Directorate Banknotes

User manual for the ECB Circulation Model for banknotes

Harald Deinhammer

For Circulation Model version 10

26 February 2018

Revision history of this manual

Revision	Date	Comment
1	19 May 2015	First version for Circulation Model V3
2	28 August 2015	Updated version for V4
3	6 June 2017	Updated version for V8.0.1
4	19 June 2017	Correction of errata, update for V8.0.2
5	9 August 2017	Update for V9
6	26 February 2018	Update for V10

Version history of the circulation model

Version	Date	Comment
1	19 January 2015	First release for Eurosystem discussion
1.1	22 January 2015	Better calculation of sensor threshold → more stable steady-state
2	28 January 2015	Integrate processing costs & defects in model Review of algorithm for constant quality target
3	19 May 2015	Included annual change in note volume, hoarding, negative net issuance, different ageing functions, simple and expert mode, recirculation.
4	28 August 2015	All calculations to VBA, New defect handling, batch processing
5-8	16 May 2017	Bug fixes, removal of need for Excel Solver, UDF function, better error handling.
8.0.1	6 June 2017	Bugfix for storing of note profiles
8.0.2	19 June 2017	Formatting fixes, fix for weekly costs on '2 Cases' sheet.
8.0.3	26 June 2017	Recirculation threshold can now be set to any value.
9	9 August 2017	Correction of inflows. Output of costs for circulation increase.
10.0.5	23 February 2018	Rearrangement of outputs and worksheets. Handling of notes put on stock at NCBs in case of countries with large inflows.

Contact for comments, critique, bugs: harald.deinhammer@ecb.europa.eu

N.B.: This manual refers to entries in the Circulation model by the naming in column A of the main sheets of the model[1] and quotes these values in brackets (i.e. [F1] relates to the NCB sorting costs).

The model is available for public download as Annex to ECB Occasional Paper 204 at:

http://www.ecb.europa.eu/pub/research/occasional-papers/html/papers-2017.en.html

---

[1] The sheets *'Main', 'WhatIf', and' Variability''*

Table of contents

1. Introduction

The Circulation Model has been developed by the ECB's Directorate Banknotes in close consultation with the NCB members of the Banknote Sorting Task Force (BSTF).

The aim of the model is to provide a better understanding on the impact of the different parameters in a banknote cash cycle on e.g. note quality or note replacement rates, with a special focus on the impact of NCB sorting policies.

This manual explains how to operate the model (Section 2) as well as the algorithm behind (Section 3). Furthermore it describes the different input parameters (Section 4) and explains the results (Section 5). Section 6 provides a step-by-step example of a single calculation step of the model, whereas in Section 7 the use of the more advanced modelling functions (*'WhatIf'* and *'Variability'*) is explained. Dedicated sections on known issues and tips and tricks can be found at the end.

2. How to use the model

2.1 Opening the model

The model is available as an Excel file. It is strongly recommended that you copy the file containing first to your local hard-drive before opening it. When the model is opened for the first time, you will be asked by Excel to turn on the included macros; this needs to be accepted as otherwise the model will not function.

Each time when opening the model, a first initialisation will take place which should take no longer than 2-3 seconds. Afterwards you will receive the following message:
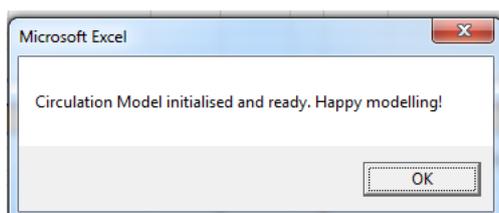
2.2 Basic principles

The model is implemented as a function (called *'CalcCirculationModel')* in Excel VBA and can be run via different worksheets. I.e. all the different worksheets use the same algorithms but just differ in the way the inputs are provided and – more so – by the way the results are displayed.

The most simple way to run the model is via the worksheet *'Main'* which provides access to all model functions[2] and also displays the results in a graphical format. The function of this sheet is to directly compare two cash-cycles to each other.

Additional functions which are useful to be accessed from any worksheet of the model have been put in a dedicated ribbon called 'Circulation Model' as can be seen in figure 1 below.
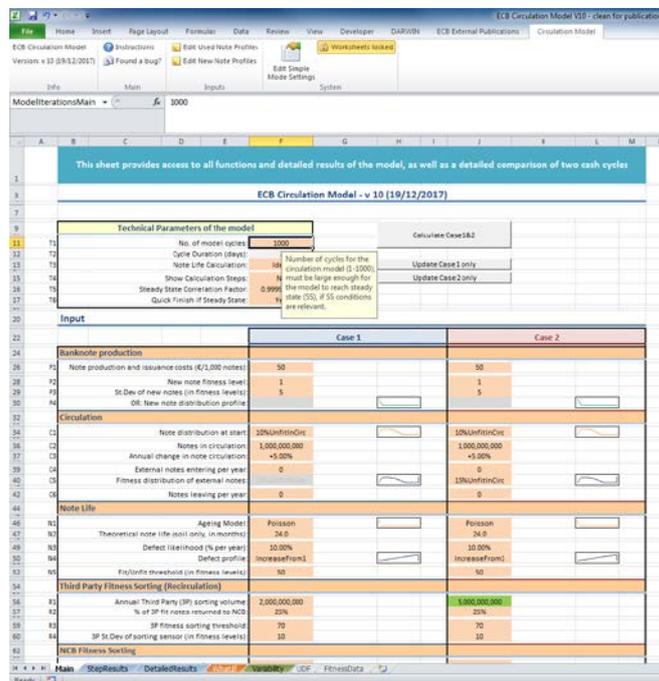

Figure 1: Sheet 'Main' of the Circulation Model with 'Circulation Model' ribbon shown

---

[2] Only the parameter [T2] Cycle duration is fixed here as 7 days (i.e. weekly cycles) have proven to deliver best results at a reasonable amount of calculation time. This parameter can be changed on the 'WhatIf' and 'Variability' if needed.

2.3 Description of the functions in the 'Circulation Model' ribbon

Instructions: Opens this document (works only if you are connected to the ESCB network and have DARWIN access, otherwise this manual is part of the public downloadable package).

Found a bug?: Opens an empty email to the author for providing feedback.

Edit Used Note Profile: Jumps to the sheet 'FitnessData', where you can edit the profiles of used notes which shall be available to the model as starting conditions. You can add or overwriting existing profiles with other profiles.

Edit New Note Profile: Same as the previous command bringing you to the range containing the profiles for new notes used by the model (these profiles are used in [P4], if [P2] and [P3] are not specified).

Edit Simple Mode Settings: This mode of operation is no longer supported in V10. The button has no more function assigned.

Worksheets locked: Shows all hidden sheets and removes any cell protection in order to study the model. Proceed with care!
In detail, this function un-hides the following additional worksheets:

'ResultsCM': Shows detailed results and fitness profiles of each calculation cycle when the model is run from the 'Main' worksheet.

'ResultsWI': Shows detailed results and fitness profiles of each calculation cycle when the model is run from the 'WhatIf' worksheet.

'Results': Shows detailed results and fitness profiles of each calculation cycle when the model is run from any other worksheet.

'Interpolation': Calculates weekly results for display in the results charts on the 'Main' sheet via interpolation irrespective of the calculation cycle duration given in [T2].

'Options': Some settings for the ribbon and dropdown menus.

2.4 Description of the buttons in the 'Main' sheet
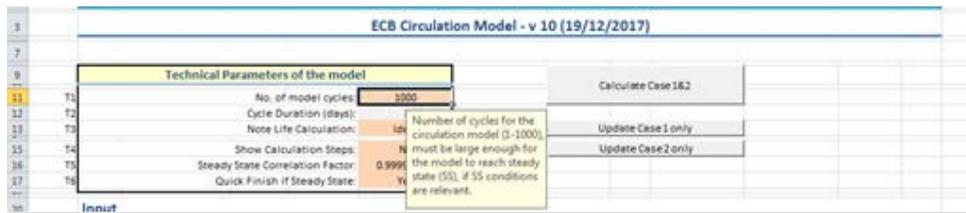
Calculate Case 1&2: Calculates both model cases and updates the results.

Update Case1/2 only: Calculates only Case1/2, leaving the other case untouched and updates the comparison figures in the Results section of the sheet.

Store Case1/2 Circ. Profile: Stores the note profile in circulation after the last run (i.e. the final steady state fitness profile in circulation shown in the results section) of either Case1 or Case2 into one of the 10 available slots of the starting profile [C1]. You need to provide the slot of the profile and a new name for the profile. If no name is specified, the existing one is kept.
N.B.: After storing a profile under a new name, the name in [C1] for the specific case is cleared and needs to be selected again.

Copy Case 1/2 Inputs to WhatIf/Variability: Takes the inputs and copies them as inputs to the 'WhatIf' or 'Variability' sheet; useful if you have set up a cash-cycle and then want to take the input cases and study how they react to changes.

