SimBEL: Calculate the best estimate in life insurance with Monte-Carlo techniques

Quentin Guibert

Univ Lyon, Université Claude Bernard Lyon 1, ISFA, Laboratoire SAF EA2429, F-69366, Lyon, France
Prim'Act, Paris, France
Email: quentin.guibert@primact.fr

R in Insurance 2017
8th June 2017, Paris, France

Joint work with G. de Kervénoaël and M. Tammar (Prim'Act)

Best estimate calculation

- Under Solvency II, liabilities in life insurance are valued based on a *market consistency* principle (Kemp, 2009; Vedani *et al.*, 2017), taking into account:
  - financial options and guarantees,
  - future management actions, e.g. profit sharing rules,
  - the policyholder' behavior,
  - both undertaking and financial risks.
- A stochastic Asset Liability Management (ALM) model based on Monte-Carlo balance sheet projection is generally implemented to compute the best estimate of liabilities (see art. 77 directive Solvency II)

$$BE_t = \mathbb{E}\left[\sum_{u>t} \delta_u CF_u\right]$$

  - δ_u, the stochastic deflator at time u ;
  - CF_u, the net payment cash-flows at time u.

European Literature

- Profit sharing rules (see e.g. Grosen and Løchte Jørgensen, 2000; Bacinello, 2001; Ballotta *et al.*, 2006; Kling *et al.*, 2007).
- ALM (see e.g. Bauer *et al.*, 2006; Hainaut, 2009).
- Policyholder's behavior (see e.g. Planchet and Thérond, 2007; Milhaud *et al.*, 2011; Bauer *et al.*, 2006; Eling and Kochanski, 2013).
- The French valuation model is rarely described, but is quite complex as insurers have a higher leeway to distribute profit sharing (Borel-Mathurin *et al.*, 2015).

Aims

- In France, most of such valuation models are developed by commercial firms or directly by insurers. They are no available for students, researchers, ...
- No package to forecast assets and liabilities is available for insurance obligations.
- An ALM model requires algorithms to forecast both assets and liabilities at a very granular level under the local gaps. It is coupled with an Economic Scenarios Generator (ESG).
- With large asset and liability portfolios, the computation can be very time consuming.

- Our aims:
  - Develop a flexible `R`-package to compute easily the best estimate of a life participating contract, especially a French euro-denominated contract.
  - Usable for the Solvency Capital Requirement(SCR) computation.
  - Flexible architecture allowing to project both data and assumptions as it is required for example for the Own Risk Solvency Assessment (ORSA) purpose.

The R-package SimBEL

- The package is implemented in an oriented object fashion in S4.
- An access to the last development version on GitHub is available on demand.
- The user guide and a large documentation (in French at the moment) is available.
- To install the package

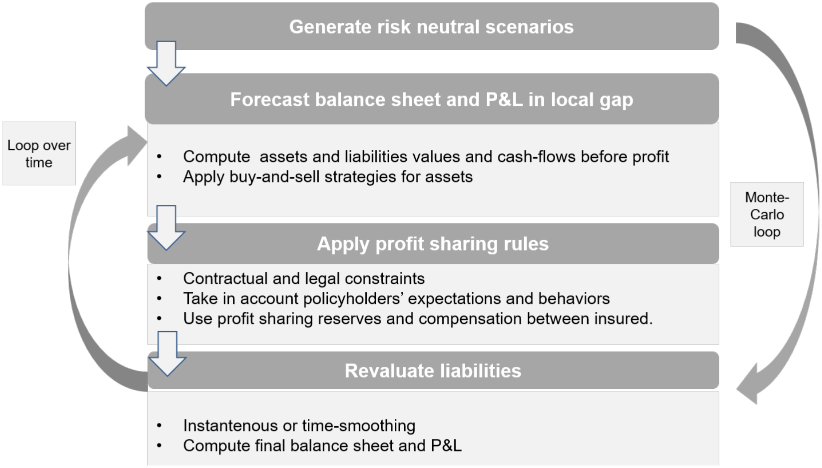
```
library(devtools)
library(githubinstall)
install_github("xxx", auth_token = "yyy")
```

