq="ve*(lambda0 + lambda1*r - lambda2*(yde/ve))")))
pcex<-sfc.addEqus(pcex,list(
  list(var="yde",eq="yd(-1)",desc="Expected disposable income"),
  list(var="h_d",eq="ve-b_h",desc="Money demand"),
  list(var="ve",eq="v(-1)+yde-cons",desc="Expected disposable income")))
pcex<-sfc.editVar(pcex,var="yd",init=86.48648)
pcex<-sfc.check(pcex)
init = datapc$baseline[56,]
pcex<-sfc.addScenario(model=pcex,vars="r_bar",values=0.035,inits=1960,
                                          ends=2010,starts=init)
```

```
datapcex<-simulate(pcex)
```

This replicates plot figure 4.3. p. 112

```
time2=c(1950:2000)
plot(time2, datapcex$scenario[as.character(time2),"h_h"]/
            datapcex$scenario[as.character(time2),"v"],
         type="l",xlab="",ylab="",lty=1,lwd=2)
par(new=T)
plot(time2,datapcex$scenario[as.character(time2),"b_h"]/
            datapcex$scenario[as.character(time2),"v"],
     lty=2, lwd=2,type="l",axes=F,ylab="",xlab="")
axis(4,pretty(c(0.750, 1.1*max(datapcex$scenario[,"b_h"]/
            datapcex$scenario[,"v"],na.rm=T))))
legend(x=1970,y=0.78,legend=c("Share of Bills",
        "Share of Money"),lty=c(2,1),lwd=2,bty="n")
grid()
```

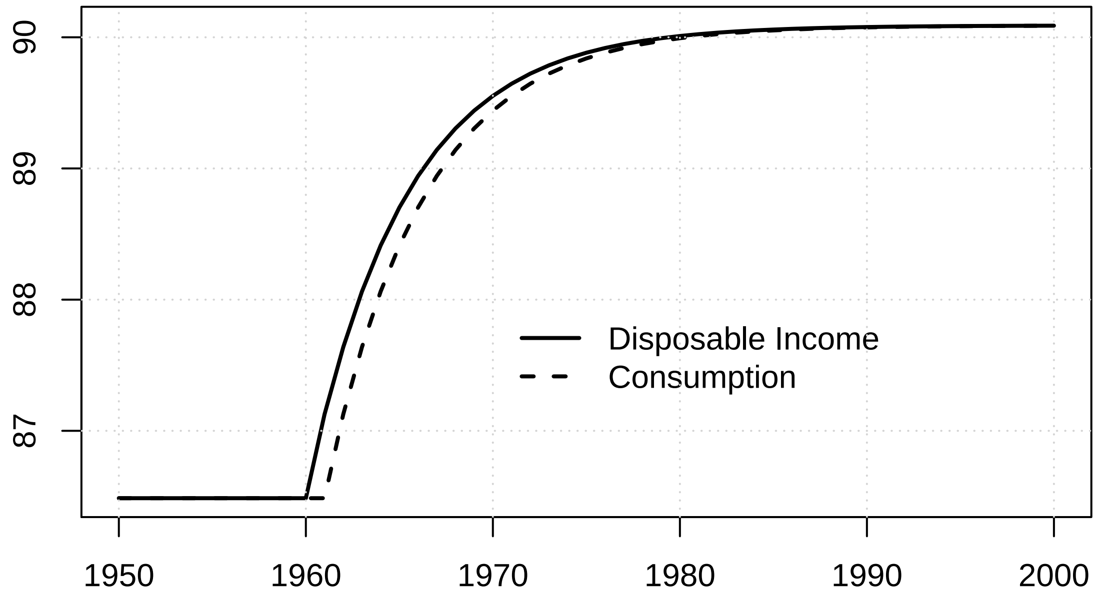


No surprise here, the change in return rates coming from the shock led to a re-allocation between Bills and Money, in favor of the former.

This replicates fig 4.4 . p 113

```
time2=c(1950:2000)
matplot(time2,datapcex$scenario[as.character(time2),c("yd","cons")]
                ,type="l",xlab="",ylab="",lty=1:2,col=1,lwd=2)
legend(x=1970,y=88,legend=c("Disposable Income","Consumption"),
             lwd=c(2,2),lty=c(1,2),bty="n")
```

```
grid()
```



These results are more surprising as increased interest rates lead to an increase both in disposable income and consumption. To understand this, one has to bear in mind that increase in interest rates leads to an increase in government spending and thus to an increase in the *fiscal stance* which determines the steady state: $Y^{\star} = \frac{G + r \cdot B_h^{\star} \cdot (1-\theta)}{\theta}$

Endogenous propensities to consume

We change the propensities to consume to incorporate the fact that they might be impacted by interest rates and thus have

$$\alpha_1 = \alpha_1 0 - \iota \cdot r_{-1} \tag{4.32}$$

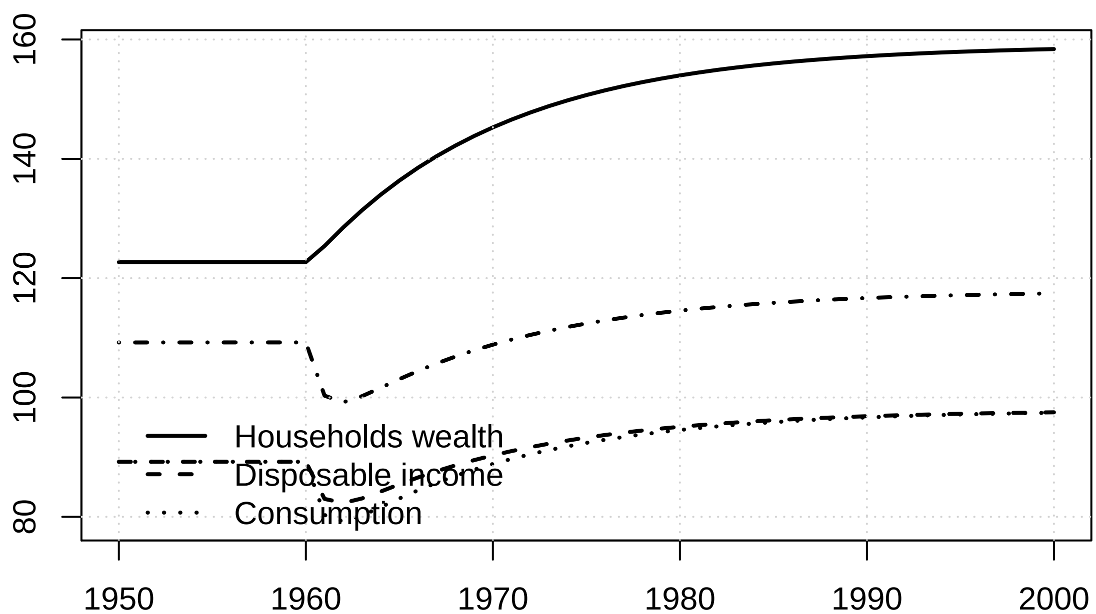
This piece of code shows how to implement this

```
pc_end<-sfc.addEqus(pcex,list(
  list(var="alpha1",eq="alpha10-iota*r(-1)")))
pc_end<-sfc.addVars(pc_end,list(
  list(var="alpha10",init="0.7",
         desc="Endoegnous propensity to consume - autonomous term"),
  list(var="iota",init="10",
         desc="Endoegnous propensity to consume - interest rate impact")))
pc_end$scenarios<-NULL
```

```
pc_end<-sfc.check(pc_end)
datapc_end<-simulate(pc_end)
init = datapc_end$baseline[56,]
pc_end<-sfc.addScenario(model=pc_end,vars="r_bar",values=0.035,
                                                inits=1960,ends=2010,starts=init)
datapc_end<-simulate(pc_end)
```

This replicates plot figure 4.9. p. 123