2.5 Description of the layout of the 'Main' sheet

The worksheet is set up to study the impact of various parameters to the sorting and ageing of notes by comparing two cases. The model inputs are at the top, whereas at the bottom of the sheet the results are displayed in different sections, as can be seen in the layout below.
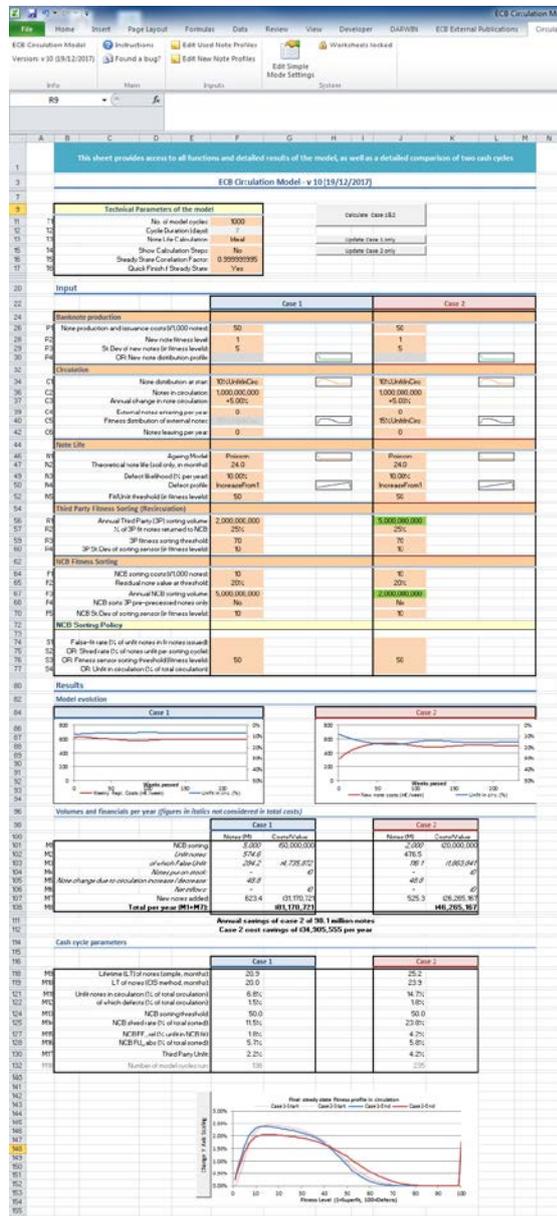
Figure 2: Overview of the various sections of the 'Main' sheet of the circulation model

In addition the sheet 'StepResults' shows the results after each model cycle (7 days) and the sheet 'DetailedResults' provides the detailed note balances per model cycle as well as multiple additional charts.

3. Circulation Model Algorithm

3.1 Algorithm flowchart

The chart below shows the basic flow of the algorithm and the different (sorting, mixing, ageing) steps applied, as well as the naming of the various banknote flows (V). Details on the various stages are found also in Chapter 4 (Inputs) and an exemplary calculation step with actual results and figures is given in Chapter 6.
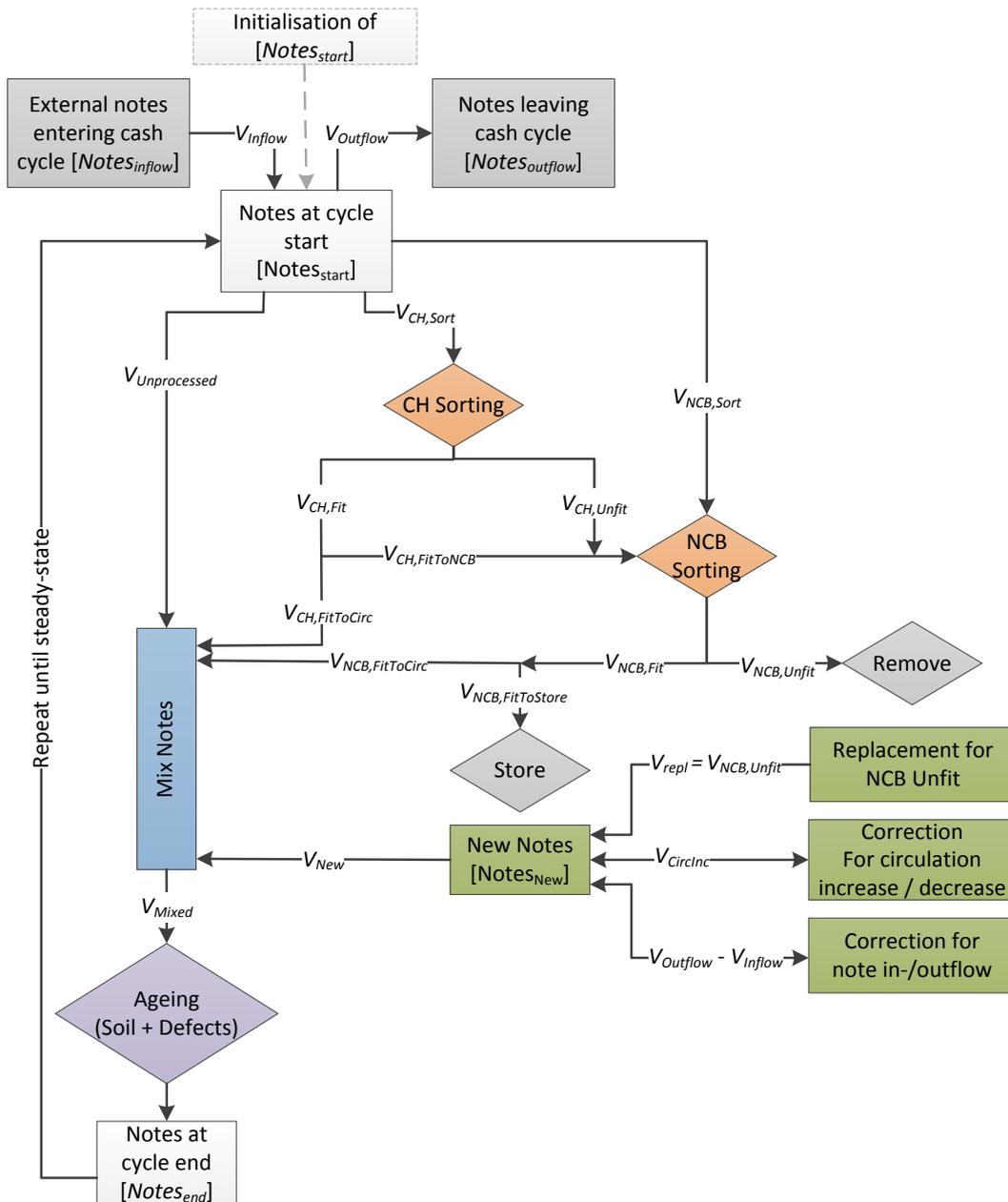
Figure 3: Algorithm flowchart of the circulation model

3.2 High-level flowchart description

1) The model takes as a starting point the fitness distribution of notes in circulation (Notes$_{Start}$)

2) The notes are first subjected to a recirculation sorting step *(CH Sorting)*[3].

3) The notes are then sorted by the central bank *(NCB Sorting)*. The profile of notes sorted by the NCB is a mix of the notes from circulation *(V$_{NCB,Sort}$)*, the recirculation unfit *(V$_{CH,Unfit}$)* and the excess recirculation fit notes sent back to the NCB (V$_{CH,FitToNCB}$).

4) Notes sorted to unfit by an NCB *(V$_{NCB,Unfit}$)* are replaced with notes specified in the profile (Notes$_{New}$)[4].

5) Additional notes with profile Notes$_{New}$ are added to the circulation (or the volume of new notes added is reduced) to simulate annual circulation growth/decrease *(V$_{CircInc}$)* and to compensate for any in- or outflows *(V$_{Outflow}$ − V$_{Inflow}$)*, resulting in the total volume of new notes added V$_{New}$.

6) All notes, i.e.

   [1] the notes not processed in this cycle *(V$_{Unprocessed}$)*,
   [2] the notes sorted to fit by cash handlers which are not returned to the NCB *(V$_{CH,FitToCirc}$)*,
   [3] the fit notes from the central bank which are not put into storage *(V$_{NCB,FitToCirc}$)*, as well as
   [4] any new notes *(V$_{New}$ = V$_{NCB,Unfit}$ + V$_{CircInc}$ + V$_{Outflow}$ − V$_{Inflow}$)*,

   are mixed together resulting in V$_{Mixed}$.

   N.B.: Since version 10 of the circulation model also cash-cycles where the NCB cannot return all fit notes back to circulation can be simulated (This is e.g. the case if the national cash cycle has large inflows or the annual circulation volume is markedly decreasing). In such case, any excess notes are stored (i.e. the volume of V$_{NCB,Fit}$ which is normally put back into circulation is reduced to V$_{NCB,FitToCirc}$ and V$_{New}$ = 0 in such case).

7) The mixed notes V$_{Mixed}$ are aged by separate functions simulating soiling and defects.