- To load the package

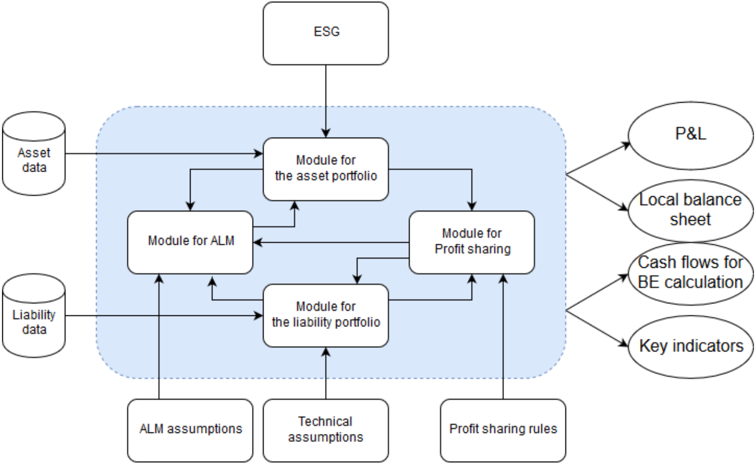
```
library(SimBEL)
```

Overview of the calculation process

The best estimate is calculated following this general process (Laurent *et al.*, 2016)



Overview of the structure (only cash flows projection)



Main functionalities

1. Liability module
   - Both saving and retirement products with participation can be modeled.
   - The liability side is modeled using model points, which represent the technical characteristics of each contract.
   - An user should provide additionally:
     - Technical assumptions (tables with mortality rates and static lapse rates, parameters for dynamic lapses),
     - expenses assumptions,
     - the current value of other provisions.

Main functionalities

1. Liability module
   - Both saving and retirement products with participation can be modeled.
   - The liability side is modeled using model points, which represent the technical characteristics of each contract.
   - An user should provide additionally:
     - Technical assumptions (tables with mortality rates and static lapse rates, parameters for dynamic lapses),
     - expenses assumptions,
     - the current value of other provisions.

Main functionalities

1. Liability module
   - Both saving and retirement products with participation can be modeled.
   - The liability side is modeled using model points, which represent the technical characteristics of each contract.
   - An user should provide additionally:
     - Technical assumptions (tables with mortality rates and static lapse rates, parameters for dynamic lapses),
     - expenses assumptions,
     - the current value of other provisions.

Main functionalities

2. Asset module
   - Four asset classes are modeled:
     - Fixed bonds (Gouv. and Corporate),
     - Equities,
     - Properties,
     - Cash account.
   - An user should provide:
     - ESG tables for asset projection,
     - a reference portfolio for future reinvestments,
     - an investment strategy,
     - the current value for asset provisions,
     - fees on asset.

Main functionalities

2 Asset module

- Four asset classes are modeled:
  - Fixed bonds (Gouv. and Corporate),
  - Equities,
  - Properties,
  - Cash account.
- An user should provide:
  - ESG tables for asset projection,
  - a reference portfolio for future reinvestments,
  - an investment strategy,
  - the current value for asset provisions,
  - fees on asset.

Main functionalities

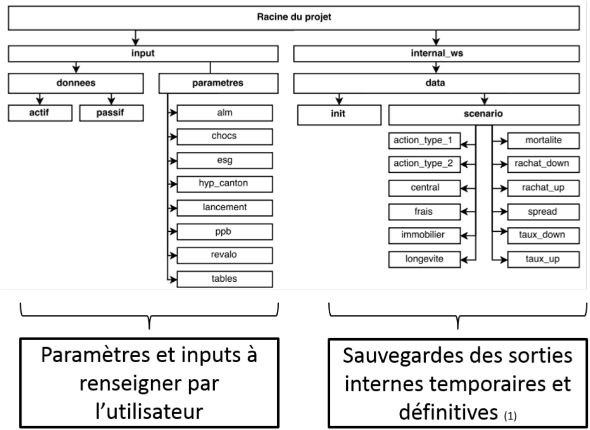
3. Balance sheet modules
   - Asset and liability are linked together with an object called a Canton.
   - When projecting a Canton, ALM and surplus appropriation scheme are applied.
4. Best estimate module
   - Project a Canton for each simulation.
   - Calculate the best estimate based on the initial situation of a canton.

Main functionalities

3. **Balance sheet modules**
   - Asset and liability are linked together with an object called a Canton.
   - When projecting a Canton, ALM and surplus appropriation scheme are applied.
4. **Best estimate module**
   - Project a Canton for each simulation.
   - Calculate the best estimate based on the initial situation of a canton.

Loading data

- Lots of data and parameters are required.
- A module is designed to feed all these data from a repository contained csv files.
- Create a Canton for each shock defined in Solvency II standard formula.