```
time2=c(1950:2000)
matplot(time2,
     datapc_end$scenario_1[as.character(time2),c("v","yd","cons","y")],
     type="l",xlab="",ylab="",lty=1:4,lwd=2,col=1)
grid()
legend(x=1950,y=100,legend=c("Households wealth","Disposable income",
            "Consumption","GDP"),lty=1:4,lwd=2,bty="n")
```

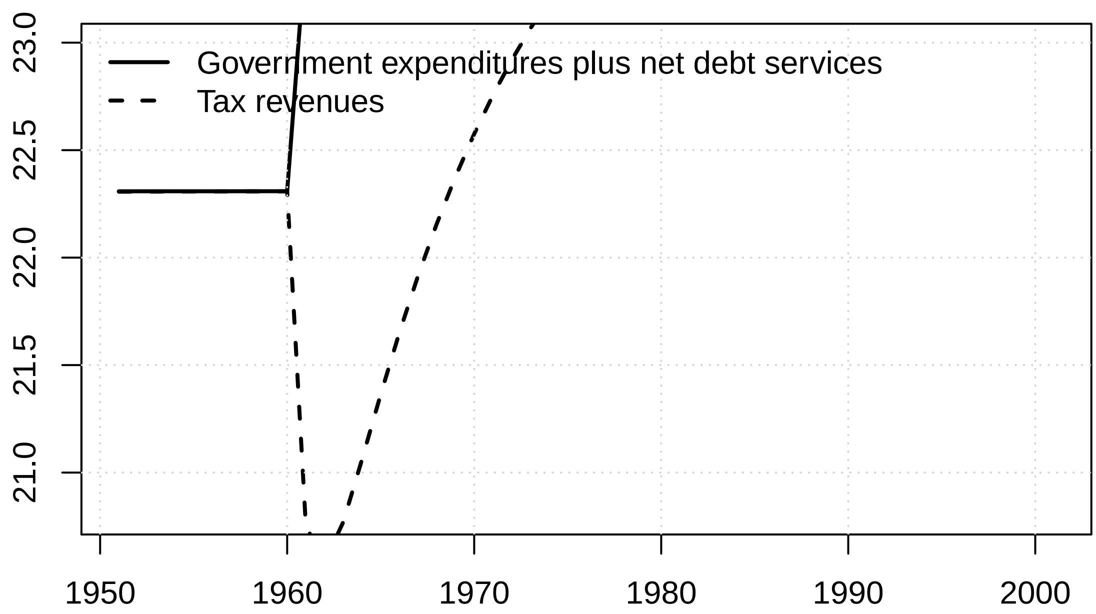


This shows that now the short-term dynamics display the Keynesian *paradox of thrift* but the long-run steady state still shows a positive impact of interests rates on GDP. The short-run recession is governed by the decrease in the propensity to consumed and is slowly compensated by the increase in wealth (and the increased public debt as a counterpart) leading to the new steady state. The steady state is still driven by the fiscal stance.

This replicates plot figure 4.10. p. 123

```
timelag=c(1950:2000)
time2=c(1951:2001)
plot(time2,
```

```
    datapc_end$scenario_1[as.character(time2),c("g")]+
          datapc_end$scenario_1[as.character(timelag),c("r")]*
          datapc_end$scenario_1[as.character(timelag),c("b_s")]-
          datapc_end$scenario_1[as.character(timelag),c("r")]*
          datapc_end$scenario_1[as.character(timelag),c("b_cb")],
    type="l",xlab="",ylab="",lty=1:4,lwd=2,col=1,ylim=c(20.8,23))
lines(time2,datapc_end$scenario_1[as.character(time2),"t"],type="l",xlab="",
          ylab="",lty=2,lwd=2)
grid()
legend("topleft",legend=c("Government expenditures plus net debt services",
          "Tax revenues"),lty=1:4,lwd=2,bty="n")
```



Debt to GDP at the steady state

$$\frac{V^\star}{Y^\star} = \frac{\frac{1-\alpha_1}{\alpha_2}}{1 + \left[\frac{\theta}{1-\theta}\right] - r \cdot \left[(\lambda_0 + \lambda_1 \cdot r) \cdot \frac{1-\alpha_1}{\alpha_2} - \lambda_2\right]} \tag{4.33}$$

The Maastricht treaty: The reference values referred to [...] are: - 3% for the ratio of the planned or actual government deficit to gross domestic product at market - 60% for the ratio of government debt to gross domestic product at market prices.

Liquidity Preferences and model LP

Value of perpetuity, interest rates, exepcted returns and capital gains

The value of a financial asset is supposed to reflect the net present value of future cash flows. Thus for a long-term bond who is never redeemed, the price p_{bL} of the bond is given by

$p_{bL} = \sum \frac{1}{(1+r_{bL})^t} = \frac{1}{r_{bL}}$

Return rate of an asset is equal to both the yield (interests, dividends) and the capital gains normalized to the nominal value of the asset. In the case of a long-term bond:

$Rr_{bL} = r_{bL}(-1) + \frac{\Delta p_{bL}}{p_{bL}(-1)} = \frac{1+p_{bL}-p_{bL}(-1)}{p_{bL}(-1)}$

However, what matter for the households when making their decision is the expected price of bonds p^e_{bL}, given the current price of bonds. This leads to the *pure expected rate of return.*

$PERr_{bL} = r_{bL} + \frac{p^e_{bL}-p_{bL}}{p_{bL}}$

Because expectation are not followed by everyone, households migh incorporate a weight ξ into their expectation formation. This leads to the *expected return rate*

$ERr_{bL} = r_{bL} + \xi \frac{p^e_{bL}-p_{bL}}{p_{bL}}$

The change in prices of bonds leads to capital gains which is one of two parts of the change in nominal value of financial assets. More precisely:

$$\begin{aligned}
\Delta(p_{BL} \cdot BL) &= (p_{bL} \cdot BL) - (p_{bL}(-1) \cdot BL(-1)) \\
&= p_{BL} \cdot BL - p_{BL} \cdot BL(-1) + p_{BL} \cdot BL(-1) - p_{bL}(-1) \cdot BL(-1) \\
&= p_{BL}\,(BL - BL(-1)) + BL(-1)\,(p_{BL} - p_{BL}(-1)) \\
&= p_{BL} \cdot \Delta BL + BL(-1) \cdot \Delta p_{BL}
\end{aligned}$$

Thus total changes in nominal values of an asset is equal to the quantity of assets bought (or sold) at the current price ($p_{BL} \cdot \Delta BL$) and to the capital gains (or losses) made on the stock of assets held at the beginning of the period ($BL(-1) \cdot \Delta p_{BL}$)

Balance Sheet

	Households	Production	Government	Central Bank	Sum
Money	+H			-H	0
Bills	+Bh		-B	+Bcb	0
Bonds	+BL.pbL		-BL.pbL		0
Net worth	-V		+V		0
Sum	0	0	0	0	0

Transaction Flow Matrix

	Households	Production	Government	Central Bank	Sum
Consumption	-C	+C			0
Gov. Exp.		+G	-G		0
Income = GDP	+Y	-Y			0

	Households	Production	Government	Central Bank	Sum
Interests on bills	+rb(-1)*Bh(-1)		-rb(1)*B(-1)	+rb(-1)*Bcb(-1)	0
Interests on bonds	+BL(-1)		-BL(-1)		0
CB profits			+r(-1)*Bcb(-1)	-r(1)*Bcb(-1)	0
Taxes	-T		+T		0
Change in Money	-Δ H			+Δ H	0
Change in Bills	-Δ Bh		+Δ B	-Δ Bcb	0
Change in Bonds	-Δ BL.pbL		+Δ BL.pbL		0
Sum	0	0	0	0	0
Memo: Capital gains	-Δ pbL.BL(-1)		+Δ pbL.BL(-1)		0

Equation list