8) The resulting note profile Notes$_{End}$ is compared to Notes$_{Start}$. If the correlation between the two profiles is higher than the specified correlation coefficient, the steady-state condition has been reached and the final results are displayed to the user. Otherwise Notes$_{Start}$ is set to Notes$_{End}$ and steps 1-7 are repeated.

---

[3] CH denoting Cash Handlers

[4] Notes$_{New}$ can be any note profile and can therefore not only represent brand new notes but also fit notes from an NCB's storage.

3.3 Basic modelling concepts

For a detailed explanation of one model cycle please refer to chapter 6.

3.3.1 Fitness levels

1) Any banknote distribution (e.g. $Notes_{Start}$, being the starting distribution of the notes in circulation) is characterised as a one-dimensional array with 100 fitness levels (1 = Superfit, 100 = Superunfit and Defect notes). The array contains the normalised share of notes for each fitness level.

2) The fitness levels are homogenously spaced with regards to sensor performance and ageing. This means that e.g. a note ages for an identical amount of fitness levels in each model cycle irrespective of its current fitness level. It also means that the NCB or recirculation sensor inaccuracy/performance in determining fit or unfit is identical across the whole fitness range[5].

3.3.2 Ageing of a banknote

3) Within each model cycle the notes age according to a selectable distribution function [N1] for an average number of soil levels ([L12] on the 'DetailedResults' sheet). [L12] is calculated by the theoretical note lifetime [N2], the cycle duration [T2], the quality of the new notes [P2, P3 and P4] and the Guideline fit/unfit threshold [N5].

4) In addition to this ageing, a banknote has a certain likelihood (expressed as percentage per year) to suffer a defect [N3] which immediately moves the note to fitness level 100 (defect plus heavily soiled notes), irrespective of its current fitness level.

5) Whereas the ageing is applied homogenously across all fitness levels, the defect likelihood [N3] is applied to the banknote distribution according to a selectable profile [N4] in relation to the notes soil level. By this approach it can e.g. be modelled that banknotes with a higher soil value have a higher likelihood of becoming defective.

3.3.3 In- and outflows of banknotes

6) The model allows the simulation of banknote inflows (where the volume [C4] and fitness profile [C5] need to be specified), as well as banknote outflows (where only the volume [C6] needs to be

---

[5] It is known from Eurosystem internal studies that the performance of a fitness sensor is not constant over the fitness range, however, usually fitness sensors operate at a certain working point at which the performance is then constant. For modelling of cash cycles the respective inaccuracy of the sensor at the set or expected working point should be used.

specified, as the fitness profile of the banknote outflows is assumed to be identical to the current circulation in the modelled cash cycle).

7) In- or outflows have no impact on the number of notes in circulation specified in [C2]. For outflows this is achieved by the missing notes being replaced in each modelling cycle with additional notes issued by the NCB having the $Notes_{New}$ profile. For inflows the respective NCB replacement volume with new notes in each cycle is reduced accordingly (see also the note in item 6) of Chapter 3.2.

3.3.4   Circulation volume increase

8) The number of new notes which needs to be added to cover for the general increase of notes in circulation or the reduction in new notes due to a decrease in the note volume (specified in [C3]) is applied in every cycle by an according increase or decrease of V_{New}. However, at the end of each cycle the note volume V_{Mixed} is normalised to the circulation figure provided in [C2]. This is so that the model can, at fixed NCB sorting volumes [F3] and third party sorting volumes [R1] reach a steady state (otherwise, at a constant sorting volume but changing total circulation the model would never reach a steady state[6]). As a result all modelling results are normalised to the starting conditions and different cases can also be compared more easily to each other.

N.B.: While the model takes into account the effect of note volume increase and decrease on the note profile, the costs output by the model are ALWAYS the costs of the cash cycle having the circulation and processing volume as specified at the starting point! By this approach the comparison of multiple cases is ensured, irrespective of the cycles/time needed to reach the steady-state.

3.3.5   Cost of replacement and false unfit banknotes

9) Banknotes lose value from fitness level 1 (where they have a value of [P1]) to the point where they are considered unfit according to NCB standards (i.e. the fit/unfit threshold [N5]); at this threshold they have a residual value of [F2], which is expressed as a percentage of [P1]. This value corresponds to the residual value of a note being fit to be issued once more.

10) When calculating the cost of false-unfit notes (fit notes shredded prematurely), only the residual value of each note is taken and not the full price of a new note.

---

[6] E.g. at an annual note volume increase of 10% at constant NCB sorting volume, the quality in circulation would constantly degrade.

3.3.6   Sorting of banknotes

11) The NCB and recirculation fitness sensors sort notes at a defined threshold (specified for recirculation in [R3] and for the NCB via one of the parameters [S1] to [S4]). These sensors have an inaccuracy in classifying notes into fit or unfit, which results in False Fit (unfit notes being re-issued) and False Unfit (fit notes prematurely destroyed by NCB or prematurely returned to NCB by the recirculation entity) notes. This inaccuracy is modelled by a normal distribution and expressed as standard deviation with regards to the 1-100 fitness levels (parameters [R4] and [F5]).

12) Notes with fitness level 100 are treated as defects and are always sorted to unfit by an NCB or recirculation sensor, irrespective of sensor threshold and sensor accuracy.

3.3.7   NCB Sorting policies

13) NCB sorting can be done in one of four modes (policies), depending on which of the parameters ([S1]-[S4]) is provided. These modes are:

   o Constant Sensor Threshold [S1]: The sensor threshold is left at a constant setting irrespective of the incoming note quality at the NCB.

   o Constant False Fit Rate [S2]: The sensor threshold at NCB sorting is in each cycle adjusted so that the notes coming from the sorting machine will contain the given amount of False Fit banknotes, if possible[7] (With this mode the quality of the notes issued by the NCB is kept constant irrespective of the input).

   o Constant Shred Rate [S3]: The sensor threshold is adjusted so that the shred rate (% of notes sorted to unfit by the NCB) is kept constant irrespective of the input note quality. As a result, the note quality reissued is directly related to the note quality sorted by the NCB.

   o Constant Quality in Circulation [S4]: The sensor threshold is adjusted at each cycle to issue notes of a quality that will best achieve the set amount of unfit in circulation. E.g if the quality in circulation is better than the desired amount the NCB threshold will be set very low and vice versa (The final quality in circulation is reached only gradually over multiple iterations)[8].

---

[7] E.g. if the input to the sorting machine contains only 5% unfit, it is impossible to issue a note population with 8% False Fit.

[8] This would require an NCB to constantly and accurately know the quality of banknotes in circulation, which in reality is not the case.

N.B.: For this sorting mode, in case the starting conditions are far from the required quality level, the model could become unstable and start to oscillate. Use this policy only with a starting distribution close to the desired quality in circulation.

In reality most NCBs are sorting with a constant sensor threshold [S1], especially as it is difficult to apply a sorting policy that changes with the note quality sorted at the NCB (e.g. false-fit rate) or is difficult to determine (quality in circulation).

The aim of the additional policies is primarily to show their impact on note quality and to check whether they could be useful to apply in case of intended changes to a cash-cycle.

4. Inputs on sheet 'Main'

This section describes the various model inputs. Their description is also available as a tool-tip on most sheets by selecting the respective field.

Figure 4: Example tool-tip

4.1 Technical parameters of the model

4.1.1 T1 - No. of model cycles

Options: 1 – 1,000 (Integer)

Function: Maximum number of cycles the model will be run, must be large enough for the model to reach steady state (SS), if SS condition is relevant.

Use: This value defines how often, as a maximum, a single model cycle as described in chapter 3.2 above is carried out. One cycle has a duration (in days) specified in [T2] below.

N.B.: See also Tips and Tricks on how to quickly reach steady state conditions.

4.1.2 T2 - Cycle Duration (days)

Options: 7 - 60 (Days passed in each cycle, Integer[9]), set fixed to 7 on the 'Main' sheet.

Function: Defines the length of one model cycle in days, i.e. after how many days [1] the progress is re-evaluated, [2] the model profiles updated and [3] checked if steady-state has been reached. Within the overall accuracy of the modelling the setting only has a small impact on the final results.

---

[9] Via the UDF function also decimal values can be set.

Use: Suggested setting: 7 days (fixed on 'Main' sheet), other settings possible on the 'WhatIf' and 'Variability' sheet, as well as via the UDF function.

4.1.3 T3 - Note Life Calculation

Options: Ideal, Real

Function: Selects one of two methods to calculate the average note ageing per cycle, based on the cycle duration [T2], theoretical note life (soil only) [N2], the Fit/Unfit Threshold [N5] and the new note distribution [P2] - [P4].

Ideal: Note life is calculated from fitness level 1 (Superfit) to the Fit/Unfit Threshold [N5]

Real: Note life is calculated from the average fitness level of new notes (specified in [P2]-[P4]) to the Fit / Unfit Threshold [N5].