Loading data

- To load the addresses for data repositories.

```
root <- new("Initialisation", root_address = getwd())
root <- set_architecture(root)
```

- To load data related to initialize a Canton.

```
init_SimBEL(racine)
```

- To create and save a Canton for each shocked situation as defined by the standard formula.

```
init_scenario(racine)
```

Canton class

- A Canton stores the current picture of the balance sheet and all parameters to project it.

```
canton@annee # Number of projection years
canton@hyp_canton # Some general assumptions
canton@mp_esg # Extraction of an ESG table for
the current projection year and simulation
```

- To call the Asset portfolio.

```
canton@ptf_fin
```

- To call the Liability portfolio.

```
canton@ptf_passif
```

Liability classes

- Two classes: euro-denominated French saving and pension guarantees.
- Each instantiation of these classes is a product with specific features in terms of expenses and contractual profit sharing rate.
- Liabilities are projected on an annual basis.
- Example for a saving product

```
ee1 <- canton@ptf_passif@eei[[1]] # The first
saving product
class(ee1) # "EpEuroInd"
```

Liability classes

- Model points are stored in a data.frame object.

```
str(ee1@mp)
```

```
'data.frame':   15 obs. of  29  variables:
# num_mp        : int  1 2 3 ...
# ...
# age           : int  40 40 40   ...
# gen           : int  1900 1900 1900  ...
# num_tab_mort  : Factor w/ 1 level "TM2": 1 1 1 ...
# chgt_enc      : num  0.007 0.007 0.007  ...
# ...
# pm            : int  900 11600 12000  ...
# nb_contr      : int  1 1 1   ...
# anc           : int  0 0 0  ...
# ...
# tx_cible      : Factor w/ 1 level "Meth1": 1 1 1 ...
# prime         : int  0 0 0  ...
# tx_tech       : num  0 0 0 ...
# ...
```

Liability cash flows

- Technical assumptions are attached to the liability portfolio.

```
canton@ptf_passif@ht # All technical assumption
# The mortality table 'TM1'
canton@ptf_passif@ht@tables_mort[["TM1"]]
```

- To compute premiums

```
prem <- calc_primes(ee1)
```

- To compute lapse and mortality rates

```
rates <- calc_proba_flux(ee1, ptf_passif@ht)
```

- To compute minimal revalorisation rates and target rates

```
tx_min <- calc_tx_min(ee1)
target <- calc_tx_cible(ee1, list(ptf_passif@ht,
 list_rd = list(0.02,0.01,0.01,0)))
```

Liability cash flows

- To compute benefits

```
ben <- calc_prest(ee1, rates, tx_min, an = 1,
 method = "normal", tx_soc = 0.155)
```

- To compute mathematical reserves

```
pm <- calc_pm(ee1, prem[["flux"]], ben[["flux"]],
target, tx_min, an = 1, method = "normal",
 tx_soc = 0.155)
```

- To forecast liability portfolio over 1 year

```
proj <- proj_annee_av_pb(an = 1, x = ptf_passif,
tx_soc = 0.155, coef_inf = 1,
list_rd = list(0.02,0.01,0.01,0))
```

Liability cash flows

- Cash-flows and mathematical reserves are aggregated by product

```
# Outputs
# Cash-flows by product
proj[["flux_agg"]]
# Mathematical reserves and the number of contracts
proj[["stock_agg"]]
```

- These outputs can be used to build some checks.

Asset cash-flows

- Four asset classes: their dynamics are given by the ESG tables.
- To print asset allocation.

```
print_alloc(canton@ptf_fin)
```

- For each class, the current "picture" of assets is stored in a data.frame.

```
# Bond portfolio
canton@ptf_fin@ptf_oblig
```

- To calculate cash-flows and market values for bonds

```
# Coupons and terminal cash-flows
calc_flux_annee(canton@ptf_fin@ptf_oblig)
# Market value
calc_vm_oblig(canton@ptf_fin@ptf_oblig,
canton@mp_esg@yield_curve)
```

Canton forecasting

- A Canton can be very easily projected over 1 year:
  - Asset and liabilities cash-flows,
  - Apply ALM and profit sharing rules,
  - Compute P&L and other balance sheet items,
  - Set the value of an updated Canton.

```
result_proj_an <- proj_an(canton, nb_annee,
 pre_on = FALSE)
canton_updated <- result_proj_an[["canton"]]
# Extract cash flows by product
result_proj_an[["output_produit"]][["flux"]]
# Extract financial results
result_proj_an[["result_fin"]]
```

Best estimate calculation

- A best estimate objet is defined with a Canton and ESG tables

```
class(be)
[1] "Be"
# The initialized Canton
be@canton
# ESG tables
be@esg
```

- Compute Monte-Carlo simulations

```
# To run simulation #10
run_be_simu(be, 10L, pre_on = F)
# To run all the simulations
be_results <- run_be(be, pre_on = F)
# Extract the amount of best estimate
be_results[["be"]]@tab_be
# Extract the average cash-flows
be_results[["be"]]@tab_flux
```

Performances

- Some functions are developed using the library `Rccp`.
- The function `run_be`() can be speed up and allows parallel computing with the package `doParallel`.
- Our performances are rather good with `R` !