```
# MODEL
# Determination of output - eq. 5.1
y = cons + g
# Regular disposable income - eq. 5.2
yd_r = y - t + r_b(-1)*b_h(-1) + bl_h(-1)
# Tax payments - eq. 5.3
t = theta*(y + r_b(-1)*b_h(-1) + bl_h(-1))
# Wealth accumulation - eq. 5.4
v = v(-1) + (yd_r - cons) + cg
# Capital gains on bonds - eq. 5.5
cg = (p_bl-p_bl(-1))*bl_h(-1)
# Consumption function - eq. 5.6
cons = alpha1*yd_r_e + alpha2*v(-1)
# Expected wealth - eq. 5.7
v_e = v(-1) + (yd_r_e - cons) + cg
# Cash money - eq. 5.8
h_h = v - b_h - p_bl*bl_h
# Demand for cash - eq. 5.9
h_d = v_e - b_d - p_bl*bl_d
# Demand for government bills - eq. 5.10
b_d = v_e*(lambda20 + lambda22*r_b - lambda23*er_rbl - lambda24*(yd_r_e/v_e))
# Demand for government bonds - eq. 5.11
bl_d = v_e*(lambda30 - lambda32*r_b + lambda33*er_rbl - lambda34*(yd_r_e/v_e))/p_bl
# Bills held by households - eq. 5.12
b_h = b_d
# Bonds held by households - eq. 5.13
bl_h = bl_d
# Supply of government bills - eq. 5.14
b_s = b_s(-1) + (g + r_b(-1)*b_s(-1) + bl_s(-1)) - (t + r_b(-1)*b_cb(-1)) -p_bl*(bl_s-bl_s(-1))
# Supply of cash - eq. 5.15
h_s = h_s(-1) + b_cb - b_cb(-1)
# Government bills held by the central bank - eq. 5.16
b_cb = b_s - b_h
# Supply of government bonds - eq. 5.17
bl_s = bl_h
# Expected rate of return on bonds - eq. 5.18
er_rbl = r_bl+chi*(p_bl_e - p_bl)/p_bl
# Interest rate on bonds - eq. 5.19
r_bl = 1/p_bl
```

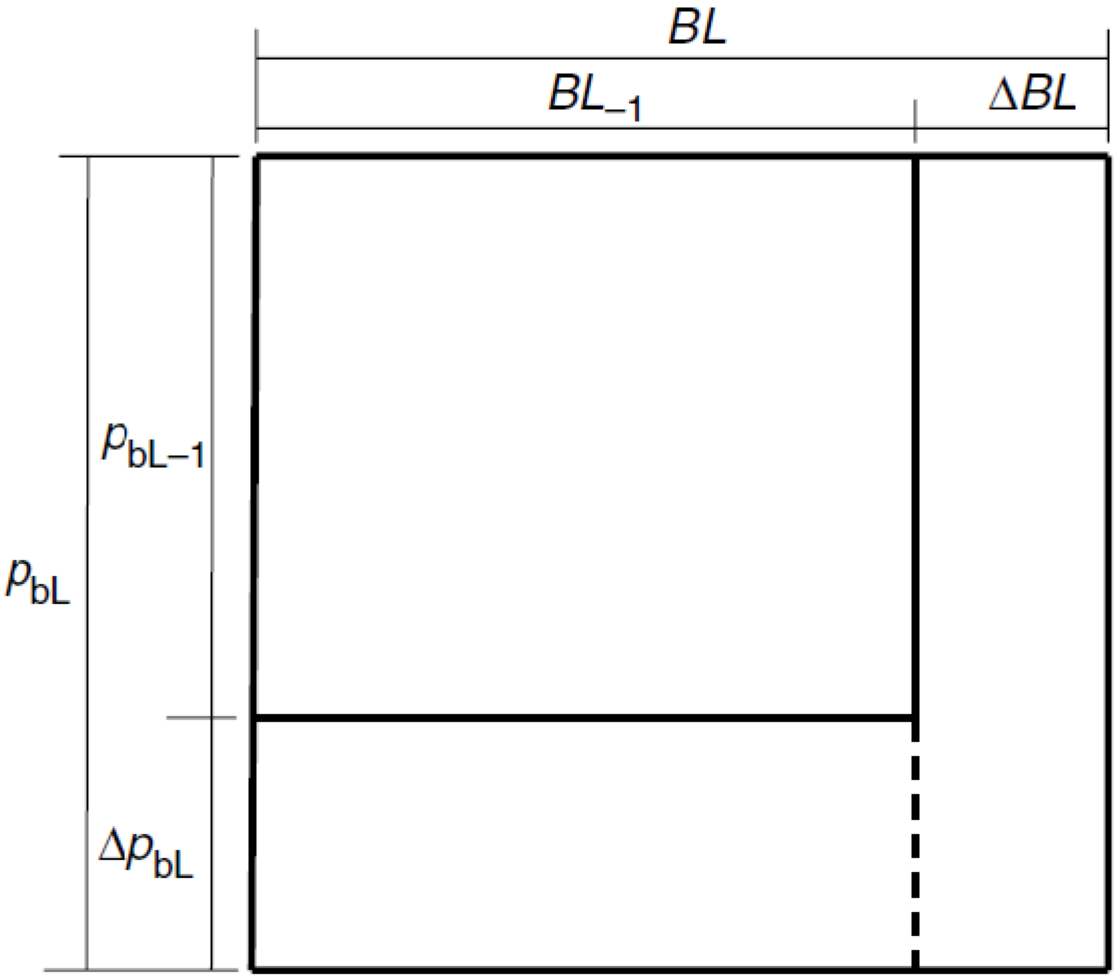
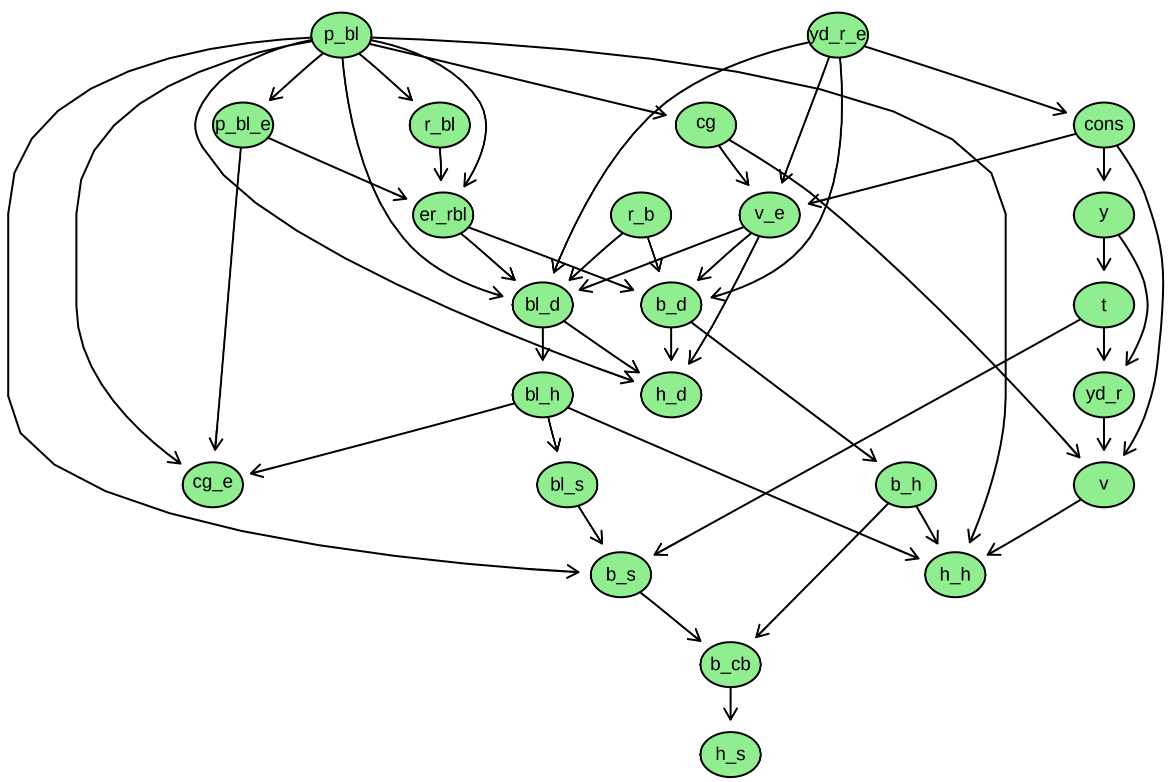


Figure 10: Ostergaard diagram