Example

Banknotes have an average life (soil only) of [N2] 12 months (i.e. 360 days), Cycle duration [T2] is set to 7 days. The new notes on average have, due to production variations, a fitness level of 10 and the Guideline fit/unfit threshold [N5] is at 50:

Results: Ideal Note ageing: (50–1) / (360/7) = 0.953 fitness levels per cycle

            Real Note ageing: (50-10) / (360/7) = 0.778 fitness levels per cycle

---

Use: If the same new note profile is selected for both cases in [P2] – [P4], 'Real' shall be selected for correct results. In case the influence of different new note qualities or production variation is studied then 'Ideal' needs to be selected, as otherwise the difference in the starting quality (average fitness level) will be cancelled out by the ageing calculation.

4.1.4 T4 - Show Calculation Steps

Options: Yes, No, Yes incl. charts

Function: Yes: Shows the model results per cycle on the 'StepResults' sheet, but does not create sparkline charts

           Yes incl. Charts: Same as above, also showing sparkline charts

No: Hides the output of any interim results but shows the results per cycle at the end similar to 'Yes'.

Use: To significantly speed up calculation.

Example calculation times: No: 0.8 sec. - Yes: 20sec - Yes incl. charts: 40 sec. Showing the calculation steps and the charts in each cycle can initially be useful to learn how the model/ note profile evolves towards steady-state.

4.1.5   T5 - Steady State Correlation Factor

Options: 0.9 – 1 (Decimal)

Function: After each cycle, the model compares the resulting profile of notes in circulation (Notes$_{End}$) with the starting profile (Notes$_{Start}$). If the correlation coefficient (using the Excel CORREL function) between the two profiles is larger than the value specified, then steady state is assumed and - if [T6] is set to 'Yes' - then the calculation is stopped.

Use: To stop calculation once steady state is reached.

4.1.6   T6 - Quick Finish if Steady State

Options: Yes, No

Function: Once the correlation coefficient is larger than specified in [T5], the calculation is stopped

Use: To stop calculation once SS is reached (as additional calculation cycles will no longer change the result).

4.2   Production parameters of the model

4.2.1   P1 - BN production and issuance costs

Options: 0 - 500 (EUR/1,000 banknotes)

Function: Cost for new banknotes of the respective denomination in EUR/1,000 notes. Shall be the total costs, including costs for issuance in order to evaluate the specific case correctly.

Use: This value is used for calculating the replacement costs for new notes and for the residual value of aged note that are sorted to unfit at the NCB (i.e. shredded) prematurely, taking into account also the value specified in [F2].

4.2.2 P2 - P4 – Specification of the new note profile

In order to simulate production variations, but also the case of issuance of used notes by the NCB as replacement for unfit, the fitness profile of new notes which are used in the model [1] to replace notes going to unfit, [2] need to be issued due to increase of the circulation volume or [3] due to in- and outflows can be specified here. This can be done by specifying a normal distribution providing the average fitness level [P2] and the standard deviation [P3] (in fitness levels). Alternatively any new note profile can be specified in the sheet 'FitnessData' from Column S onwards and selected in [P4]. If [P4] is not empty, then the values specified in [P2] and [P3] will be ignored.

Parameter: P2 - New Note fitness level

Options: 1 – 99 (Decimal)

Function: The average fitness level of a new note.

Parameter: P3 - St. Dev of New Notes

Options: 0 - 100 (Decimal)

Function: The standard deviation (in fitness levels) of new notes issued (i.e. their production variation).

N.B.: In case P2 and P3 are specified, the not useable part of the distribution (i.e. entries that would have negative fitness levels or a fitness level of 0 due to the normal distribution function) will be ignored and the positive part will be normalised.

Parameter: P4: New Note distribution

Options: Any profile in sheet 'FitnessData' from column P onwards. If left blank (by deleting any entry in the field using the 'DEL' key) then [P2] and [P3] are used. The profile provided in the sheet 'FitnessData' does not need to be normalised (will be done by the model).

4.3 Circulation parameters of the model

4.3.1 C1 - Note distribution at start

Options: Any profile specified in sheet 'FitnessData' from column C to L can be selected. The profile provided does not need to be normalised (will be done by the model).

Function: The distribution profile of the notes in circulation which is used as input in the first sorting cycle. New profiles can be added in the sheet 'FitnessData' in columns C to L, or results of the model can be saved as a new fitness distribution using the 'Store Case 1/2 circ. profile' buttons at the bottom section of the 'Main' sheet.

Use: Relevant for how quickly a steady-state is reached or for dynamic modelling exercises where the evolution of a current note profile in circulation shall be studied. The closer this starting profile is to the steady state, the fewer cycles are required to reach the final results. In other cases where the SS condition is not relevant, but where the evolution of the circulation over time is interesting (e.g. studying the issuance of a new series), a different starting profile (e.g. 'New Notes') can be selected and the results of each model cycle can be studied to see how the circulation changes over time.

4.3.2   C2 - Notes in circulation

Options: Larger than 0 (Integer)

Function: Amount of banknotes in circulation for the investigated cash cycle.

Use: Typical input is the number of active notes in circulation per denomination. Notes which are known to be outside of the respective cash-cycle, lost, destroyed or hoarded should be excluded.

4.3.3   C3 - Annual change in note circulation

Options: -100% to +100%

Function: Annual increase or decrease of circulation volume due to demand changes. Will be considered for note quality and sorting parameters, BUT NOT for overall replacement costs[10] in order to allow a neutral comparison between two cases with different note growth rates. The costs/savings due to circulation increase/decrease are output for information in the results [M5]. All costs given are - for comparison - given based on the starting volume provided in [C2] (See also chapter 3.3.4).

Use: An increase in the note volume in circulation has a 'cleaning effect' on the overall circulation as more new banknotes are added to the cash cycle, making the overall

---

[10] In the charts for the evolution of the two cases in the Results section of the 'Main' sheet, as well as in the 'StepResults' sheet the weekly costs include all costs including growth/decrease and in-/outflows.

quality better; a decrease in the note volume has a contrary effect, as no new notes can be issued for some of the unfit notes taken out of circulation.

4.3.4   C4 – External notes entering per year

Options: Larger than or equal to 0 (Integer)

Function: The number of notes per year entering a cash-cycle from outside having the profile specified in [C5]. As the note volume in circulation is kept constant by the model, an inflow of notes is compensated by an according reduction of new notes added in each cycle.

Use: To simulate flows of notes entering a national cash cycle (e.g. for countries having a negative net-issuance, see also section 3.3.3).

4.3.5   C5 - Fitness distribution of external notes

Options: Any profile specified in sheet 'FitnessData' from column AD to AM can be selected. The profile provided does not need to be normalised (will be done by the model).

Function: The distribution profile applied to the number of notes entering the cash cycle specified in [C4].

Use: Used to model the impact of inflows of (cleaner or more soiled) notes into a national cash cycle.

4.3.6   C6 – Notes leaving per year

Options: Larger than or equal to 0 (Integer)

Function: The number of notes per year leaving a cash-cycle. The notes leaving are assumed to have the fitness profile of the current circulation. As the note volume in circulation is kept constant by the model, an outflow of notes is compensated by an according increase of new notes added in each cycle.

Use: To simulate flows of notes leaving a certain cash cycle (e.g. for countries having a positive net-issuance, see also section 3.3.3).

4.4   Note Life parameters of the model

4.4.1   N1 - Ageing Model

Options: List of pre-defined ageing functions

Function: Defines which ageing function is applied to the notes in circulation. The following functions are available (using the built in Excel functions to generate the distributions)[11]:

Poisson: Poisson_Dist with mean = Average Ageing per cycle

Exponential: Expon_Dist with lambda = f(Average Ageing per cycle)

LogNormal 0.8: LogNorm_Dist with mean = f(Average Ageing per cycle)
standard_dev = 0.8

Gamma a1/a9/a10/: Gamma_Dist with   alpha = 1, 9 or 10 and
beta = f(Average Ageing per cycle)

Weibull k1.5: Weibull_Dist with   alpha = 1.5
beta = f(Average Ageing per cycle)

Use: The notes age on average, per cycle according to the calculations shown in 4.1.3 (T3 - Note Life Calculation, see also the example there). This average amount of loss of fitness levels per cycle can be achieved according to various distributions as shown in the figure below[12]. The ageing model selected here defines the distribution of notes which age less or more than the average and sets the according profile, which is applied in each cycle to the circulation notes.

N.B.: The exact ageing model has shown to have only a limited impact on the model results. Poisson ageing is suggested as an approximation for the average cash cycle.

---

[11] The model requires discrete values for the ageing probability per increase in fitness level. The only implemented function having a discrete distribution is the Poisson distribution. For all other functions the required parameter to achieve the average ageing per cycle defined as described in section 4.1.3 is determined via a linear approximation algorithm; the mentioned shape parameter of the distribution function is defined so that the resulting curve has - in its discrete form - the defined ageing per cycle.