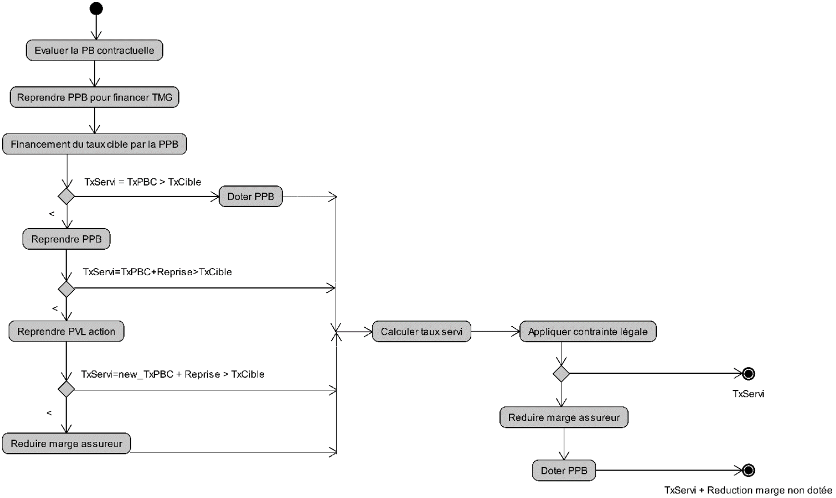
Table: Performances with 1,000 simulations in minutes by using 1 core (Intel Core I7-5500U 2.40GHz) and 8 GB RAM.

Number of model points for each Asset class	Number of Saving and Retirement model points (for each)	Computation time
100	100	13.2
100	1000	14.0
100	10000	24.7
1000	100	13.8
1000	1000	15.0
1000	10000	26.1

Perspectives

- Increase performances.
- Include UL products and contract an ohter specificities.
- Add new asset classes (e.g. floating rate bonds).
- Develop a toolkit with indicators for the results analysis.
- Take the inputs from the user with more security.
- Extend our group of developers.

Profit sharing algorithm



References I

Bacinello, A. (2001). Fair Pricing of Life Insurance Participating Policies with a Minimum Interest Rate Guaranteed. *ASTIN Bulletin*, **31**(2), 275–297.

Ballotta, L., Haberman, S., and Wang, N. (2006). Guarantees in With-Profit and Unitized With-Profit Life Insurance Contracts: Fair Valuation Problem in Presence of the Default Option. *Journal of Risk and Insurance*, **73**(1), 97–121.

Bauer, D., Kiesel, R., Kling, A., and Ruß, J. (2006). Risk-neutral valuation of participating life insurance contracts. *Insurance: Mathematics and Economics*, **39**(2), 171–183.

Borel-Mathurin, F., Darpeix, P.-E., Guibert, Q., and Loisel, S. (2015). Main Determinants of Profit Sharing Policy in the French Life Insurance Industry. PSE Working Papers 2015-16, Paris School of Economics, Paris.

Eling, M. and Kochanski, M. (2013). Research on lapse in life insurance: what has been done and what needs to be done? *Journal of Risk Finance*, **14**(4), 392–413.

Grosen, A. and Løchte Jørgensen, P. (2000). Fair valuation of life insurance liabilities: The impact of interest rate guarantees, surrender options, and bonus policies. *Insurance: Mathematics and Economics*, **26**(1), 37–57.

Hainaut, D. (2009). Profit sharing: a stochastic control approach. *Bulletin Français d'Actuariat*, **9**(18), 65–78.

References II

Kemp, M. (2009). *Market Consistency: Model Calibration in Imperfect Markets*. Wiley, 1 edition.

Kling, A., Richter, A., and Ruß, J. (2007). The interaction of guarantees, surplus distribution, and asset allocation in with-profit life insurance policies. *Insurance: Mathematics and Economics*, **40**(1), 164–178.

Laurent, J.-P., Norberg, R., and Planchet, F., editors (2016). *Modelling in Life Insurance – A Management Perspective*. EAA Series. Springer International Publishing.

Milhaud, X., Loisel, S., and Maume-Deschamps, V. (2011). Surrender triggers in life insurance: what main features affect the surrender behavior in a classical economic context? *Bulletin Français d'Actuariat*, **11**(22), 5–48.

Planchet, F. and Thérond, P.-E. (2007). Allocation d'actifs selon le critère de maximisation des fonds propres économiques en assurance non-vie : présentation et mise en oeuvre dans la réglementation française et dans un référentiel de type Solvabilité 2. *Bulletin Français d'Actuariat*, **7**(13), 10–38.

Vedani, J., Karoui, N. E., Loisel, S., and Prigent, J.-L. (2017). Market inconsistencies of market-consistent European life insurance economic valuations: pitfalls and practical solutions. *European Actuarial Journal*, pages 1–28.