```
# Expected price of bonds - eq. 5.20
p_bl_e = p_bl
# Expected capital gains - eq. 5.21
cg_e = chi*(p_bl_e - p_bl)*bl_h
# Expected regular disposable income - eq. 5.22
yd_r_e = yd_r(-1)
# Interest rate on bills - eq. 5.23
r_b = r_bar
# Price of bonds - eq. 5.24
p_bl = p_bl_bar
```

Let's have a look at the graph of model LP

```
lp<-sfc.model(fileName="LP.txt",modelName="Model Liquidity Preference")
plot.dag(lp)
```



The adding up of the portfolio behavior

The vertical conditions state that the sum of the exogenous components sums to 1, i.e. you cannot allocate more than the wealth and that the effect of the various return rates and disposable income are substracting the holding of one asset to allow the increase of the holding of another.

$$\lambda_{10} + \lambda_{20} + \lambda_{30} = 1 \quad \text{(ADUP.1)}$$
$$\lambda_{11} + \lambda_{21} + \lambda_{31} = 0 \quad \text{(ADUP.2)}$$
$$\lambda_{12} + \lambda_{22} + \lambda_{32} = 0 \quad \text{(ADUP.3)}$$
$$\lambda_{13} + \lambda_{23} + \lambda_{33} = 0 \quad \text{(ADUP.4)}$$
$$\lambda_{14} + \lambda_{24} + \lambda_{34} = 0 \quad \text{(ADUP.5)}$$

The horizontal conditions, added by Godley (1996) state that the impact of an increase in the 'own rate' of an asset should be equal to the impact of a fall in all the other rates.

$$\lambda_{11} = -(\lambda_{12} + \lambda_{13}) \quad \text{(ADUP.6)}$$
$$\lambda_{22} = -(\lambda_{21} + \lambda_{23}) \quad \text{(ADUP.7)}$$
$$\lambda_{33} = -(\lambda_{31} + \lambda_{32}) \quad \text{(ADUP.8)}$$

Other authors propose the symetry conditions that state that an increase in the return rate of one asset A will have the same impact on the holding of asset B as the increase in the return rate of asset B will have on the holding of asset A

$$\lambda_{12} = \lambda_{21} \quad \text{(ADUP.9)}$$
$$\lambda_{13} = \lambda_{31} \quad \text{(ADUP.10)}$$
$$\lambda_{23} = \lambda_{32} \quad \text{(ADUP.11)}$$

Note that vertical + symetry imply automatically horizontal but that vertical + horizontal does not necessarily imply symetry.

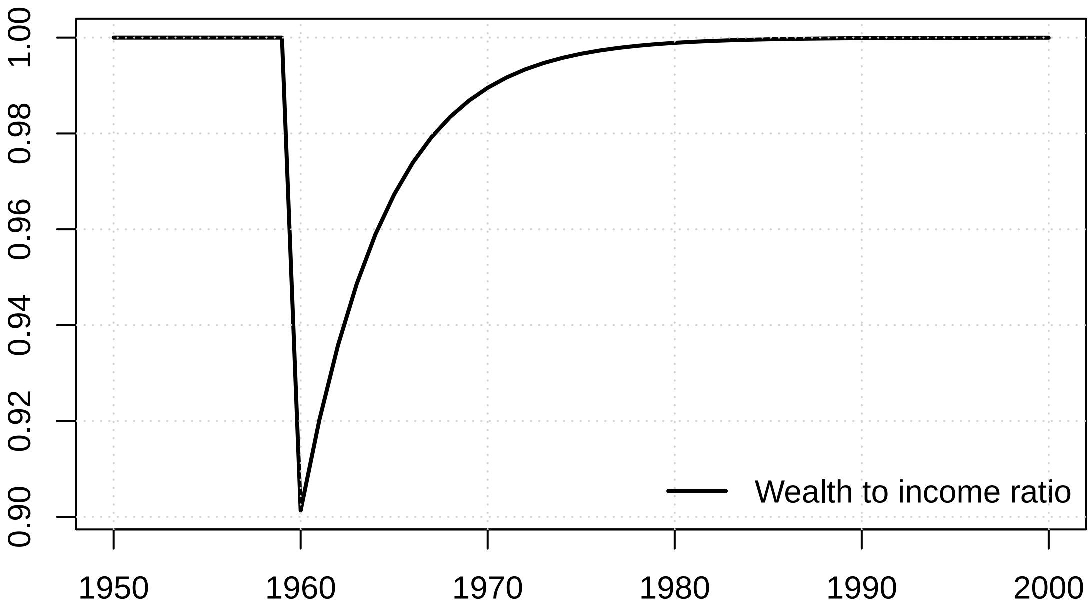
Higher interest rates

The first scenario run looks at the impact of interest rates (short- and long-term) on real demand. We assume that the government increases short-term rates from 3% to 4% and the long-term rate from 5% to 6.66%. We assume that this increase is cmpletely unexcpected and that the Treasury is believed in that no other change will occur.

```
lp<-sfc.addScenario(model=lp,vars=list(c("r_bar","p_bl_bar")),values=list(c(0.04,15)),
                    inits=1960,ends=2000)
datalp<-simulate(lp)
```

This replicates plot figure 5.2. p. 152

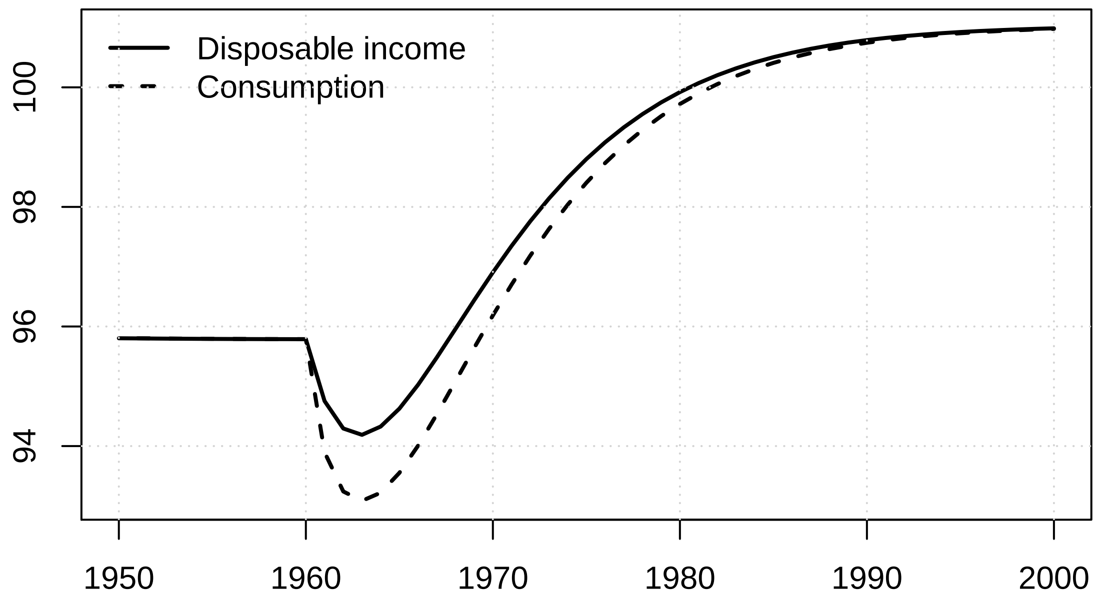
```
time2=c(1950:2000)
plot(time2, datalp$scenario_1[as.character(time2),"v"]/
         datalp$scenario[as.character(time2),"yd_r"],
     type="l",xlab="",ylab="",lty=1,lwd=2)
legend("bottomright",legend=c("Wealth to income ratio"),lty=c(1),lwd=2,bty="n")
grid()
```



We see that the increase in long-term interest rate leads to capital losses on bonds, leading to a decrease in welath to income ratio. However because interest rates have increase, households have now a higher disposable income leading to a replenishing of their wealth. Note that as in model PC, the steady state GDP is given by $Y^{\star} = \frac{G+(r_B \cdot B_h^{\star} + BL_S^{\star}) \cdot (1-\theta)}{\theta} = \frac{G_{NT}}{\theta}$

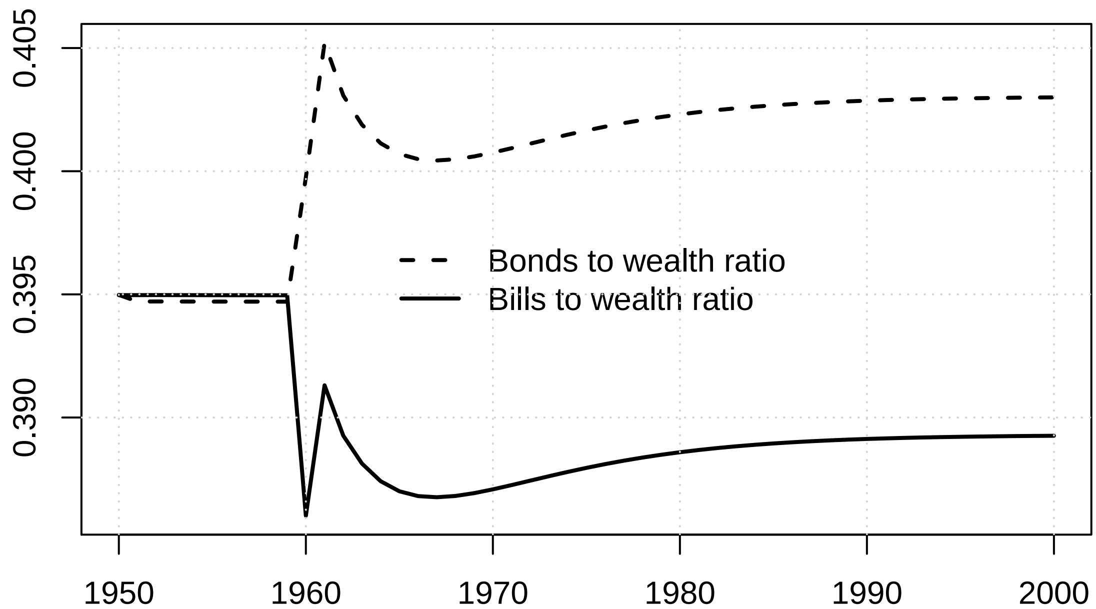
This replicates plot figure 5.3. p. 152