[12] As an extreme example, if notes on average lose 5 fitness levels per cycle, this can be 'achieved' by all notes losing exactly 5 fitness levels or e.g. 75% of the notes not ageing at all and 25% of the notes ageing by 20 levels.
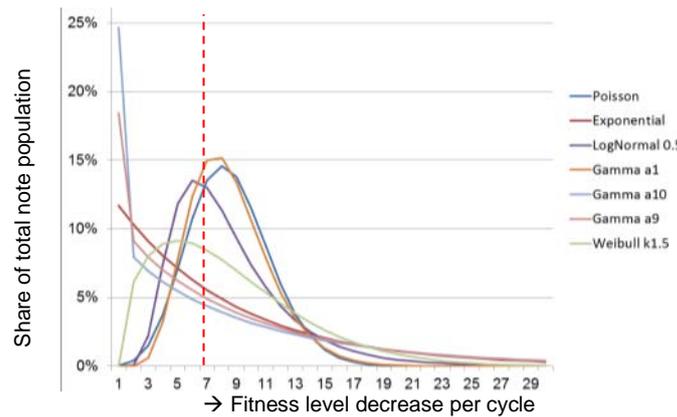
Figure 5: Various ageing distribution functions, all having an Average Ageing per cycle of 7.5 fitness levels.

4.4.2 N2 - Theoretical note life (soil only)

Options: Between 1 and 1,000 (months)

Function: Average lifetime of a note until it reaches the Guideline Fit-Unfit Threshold [N5], <u>if it does not suffer a defect</u>.

Use: This is the <u>theoretical</u> average lifetime of a note from its first issuance to the fit/unfit threshold. This figure does not take into account when the note will actually be sorted out (this is determined by the NCB sorting threshold which could be more or less severe than the general defined fitness threshold, but also the return time to the NCB as well as in- and outflows), or whether a note has prematurely become unfit due to suffering a defect. The <u>actual lifetime</u> of a note, taking also into account NCB sorting policies and defects is given as a result in [M9] and [M10].

N.B.: For modelling a real cash cycle, these two model result values should match a central bank's actual.

4.4.3 N3 –Defect likelihood / Year

Options: 0 - 100,000 %

Function: The likelihood in % per year that a note in circulation will suffer a defect (tear, tape, hole, graffiti ...) that - when found either during NCB sorting or third party sorting - will result in the note being sorted to unfit irrespective of the fitness level it had prior to becoming defective. This ageing is applied in each cycle ON TOP of the theoretical note life defined in [N2]. This factor is applied to the circulation note profile, weighted across the fitness

levels using the defect profile specified in [N4]. Defective notes will be assigned the fitness level 100 (super-unfit) and sorted to unfit at the next time the note is sorted by a third party or the central bank.

Use: To model defects (together with the profile [N4] below).

4.4.4 N4 – Defect profile name

Options: List of pre-defined defect profiles

Function: To model that the likelihood of a note becoming defective is related to its overall fitness level. Any profile can be specified in the sheet *'FitnessData'* from column AQ onwards. The selected profile is weighting the total number of defects which are applied to all notes (specified by [N3] above) across the note distribution in circulation.

Use: To model defects, together with the parameter [N3] above.

On the modelling of defects

The approach implemented in the model is based on the real-life experience that typically defect notes removed from circulation are already soiled, i.e. the ageing of a note makes it more likely for them to become e.g. torn or have tapes applied to them.

Below 3 examples of applying different profiles are shown:

'Increase from 1' Profile:

The probability of a note to become defective is increased with its soil level.

N.B.: With this approach good matches to real-life quality data were found for most cash cycles.

'Constant' Profile:

All notes in circulation have the same probability to suffer a defect, except those already having a fitness level of 100 (i.e. are defective).

'Increase 50-75' Profile:

In this case notes below 50 soil levels never become defect and notes above 75 have the maximum likelihood to become defective within a year. Notes between 50 and 75 fitness levels have a linearly increasing probability to become defective.

4.4.5 N5 - Fit/Unfit Threshold

Options: 1 - 100 (integer)

Function: The fitness threshold defined by the central bank. Notes above this threshold are unfit according to a defined guideline, notes at or below threshold are fit according to the guideline.

N.B.: <u>This is the fitness level of a 'limit banknote' and not necessarily the NCB sorting threshold</u>. An NCB can sort either at this threshold but also more or less severely to adjust the circulation quality as needed.

Increasing this parameter means to apply in general a more lenient fitness standard and has widespread impact on the model results. Normally this is set to 50, only adjust if necessary.

Use: To define the 'true' fit/unfit threshold of a banknote.

4.5 Third Party Fitness Sorting (Recirculation) parameters of the model

Sorting by third parties is modelled similar to NCB fitness sorting at a fixed threshold, with the third parties using also fitness sensors that have normal distributed errors. All notes sorted as unfit by the third party (whether correct unfit or false unfit) are returned to the NCB and processed there. In addition, as is the case for some cash cycles, a certain percentage of notes sorted to fit by third parties can be defined to be returned to the NCB. This is to model excess fit stocks of third parties which are returned to the NCB.

4.5.1 R1 – Annual Third Party sorting volume

Options: Any non-negative integer

Function: Number of notes processed by third parties (can be 0).

Use: For modelling the recirculation done by third parties, put to zero if no recirculation shall be modelled, in which case [R2], [R3] and [R4] are greyed out and not used.

4.5.2 R2 – % of Third Party Fit Notes returned to NCB

Options: 0 - 100 %

Function: The percentage of fit notes which (in each model cycle) are returned by third parties to the central bank for sorting.

Use: To model the impact of these notes when processed by an NCB.

A word on the note balance

The number of notes sorted at the NCB [F3] normally consists of (see also figure 3):

            100% of Third Party Unfit notes

+   [R2]% of Third Party Fit notes

+   Notes from Circulation

In each cycle the number of notes from circulation is adjusted, so that the total NCB sorting volume sums up to [F3]. In case the model should reach a stage where the number of Third Party Unfit + Third Party Fit to NCB is larger than the NCB sorting volume, the calculation is aborted and an error is displayed. In this case either:

    - The NCB sorting volume needs to be increased

    - [R2] needs to be reduced

    - The number of third party unfit notes needs to be reduced (e.g. by more lenient sorting)

Alternatively [F4] can be set to 'Yes'. In this case the NCB sorts only notes pre-processed by cash-handlers. I.e. the NCB sorting volume consists of all Third Party Unfit notes plus the remainder to [F3] being filled with Third Party Fit notes ([R2] is greyed out and not used in this case).

4.5.3   R3 – Third Party Fitness Threshold

Options: 0 - 100 (decimal)

Function: The threshold at which third parties do fitness sorting. In most cases this value is larger than the Guideline Fit/Unfit Threshold [N5], as cash-handlers typically sort less severely than central banks.

Use: To model the difference between NCB and third party sorting standards.

4.5.4   R4 – Third Party Standard Deviation of Sorting Sensor

Options: Larger than 0, decimal

Function: Standard deviations (inaccuracy) of the fitness sensor in fitness levels, assuming a normal distribution. This parameter is similarly applied to the third party sorting, as [F5] is to NCB sorting.

Use: A figure for the inaccuracy of the recirculation fitness sensor. Similar values as for NCB fitness sensors can be used.

4.6   NCB fitness sorting parameters of the model

4.6.1   F1 – NCB Sorting costs (including re-issuance)

Options: 0 – 500 (EUR/1,000 notes)

Function: Average costs for processing (incl. receipt, sorting, packing, re-issuance or destruction)
              1,000 notes at an NCB.

Use: The total cost for NCB processing.

4.6.2   F2 – Residual note value at threshold:

Options: 0 – 100%

Function: The percentage of its original value [P1] a note has at the Fit/Unfit threshold [N5], i.e. the
              residual value of a note that can be issued just once more. For all other fitness levels
              between 1 (Superfit) and the fit/unfit threshold [N5], values are linearly interpolated (see
              figure below). Notes with a fitness value higher than [N5] (i.e. are more unfit) are
              considered to have no more value.

Use: For calculating the actual value of the false unfit notes destroyed by an NCB taking into
              account its residual value (has no impact on any other model results).
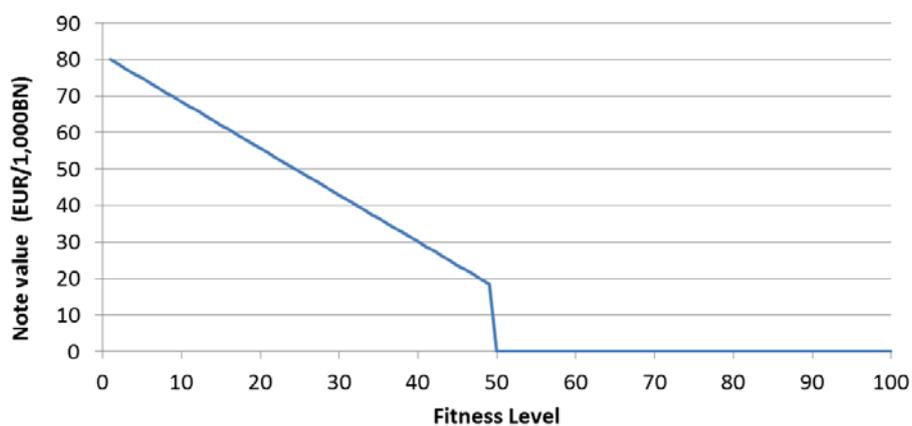

Figure 6: Value of notes at each fitness level (New note issuance costs [P1]:
80 EUR/1,000notes, Fit/Unfit Threshold [N5]: 50 Fitness Levels, Residual note value [F2]: 20%)

4.6.3   F3 - Annual NCB sorting volume

Options: Larger than 0, integer

Function: Annual number of notes sorted at an NCB, consisting of [1] notes coming from circulation (if [F4] is set to 'No'), [2] unfit recirculated notes and [3] – if specified - a share of fit recirculated notes. <u>In case the annual change in circulation volume [C3] is non-zero, the sorting volume will scale with the notes in circulation</u>[13], as otherwise no steady state can be reached (e.g. if the note volume continuously increases while the sorting volume would remain constant, the quality in circulation would continuously decrease).