```
time2=c(1950:2000)
matplot(time2, datalp$scenario_1[as.character(time2),c("yd_r","cons")],
          type="l",xlab="",ylab="",lty=c(1,2), col=1,lwd=2)
legend("topleft",legend=c("Disposable income","Consumption"),lty=c(1,2),lwd=2,bty="n")
grid()
```



This replicates plot figure 5.4. p. 153

```
time2=c(1950:2000)
matplot(time2, cbind(datalp$scenario_1[as.character(time2),c("b_h")],
            datalp$scenario_1[as.character(time2),c("p_bl")]*
            datalp$scenario_1[as.character(time2),c("bl_h")])/
                datalp$scenario_1[as.character(time2),c("v")],
         type="l",xlab="",ylab="",lty=c(1,2), col=1,lwd=2)
legend("center",legend=c("Bonds to wealth ratio","Bills to wealth ratio"),
        lty=c(2,1),lwd=2,bty="n")
grid()
```



Introducing household liquidity

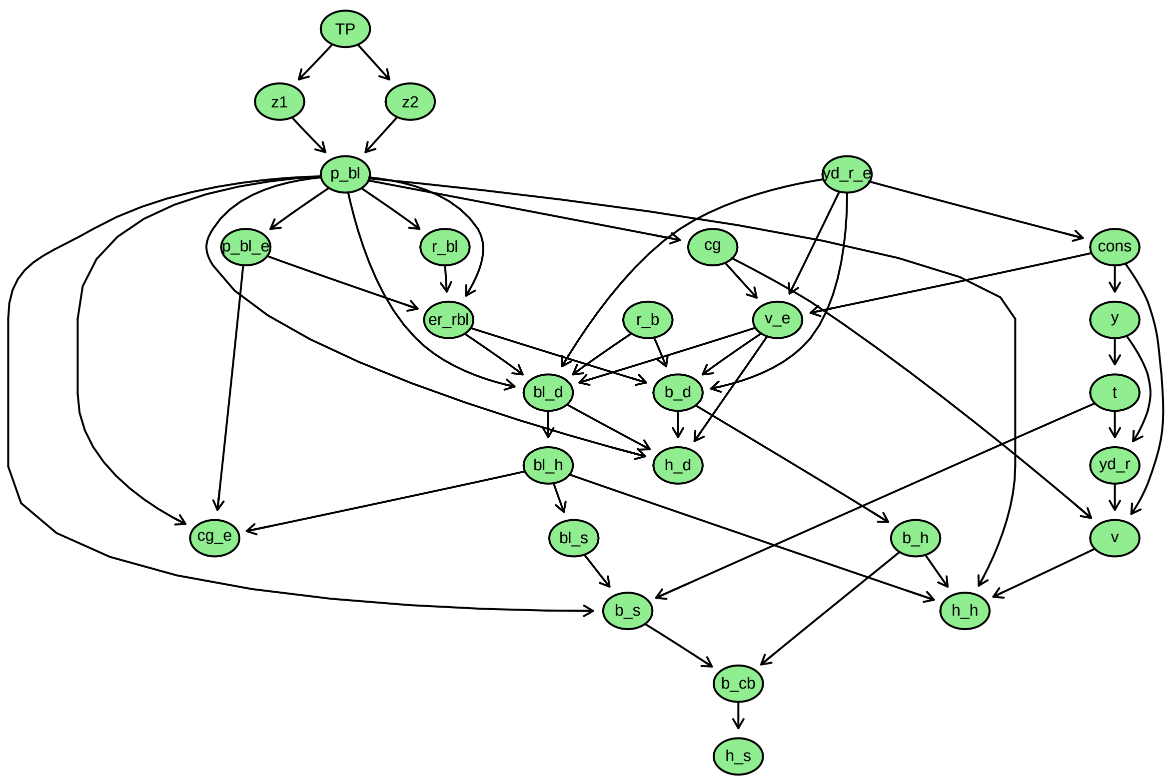
In order to introduce liquidity preferences, we need to modify the model. The first modification is to change the expectation on bonds price to include (i) a learning element and (ii) a stochastic term. Because expectation are going to play a role, we also need to modify the price equation for bonds such that it now reflects a corridor approach to bond pricing. The assumption is thus that the Treasury will try to pin down the price of bonds but only to a certain extent. If the demand is too large (or too low) the treasury will let prices (and interest) float. We assume that the government aims at a certain debt strcuture in terms of maturity.

```
#Adding the equations
lp2<-sfc.addEqus(lp,list(
    list(var="z1",equ = "TP>top"),
    list(var="z2",equ = "TP<bot"),
    list(var="TP",equ = "(bl_h(-1)*p_bl(-1))/(bl_h(-1)*p_bl(-1)+b_h(-1))")))
#Adding the parameters
lp2<-sfc.addVars(lp2,list(
    list(var="betae",init=0.5),
    list(var="beta",init=0.01),
    list(var="add",init=0),
    list(var="top",init=0.495),
    list(var="bot",init=0.505)
))
#Adding initial values to the endogenous variables that have lags
lp2<-sfc.editEnd(lp2,var="p_bl_e", init=20)
lp2<-sfc.editEnd(lp2,var="p_bl", init=20)
```

```r
#Modifying the existing equations
lp2<-sfc.editEqus(lp2,list(
    list(var="p_bl", eq="(1+z1*beta-z2*beta)*p_bl(-1)"),
    list(var="p_bl_e",equ = "p_bl_e(-1)-betae*(p_bl_e(-1) - p_bl) + add")))
#Removing all existing scenarios (coming from LP)
lp2$scenarios<-NULL
#Checking that the model is complete
lp2<-sfc.check(lp2,fill=F)
#Adding the scenario
lp2<-sfc.addScenario(model=lp2,vars=list(c("r_bar")),values=list(c(0.035)),inits=1960,
                                            ends=2000)
lp2<-sfc.addScenario(model=lp2,vars=list(c("add")),values=list(c(-3)),inits=c(1955),
                                            ends=1955)
```

Let's have a look at the graph of model LP2

```r
plot.dag(lp2)
```



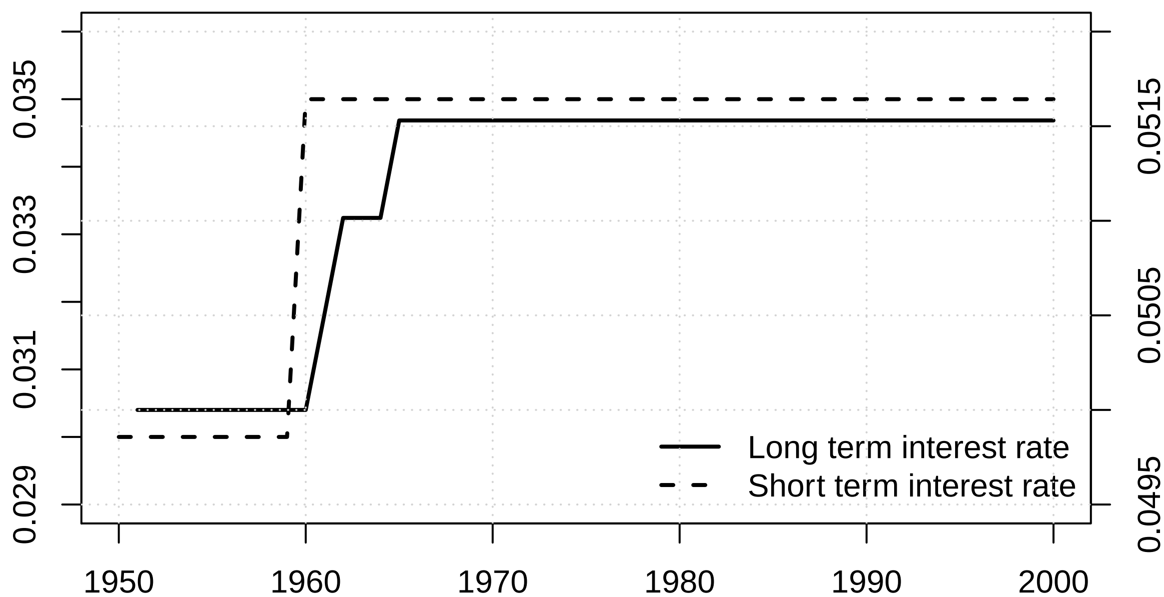
We are now ready to simulate the model

```r
datalp2<-simulate(lp2)
```

This replicates plot figure 5.5. p. 156

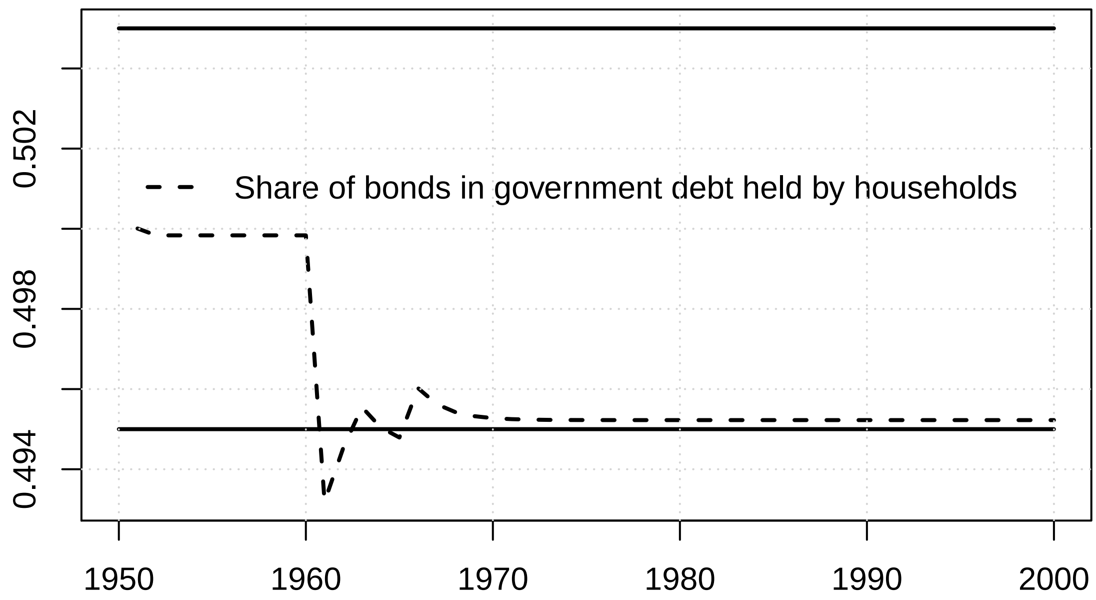
```r
time2=c(1950:2000)
plot(time2,datalp2$scenario_1[as.character(time2),"r_b"],
     lty=2, lwd=2,type="l",ylab="",xlab="",ylim=c(0.029,0.036))
par(new=T)
```

```
plot(time2, datalp2$scenario_1[as.character(time2),"r_bl"],
         type="l",xlab="",axes=F,ylab="",lty=1,lwd=2,ylim=c(0.0495,0.052))
axis(4,pretty(c(0.0495,0.052)))
legend("bottomright",legend=c("Long term interest rate",
        "Short term interest rate"),lty=c(1,2),lwd=2,bty="n")
grid()
```



This replicates plot figure 5.6. p. 156

```
time2=c(1950:2000)
matplot(time2, datalp2$scenario_1[as.character(time2),c("TP","bot","top")],
         type="l",xlab="",ylab="",lty=c(2,1,1), col=1,lwd=2)
legend(x=1950,y=0.502,legend=c("Share of bonds in government debt held by households"),
             lty=c(2),lwd=2,bty="n")
grid()
```

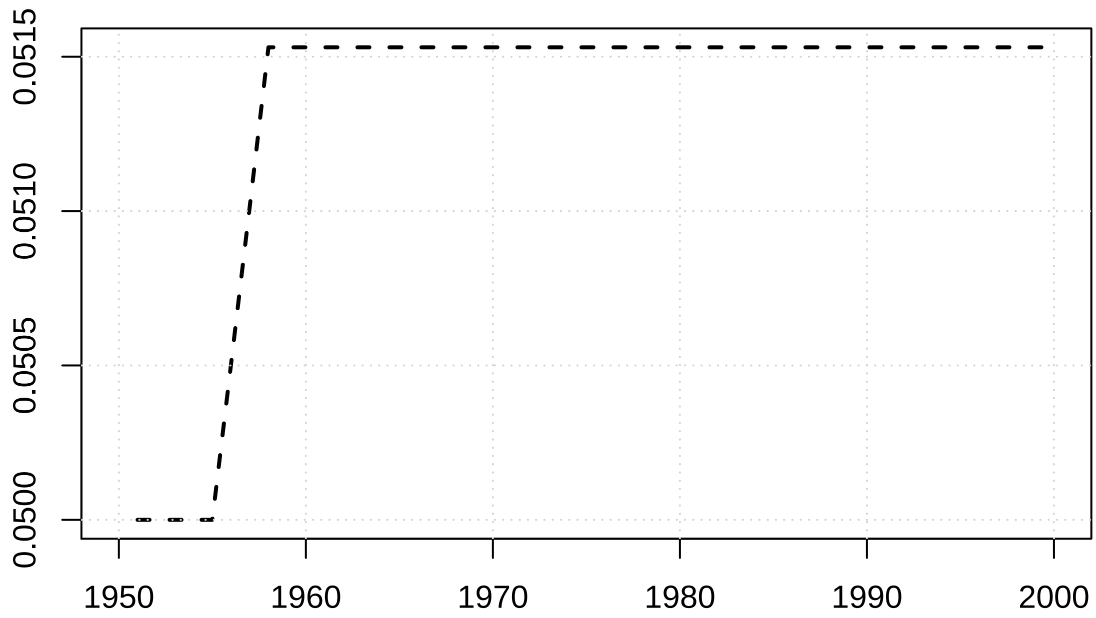


Let's add a second scenario by which there is an expected fall in the price of long-term bonds.