Use: For modelling the NCB sorting volume in the specific cash-cycle.

4.6.4   F4 - NCB sorts 3P pre-processed notes only

Options: Yes, No

Function: If Yes is selected then the NCB sorting volume [F3] is composed of 3P Unfit and 3P Fit only; [R2] is ignored in this case and $V_{NCB,Sort} = 0$.
If No is selected the NCB sorting volume [F3] consists of 3P Unfit, the amount of 3P Fit specified in [R2] and the remainder being notes from circulation ($V_{NCB,Sort}$)

Use: For easier simulation of cash cycles where the NCB does not sort notes from circulation.

4.6.5   F5 – NCB Standard Deviation of sorting sensor

Options: Larger than 0, decimal

Function: Standard deviations (inaccuracy) of the fitness sensor in fitness levels assuming a normal distribution. The figure can be established by sorting a test-deck composed of notes with varying, known fitness levels.

Use: A value for the inaccuracy of the sorting sensor, typical values are in the range of 5 (for a good fitness sensor) to 20 (for a below average) fitness units.

4.7   NCB Sorting policies of the model

The model supports four different NCB sorting policies. Each policy applies a different logic to setting the operational fit/unfit threshold in each model cycle. These four policies are:

---

[13] If the note volume increases by e.g. 5% p.a., so will the NCB sorting volume as well as the recirculation sorting volume.

4.7.1   S1 - Constant fitness sensor sorting threshold

Options: 1 – 100 fitness levels

Function: If selected, the sorting is carried out with this fixed setting of the fitness sensor.

Use: Implements a sorting policy that runs at a fixed sensor setting irrespective of the incoming note quality (like done at most NCBs).

4.7.2   S2 - Constant False Fit Rate (FFR)

Options: 0.1% - 100%

Function: The percentage of unfit notes (i.e. those with a threshold larger than [N5]) in the notes re-issued by an NCB.

Use: Implements a sorting policy that keeps the quality of the notes issued by an NCB at a constant level by adjusting the sensor threshold to compensate for variations in the input note quality.
N.B.: The algorithm tries to find the closest match but in some cases may not be able to do so. E.g. if the input quality to the NCB contains only 5% unfit, it is impossible to have a sorting threshold issuing 8% false fit (i.e. unfit).

4.7.3   S3 - Constant Shred Rate

Options: 0.1% - 100%

Function: The percentage of the notes destroyed in each cycle which are sorted at an NCB is kept constant.

Use: Implements a sorting policy that adjusts the fitness sensor in each cycle so that the defined percentage of the most unfit notes is shredded.

4.7.4   S4 – Constant Unfit in Circulation

Options: 0.1% - 100%

Function: Specifies a quality target of the quality in circulation for the NCB sorting. If specified, the model adjusts the sorting threshold so as to keep the % of unfit notes in circulation at this level.

Use: The impact of an NCBs sorting on the quality is long-term. The algorithm is doing a forward looking 'guessing' on the quality in circulation and is UNSTABLE if the starting profile is far from the target profile. If this should happen, first try adjusting the NCB

sensor threshold [S1] to find the desired % Unfit in circulation, save the resulting circulation profile and re-run the model with this profile as starting profile, now applying the quality algorithm for final results. See also the section on Tips and Tricks.

A word on the different sorting policies

In steady state conditions any of the four parameters which can be fixed for the NCB sorting policy (Sensor Threshold, False Unfit Rate, Shred Rate, Quality in Circulation) will result in identical steady stated results. I.e. if a certain cash cycle has in its steady state a FU rate of 5%, a Shred Rate of 10%, a Sensor Threshold of 45 and a quality of 15% unfit in circulation, setting either of the four sorting policies to the respective value will deliver identical results.

The main reason for the different policies is to study the impact of them on the quality in circulation over time (i.e. how quickly from a starting condition the new steady state is reached depending on the sorting policy). See also the detailed description of a sorting step in chapter 6 for further explanations.

---

5.   Model Results

The following chapter provides an overview of the results of the model as shown on the 'Main' sheet with the additional results shown on the 'StepResults' and 'DetailedResults' sheet.

## 5.1   Results on 'Main' sheet

5.1.1   Model evolution
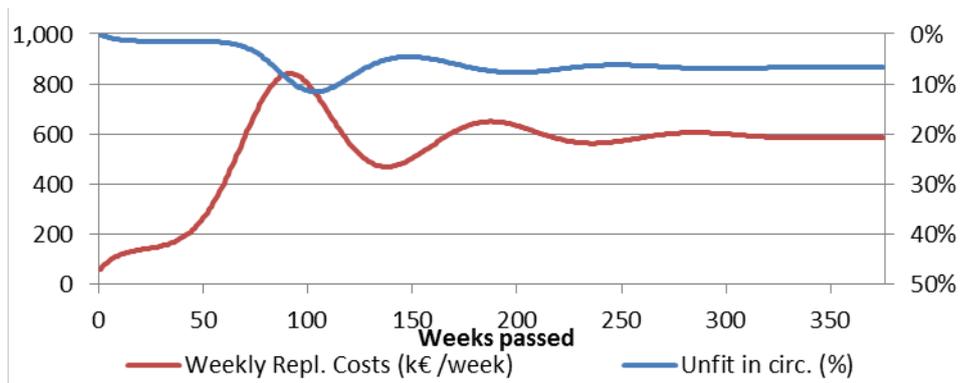
Figure 7: Model evolution chart showing weekly note replacement costs (left y-axis) and % unfit in circulation (right y-axis)

The top shows how the two of the key model parameters evolve over time, and how note replacement and quality stabilise over time.

5.1.2   Volumes and financials per year

			Case 1	
			Notes (M)	Costs/Value
M1	NCB sorting:		*5,000*	€50,000,000
M2	*Unfit notes:*		*562.6*	
M3	*of which False Unfit:*		*279.2*	*€4,681,969*
M4	*Notes put on stock:*		*-*	*€0*
M5	*Note change due to circulation increase / decrease:*		*48.8*	
M6	*Net inflows:*		*-*	*€0*
M7	New notes added:		611.4	€30,572,263
M8	**Total per year (M1+M7):**			**€80,572,263**

Figure 8: Result table showing key <u>annual</u> note volumes and – where applicable – costs.

The table shown above provides the following, final (steady-state) results per year (results are also provided if the model does not arrive at steady-state, but a warning is also shown):

M1   NCB sorting volume ([F3]) and costs ([F1] * [F3])

M2   Notes sorted by NCB to unfit

M3   Volume of False Unfit notes and their residual value (taking into account [F2])

M4   Notes put on stock. Normally 0, only in case of an NCB having inflows so large that not all notes sorted can be reissued this figure is positive.

M5   Note change due to circulation increase / decrease specified in [C3]

M6   Net inflows as specified in [C4] with their value taking into account the fitness distribution of the inflowing notes.

M7   New notes (as specified in P2-P4) added for replacement / growth / flow compensation.

M8   Total NCB cash cycle costs per year consisting of processing costs and costs for new notes (M1+M7).

5.1.3   Cash cycle results

					Case 1	
M9		Lifetime (LT) of notes (simple, months):			16.7	
M10		LT of notes (CIS method, months):			16.2	
M11		Unfit notes in circulation (% of total circulation):			9.1%	
M12		of which defects (% of total circulation):			1.7%	
M13		NCB sorting threshold:			50.0	
M14		NCB shred rate (% of total sorted):			14.3%	
M15		NCB FF_rel (% unfit in NCB fit):			2.5%	
M16		NCB FU_abs (% of total sorted):			6.7%	
M17		Third Party Unfit:			2.7%	
M18		Number of model cycles run:			353	

Figure 9: Result table showing key cash cycle results.

M9   Actual lifetime of notes given in months (simple method, not taking into account any annual growth):

{Notes in Circulation [C2] / Annual replacement volume [M7]} * 12

M10   Actual lifetime of notes given in months (ECB method, taking into account volume changes):

$$LT(months) = \frac{NotesInCirculation}{\frac{AnnualReplacementVolume + NotesIssuedPerYearInclIncrease}{2}} * 12$$

or in circulation model variables:

$$LT(months) = \frac{[C2]}{\frac{2*[M7]+[C2]*[C3]}{2}} * 12$$

M11   Unfit notes in circulation Total: % of notes with a fitness value higher than [N5]

(N.B.: For modelling, this value should match the figures of the respective circulation).