```
lp2<-sfc.addScenario(model=lp2,vars=list(c("add")),values=list(c(-3)),inits=c(1955),
                                                  ends=1955)
datalp2<-simulate(lp2)
```

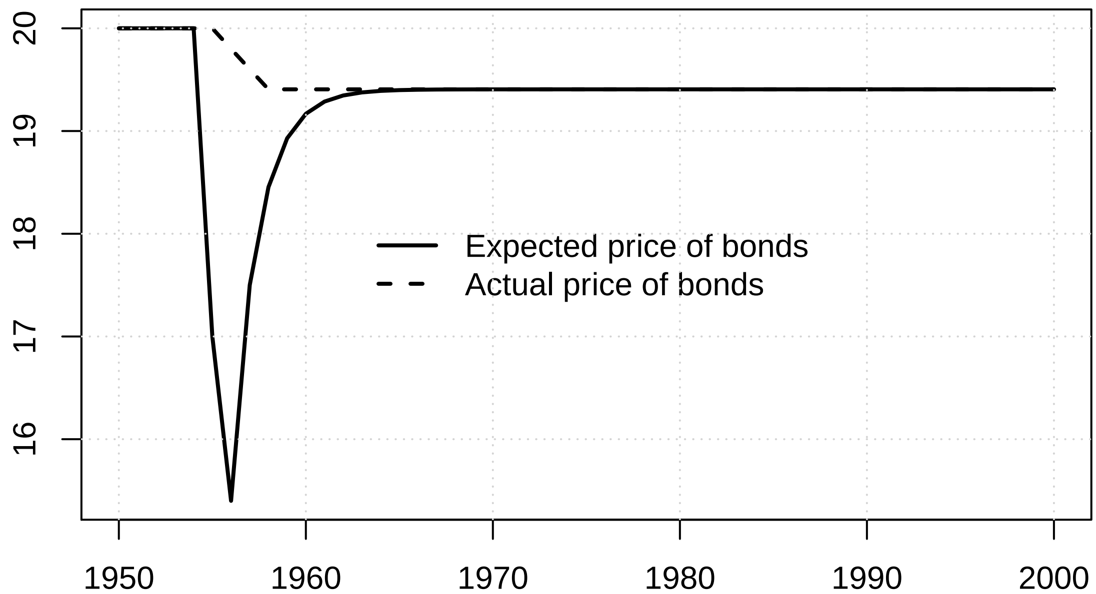
This replicates plot figure 5.7. p. 157

```
time2=c(1950:2000)
matplot(time2, datalp2$scenario_2[as.character(time2),c("r_bl")],
         type="l",xlab="",ylab="",lty=c(2,1,1), col=1,lwd=2)
legend(x=1950,y=0.502,legend=c("Share of bonds in government debt held by households"),
             lty=c(2),lwd=2,bty="n")
grid()
```



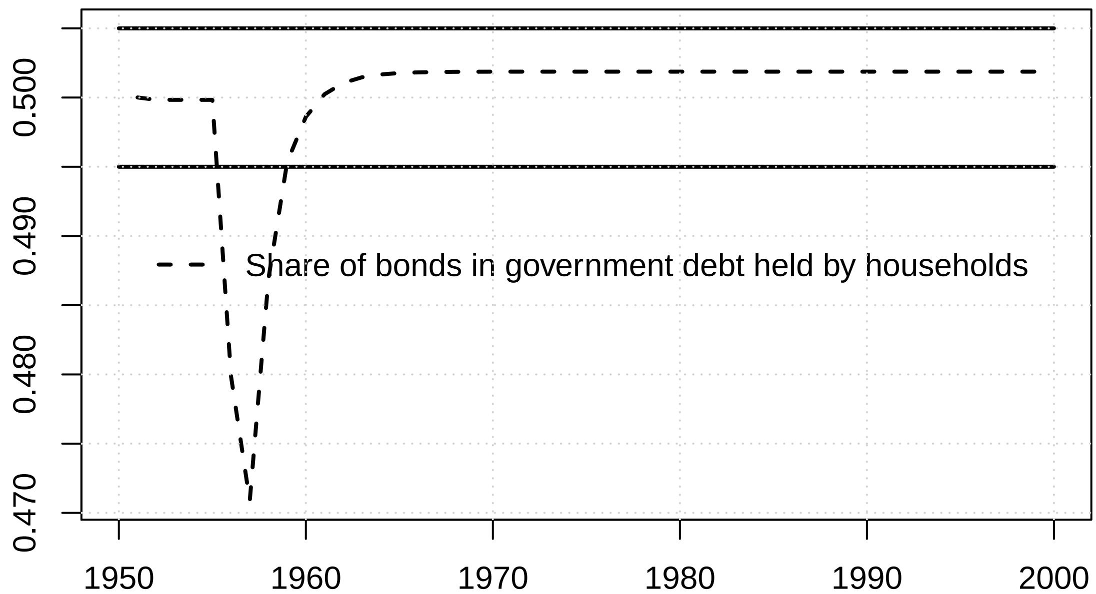
This replicates plot figure 5.8. p. 158

```
time2=c(1950:2000)
matplot(time2, datalp2$scenario_2[as.character(time2),c("p_bl_e","p_bl")],
         type="l",xlab="",ylab="",lty=c(1,2), col=1,lwd=2)
legend("center",legend=c("Expected price of bonds","Actual price of bonds"),
              lty=c(1,2),lwd=2,bty="n")
grid()
```



This replicates plot figure 5.9. p. 158

```
time2=c(1950:2000)
matplot(time2, datalp2$scenario_2[as.character(time2),c("TP","bot","top")],
         type="l",xlab="",ylab="",lty=c(2,1,1), col=1,lwd=2)
legend("center",legend=c("Share of bonds in government debt held by households"),
              lty=c(2),lwd=2,bty="n")
grid()
```



Modelling the firm: price, investment and profits

The role of banks: credit creation and investment

- Firms will always have an investment function determinin the desired level of production capacity.
  - Model BMW: $K^T = \kappa \cdot Y_{-1}$
- However, the realisation of investment might be constrained by banks credit, leading to an eventual different between desired investment and actual investment.
  - Model BMW: $I_d = \gamma \cdot (K^T - K_{-1}) + DA$ where γ can be interpreted as the outcome between firms and banks interactions.

Functional income distribution

- The usual way to distribute income is between wage and profits (i.e. looking at wage share and profit share).
  - It is however often forgotten that profits are then decomposed further between different type of profits: retained earnings and distributed dividends.
  - But that's not the end of the story: we can disaggregate even further profits between gross profits and net profits, the difference being 1. interest payments, 2. taxes and 3. capital amortisation
  - It is usually the net profits that are then distributed between retained earnings and dividends.
- In model BMW: depreciation allowances (amortisation) are accounted for explicitely ($DA = \delta.K_{-1}$), so are interest payments. We thus have the following functional distribution of income: $Y = WB + r_{l,-1} \cdot L_{-1} + DA$

- Note that in this case, the wage bill is endogenous since both interests payments and depreciation depend only on past values

Consumption function

- In model BWM, tyhe consumption function contains an autonomous term: $C = \alpha_0 + \alpha_1 \cdot YD + \alpha_2 \cdot M_{-1}$.
  - The autonomous term plays an important role in the dynamics of the model as it "kick-starts" the model. It plays the same role as government expenditures in the previous model we've seen.

Steady state of BMW

- Let's do it manualy, starting from the list of equations:

```
#Supply of consumption goods - eq. 7.1
 c_s = c_d
#Supply of investment goods - eq. 7.2
 i_s = i_d
#Supply of labour - eq. 7.3
 n_s = n_d
#Transactions of the firms
#GDP - eq. 7.5
 y = c_s + i_s
#Wage bill - eq. 7.6
 wb_d = y - r_l(-1)*l_d(-1) - af
#Depreciation allowances - eq. 7.7
 af = delta*k(-1)
#Demand for bank loans - eq. 7.8
 l_d = l_d(-1) + i_d - af
#Transactions of households
#Disposable income - eq. 7.9
 yd = wb_s + r_m(-1)*m_h(-1)
#Bank deposits held by households - eq. 7.10
 m_h = m_h(-1) + yd - c_d
#The wage bill
#"Supply" of wages - eq. 7.13
 wb_s = w*n_s
#Labour demand - eq. 7.14
 n_d = y/pr
#Wage rate - eq. 7.15
 w = wb_d/n_d
#Household behaviour
#Demand for consumption goods - eq. 7.16
 c_d = alpha0 + alpha1*yd + alpha2*m_h(-1)
#The investment behaviour
#Accumulation of capital - eq. 7.17
 k = k(-1) + i_d - da
#Depreciation allowances - eq. 7.18
 da = delta*k(-1)
#Capital stock target - eq. 7.19
 k_t = kappa*y(-1)
# below is an additional equation I've defined to get output to capital ratio as in figure 7.4, the mod
OCR = y/k(-1)
```

```r
#Demand for investment goods - eq. 7.20
 i_d = gamma*(k_t - k(-1)) + da
```

- Important points to discuss:
  - Role of autonomous consumption
  - The existence of non-negative GDP steady state
  - The equality between capital stock and wealth

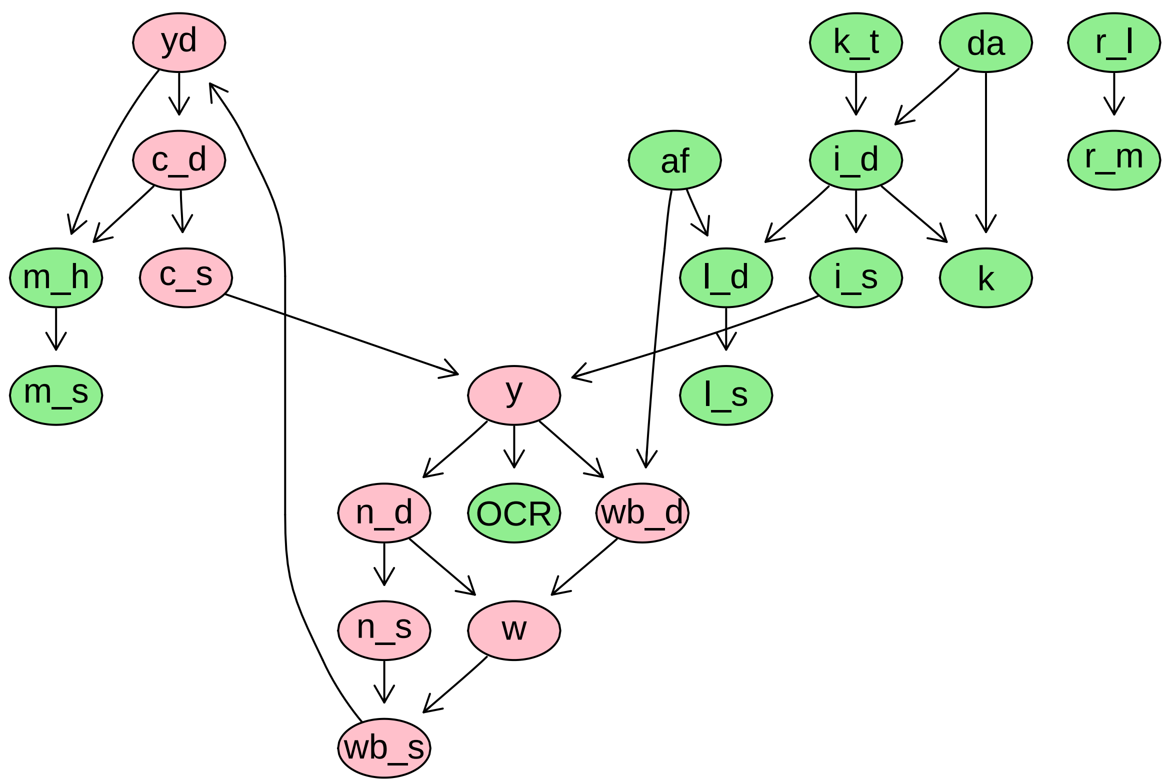
Out of equilibrium values

- To do so, let's have a look at the graph representation of BMW:

```r
BMW<-sfc.model("BMW.txt")
```

```
## Warning in sfc.check(model, fill = fill): The following variables have lags
## but no initial values: - r_l - r_m
```

```r
plot.dag(BMW)
```



- We need to solve the pink cycle and express Y (and K) as a function of only lagged variables.

Stability

- Very brief intro because stability analysis of difference system is much more complicated than for differential systems (will be covered by Devrim in the second part of the semester).

- The stability of a system of equations $z_t = A \cdot z_{t-1} + c$ can be analysed via the trace, determinant and discriminant of A
  - trace = sum of diagonal terms, for 2x2 matrix: $tr_A = a_{11} + a_{22}$
  - determinant, for a 2x2 matrix: $det_A = |A| = a_{11} \cdot a_{22} - a_{21} \cdot a_{12}$
  - discriminat: $\Delta = tr_A^2 + 4 \cdot det_A$
- Necessary condition for stability is that tyeh absolute value of the determinant is smaller than 1
- If this is respected, then either the determinant is positive or if it is negative, the trace has to be smaller than 2 if and only if $-det_a < 1 - |tr_A|$.
- We won't go in the details, but the analysis highlight some important characteristics:
  - the system will converge provided the marginal propensity to save is higher than the marginal propensity to invest

More on profits and inventories

- Inventories are fundamental for the working of a firm in terms of production smoothing, response to sudden changes in demand and dealing with uncertainty. They are very much related to the notion of time in SFC (and other types of post-Keynesian) models. Time is historical, based on past observation and projected in the future by expectations, on the contrary of mainstream models where the future is pulled in the present.
- Example of a current account matrix for firms

Components	Firms Current account	Firms Capital account
Sales	$+S$	
Change in the value of inventories	$+\Delta IN$	-ΔIN
Wages	$-WB$	
Interest on loans	$-r_{l-1}L_{-1}$	
Entrepreneurial profits	$-F$	
Change in loans	$+\Delta L$	
Sum	0	0

- Entrepreneurial vs total profits
  - Entrepreneurial profits: $F = S - (WB - \Delta IN + r_{l,-1}IN_{-1})$
  - Total profits: $F = S - (WB - \Delta IN)$
- Historical wage costs vs total historic costs
  - Historical wage costs: $HWC = (WB - \Delta IN)$
  - Total istorical costs: $HC = (WB - \Delta IN) + r_{l,-1}IN_{-1}$
- We can show that profits can always be expressed as a fraction of historical costs: $F = \phi' HC$, implying that sales can also be expressed as a function of historical costs only
  - $S = F + HC
  - $S = (1 + \phi')HC$
  - But Sales are also $S = s.p$, we can thus show that $p = (1 + \phi')HUC$ where $HUC = \frac{HC}{s}$ is the historical unit cost.
- Important distinction between entrepreneurial profits and cash flows and the definition of national accounts profits
  - Entrepreneurial profits: $F = S - (WB - \Delta IN + r_{l,-1}IN_{-1})$
  - Cash flow: $CF = S - WB - r_{l,-1}IN_{-1}$
  - NIPA profits: $F_{nipa} = S - WB + \Delta IN - \Delta UCin_{-1}$ in order to remove stock appreciation or inventory valuation adjustment which are not "proper flows"

References

Foley, D. 1975. "On Two Specifications of Asset Equilibrium in Macroeconomic Models." *Journal of Political Economy* 83 (2).

Godley, W., and M. Lavoie. 2007. *Monetary Economics An Integrated Approach to Credit, Money, Income, Production and Wealth.* Palgrave MacMillan, New York.

Godley, Wynne. 1999. "Seven Unsustainable Processes: Medium-Term Prospects and Policies for the United States and the World." Strategic Analysis. The Levy Economic Institute of Bard College.

Turnovsky, S. 1977. *Macroeconomic Analysis and Stabilization Policy.* Cambridge University Press.