M12   Defects Holes/Tears/Dog-ears: % of total notes with defects in circulation (i.e. % of notes with a fitness level of 100)

M13   NCB fitness sorting threshold in the last cycle

M14   Total % of notes sorted by the NCB which are destroyed (sum of False Unfit + Correct Unfit)

M15   False Fit relative: % False Fit as a percentage <u>of the total notes reissued</u> by the NCB (consisting of correct fit and false fit)

M16   False Unfit absolute: % False Unfit notes as outcome of the NCB sorting process <u>as percentage of the notes sorted</u>.

M17   Total % of notes sorted by the third parties which are returned to the NCB as unfit.

M18   Number of cycles (weeks) until model reached steady state

5.1.4   Additional results
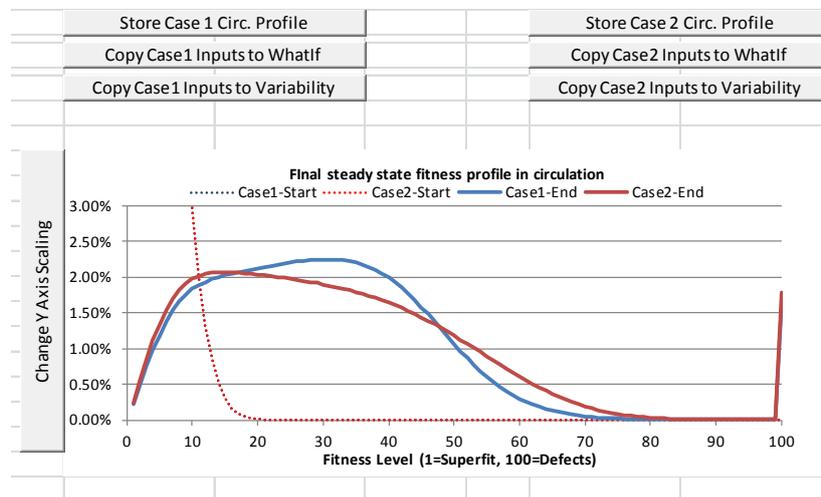
Figure 10: Result table showing additional model results

At the bottom of the 'Main' sheet three buttons exist for each case to either store the resulting note circulation profile (shown in the chart above) for later use, or copy the input parameters for each cash cycle to the 'WhatIf' or 'Variability' sheet for additional analysis (see Section 2.4).

The below chart then shows the quality profile of notes in circulation with the peak at 100 denoting the defective and heavily soiled notes.

5.2   Results on 'StepResults' sheet

Days passed	Total Unfit in Circulation	Defects in Circulation	NCB False Fit relative (%)	NCB False Unfit absolute (%)	NCB Sensor Threshold	NCB Unfit (Shred) Rate (%)	New notes added to (+) / fit notes removed from circulation (-) per cycle	Replacement Costs per cycle incl. growth and flows	Note profile sorted by NCB (Input)	Note profile in Circulation (Out) OR Correlation Coefficient
0	0.00%	0.00%	0.0%	0.0%	50.00	0.0%	-4,816,252	€0		0.981319
7	0.28%	0.21%	0.0%	0.0%	50.00	0.4%	-4,467,694	€0		0.987811
14	0.53%	0.39%	0.0%	0.1%	50.00	0.7%	-4,155,575	€0		0.990367
21	0.75%	0.54%	0.1%	0.1%	50.00	1.0%	-3,875,122	€0		0.991684

Figure 11: Table showing some key results per model cycle

The two tables on the 'StepResults' sheet show the model evolution in each cycle. If [T4] is set to 'No' or 'Yes', then the last row shows the correlation coefficient between this and the previous run. If [T4] is set to 'Yes, incl. Charts' then the last two rows show sparkline charts of the last note profile sorted by the NCB and the note profile in circulation.

N.B.: The costs provided for the Replacement and False Unfit given in the table are per cycle (i.e. 7 days). The values given in the main results ([M3]-[M7]) are annual figures.

No.	Item	Value (all for the respective cycle)
1	Days passed[14]	Cycle Duration [T2] * Cycle Number
2	Total Unfit in Circulation	[M11]
3	Defects in Circulation	[M12]
4	NCB False Fit relative (%)	[M15]
5	NCB False Unfit absolute (%)	[M16]
6	NCB Sensor Threshold	[L1]
7	Shred Rate (%)	[L11]
8	New notes added to (+) / fit notes removed from circulation (-) per cycle	The residual value of false unfit notes destroyed per week ~[M7, number]/52
9	Replacement Costs per week	~[M7, value]/52 , can be negative if inflows are so large that NCB puts notes on stock.
10	Notes sorted by NCB (input)*	The fitness profile of the notes sorted at the NCB. I.e. the summary profile of circulation notes, third party unfit and a share of third party fit.
11	Notes in circulation (output)*	Same profile as shown in the lower centre result chart for the respective cycle (Notes$_{End}$ in figure 3)

*) Only if [T4] is set to 'Yes, incl. Charts', otherwise the correlation coefficient between this cycle and the last is shown in column 11.

5.3   Results on 'DetailedResults' sheet

The first table shows the detailed results for the last sorting cycle:

---

[14] The last line of the table always shows the maximum number of days of Case 1 and Case 2 for which model ran. Necessary for charting.

L1   The NCB sensor threshold according to the NCB sorting policy specified in [S1]-[S4] that was used in the last sorting cycle.

L2   The average soil level of the notes sent to the NCB sorting machine.

L3/L4   The percentage of Fit/Unfit notes in the notes sent to the NCB sorting machine.

L5 – L8 Correct/False Fit and Unfit notes as outcome of the NCB sorting process at the sensor threshold [L1] as percentage of the total sorting volume.

L9   % False Fit as a share of the total notes reissued by the NCB (consisting of correct fit and false fit)

L10   % False Unfit as a share of the total notes destroyed by the NCB (consisting of correct unfit and false unfit)

L11   Total % of notes sorted by the NCB BSM which are destroyed (Sum of False Unfit + Correct Unfit)

L12   The fitness decrease (note-ageing) per cycle according to the note life and the ageing method chosen (see section 4.1.3 for an example).

5.4   Note balance in the last model cycle

For more information how these figures are calculated, please refer to the example given in chapter 6.

B1   Total number of notes in circulation (=[C2])

B2   The number of notes not processed (i.e. just circulating without being seen by an NCB or third party sorting machine) within 7 days.

N.B.: Values for [B2] – [B14] apply to the number of notes

in each sorting cycle (i.e. 7 days).

B3   Number of notes sorted by third parties

B4/B5   Notes classified as Fit/Unfit by a third parties ([B3] = [B4] + [B5])

B6   Number of notes sorted by the NCB

B7/B8/B9 Composition of notes sorted at NCB of circulation, third party unfit and recirculation fit notes ([B6] = [B7] + [B8] + [B9])

B10/B11 Notes classified as fit/unfit by an NCB sorting machine ([B6] = [B10] + [B11])

B12   The number of notes which are coming into the cash cycle in each cycle of 7 days (as specified per year in [C4])

B13   The number of notes which are leaving the cash cycle in each cycle of 7 days (as
             specified per year in [C6])

B14   Number of fit notes put on stock at the NCB which could not be re-issued due to
             substantial inflows

B15   New notes added/removed due to change in circulation volume [C3]

B16   Total new notes added/removed ([B11] - [B12] + [B13] - [B14] + [B15])

5.5   Average note return times

RT1 – RT3 The average return time of a note to come back to the NCB, a third party or either of
           them.

5.6   Additional graphical results

Further below various charts show the model evolution and results of the last sorting step.

WARNING: In case of different annual note growth rates [C3], the comparative case1/2 results are influenced by the different amount of new notes flowing in, with new notes having a 'cleaning function' on the overall circulation. The costs outputted DO NOT include the costs for the increased issuance, but are based on the starting volume for better comparison. See the detailed calculation steps in chapter 6 for clarification.

6. Example model cycle

6.1 The case

This section provides a step-by-step explanation of one calculation step of the model. For ease of understanding a fictitious case with round figures is taken, assuming that the starting note quality in circulation is homogenously distributed (all fitness levels from 1-100 have a share of 1% each in circulation, i.e. the circulation contains 50% unfit notes ([C1]=Homogeneous) and one single sorting cycle is applied with the NCB policy to achieve a false unfit rate of 8% in the notes reissued. All starting parameters are shown in the figure below.

	Technical Parameters of the model		
			Calculate C
T1	No. of model cycles:	1	
T2	Cycle Duration (days):	7	
T3	Note Life Calculation:	Real	Update Ca
T4	Show Calculation Steps:	Yes incl. Charts	Update Ca
T5	Steady State Correlation Factor:	0.999999995	
T6	Quick Finish if Steady State:	Yes	

	Input		
			Case 1
	Banknote production		
P1	Note production and issuance costs (I/1,000 notes):	50	
P2	New note fitness level:		
P3	St.Dev of new notes (in fitness levels):		
P4	OR: New note distribution profile:	SomeVariation	
	Circulation		
C1	Note distribution at start:	Homogenous	
C2	Notes in circulation:	1,000,000,000	
C3	Annual change in note circulation:	+10.00%	
C4	External notes entering per year:	250,000,000	
C5	Fitness distribution of external notes:	20%UnfitInCirc	
C6	Notes leaving per year:	125,000,000	
	Note Life		
N1	Ageing Model:	Poisson	
N2	Theoretical note life (soil only, in months):	25.0	
N3	Defect likelihood (% per year):	10.00%	
N4	Defect profile:	IncreaseFrom1	
N5	Fit/Unfit threshold (in fitness levels):	50	
	Third Party Fitness Sorting (Recirculation)		
R1	Annual Third Party (3P) sorting volume:	1,000,000,000	
R2	% of 3P fit notes returned to NCB:	10%	
R3	3P fitness sorting threshold:	60	
R4	3P St.Dev of sorting sensor (in fitness levels):	15	
	NCB Fitness Sorting		
F1	NCB sorting costs (I/1,000 notes):	10	
F2	Residual note value at threshold:	20%	
F3	Annual NCB sorting volume:	3,000,000,000	
F4	NCB sorts 3P pre-processed notes only:	No	
F5	NCB St.Dev of sorting sensor (in fitness levels):	10	
	NCB Sorting Policy		
S1	Fitness sensor sorting threshold (fitness levels):		
S2	OR: False-fit rate (% of unfit notes in fit notes issued):	8.00%	
S3	OR: Shred rate (% of notes unfit per sorting cycle):		
S4	OR: Unfit in circulation (% of total circulation):		

Figure 12: Starting conditions for the exemplary single step model calculation.

6.2 The calculation

The calculation follows in its basic principle the flowchart in figure 3. In the following each step is described. Items in square brackets refer to model inputs. For the variable names, the same names as used in the VBA code are used for easier traceability. Variables starting with 'A_' indicate one-dimensional arrays, usually from 1 to 100 containing a value per fitness level. Variables ending with 'PC' are values 'Per Cycle' of the model. The model is implemented as a VBA function called *'CalcCirculationModel'* in the *'mCircModel'* module of the Excel sheet with the source code being freely accessible.

1. <u>Calculate price per new note</u>

   BNPriceNewPerNote = BNPriceNew[P1] / 1000 = 50 / 1000 = <u>0.05</u>

2. <u>Calculate notes sorted at NCB per cycle</u>

   TotalNotesSortedAtNCB_PC = NCBProcessingVolume[F3] * ModelCycleDuration[T2] / 365 = 3 bln * 7 / 365 = <u>57.5 mln</u>

3. <u>Calculate notes recirculated per cycle</u>

   NotesRecirculatedPC = RecirculationSortingVolume [R1] * ModelCycleDuration[T2] / 365 = 1 bln * 7 / 365 = <u>19.2 mln</u>

4. <u>Calculate note amount change per cycle (due to growth or reduction)</u>

   NoteChangePC=NotesInCirculation[C2] * ((1+AnnualCirculationChange[C3]) ^ (1/(365/ModelCycleDuration[T3]))-1) =

   $\qquad$ = 1bln * (1+0.1) ^ (1/(365/7))-1) = <u>1.8 mln</u>

5. <u>Calculate incoming notes per cycle</u>

   ExternalNotesIncomingPC = ExternalNotesIn[C4] * ModelCycleDuration[T2]/ 365 = 250 mln * 7 / 365 = 4.<u>8 mln</u>

6. <u>Calculate notes leaving per cycle</u>

   NotesLeavingPC = NotesLeaving[C6] * ModelCycleDuration[T2] / 365 = 125mln * 7 / 365 = <u>2.4 mln</u>

7. <u>Calculate defect probability per cycle</u>

    DefectsPC = CEperYear[N3] / 365 * ModelCycleDuration = 0.1 / 365 * 7 = <u>0.19%</u>

8. <u>Do the recirculation sorting on the incoming note profile and obtain the volume of recirculation fit and unfit per cycle</u>

   The notes in circulation (A_NotesIn, as specified by profile [C1]) are subjected to a recirculation sorting step, with the recirculation sensor having a characteristic of A_RecircSensProfile, defined by the recirculation threshold [R3] of 60 and the recirculation standard deviation [R4] of 15. A_RecircSensProfile is created using the below Excel function / code:

```
For i = 1 to 99
        A_RecircSensProfile(i) = NormDist (i, 60, 15,TRUE)
Next i
A_RecircSensProfile(100) = 1 'N.B.: notes at level 100 (i.e. defective) are always sorted to unfit
```

Now, for each Fitness level, A_RecircSensProfile is applied to A_NotesIn resulting in A_RecircUnfit; the same is done for (1 - A_RecircSensProfile), obtaining A_RecircFit; The below code is used for this:

```
For i = 1 To 100
        A_RecircUnfit(i)      = A_NotesIn(i) *  A_ RecircSensProfile(i)
        A_RecircFit(i)        = A_NotesIn(i) *  (1-A_ RecircSensProfile(i))
Next i
```

The complete input and result profiles are shown in the charts below:

Input
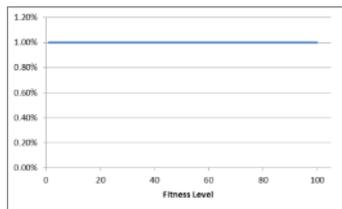
A_NotesIn (Sum = 100%), homogenous profile
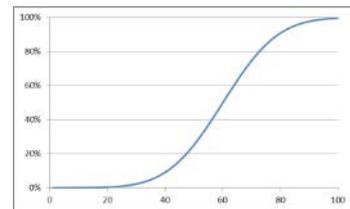
A_RecircSensProfile(Mean=60 StDev=15)

Resulting profiles (normalised)
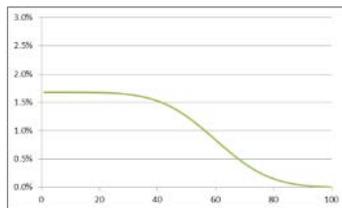
A_RecircFit
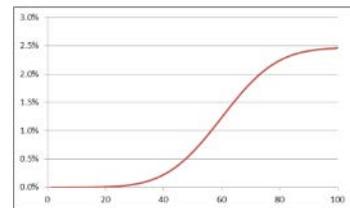
A_RecircUnfit

The overall percentage of recirculation Fit and Unfit notes is calculated by summing up A_RecircUnfit and A_RecircFit respectively, resulting in:

RecircUnfit = 40.5%

RecircFit   = 59.5%

The amount of Recirc-Fit and Recirc-Unfit per cycle is calculated as follows:

RecircUnfitPC = RecircUnfit * NotesRecirculatedPC = 40.5% * 19.2 mln = 7.8 mln

RecircFitPC = RecircFit * NotesRecirculatedPC = 59.5% * 19.2 mln = 11.4 mln

9. Calculate RecircFit sorted and not sorted by NCB per cycle

RecircFitSortedByNCB_PC = RecircFitPercBackToNCB[R2] * RecircFitPC = 10% * 11.4 mln = 1.1 mln

RecircFitNotSortedByNCB_PC = (1-RecircFitPercBackToNCB[R2]) * RecircFitPC = (100%-10%)*11.4 mln = 10.3 mln

10. Calculate circulation notes sorted at NCB per cycle

This is simply the amount of the total NCB sorting volume per cycle minus the unfit from recirculation and the fit from recirculation returned to the NCB.

$$CircNotesSortedAtNCB\_PC = TotalNotesSortedAtNCB\_PC - RecircUnfitPC - RecircFitSortedByNCB\_PC$$

$$= 57.5 \text{ mln} - 7.8 \text{ mln} - 1.1 \text{ mln} = \underline{48.6 \text{ mln}}$$

11. <u>Calculate notes not processed per cycle</u>

$$NotesNotProcessedPC = NotesInCirculation[C2] - CircNotesSortedAtNCB\_PC - NotesRecirculatedPC$$

$$= 1 \text{ bln} - 48.6 \text{ mln} - 19.2 \text{ mln} = \underline{932.2 \text{ mln}}$$

12. <u>Create the profile of notes sorted at the NCB (mix of circulation notes, recirculation unfit and recirculation fit)</u>

The weighted average of the arrays A_NotesIn, A_RecircUnfit and A_RecircFit is created as A_NotesToSortByNCB using the below code[15] resulting in the array shown below:

```
For i = 1 To 100
    A_NotesToSortByNCB(i) = (RecircUnfitPC * A_RecircUnfit(i) + RecircFitSortedByNCB_PC * A_RecircFit(i) +
                            CircNotesSortedAtNCB_PC * A_NotesIn(i)) / TotalNotesSortedAtNCB_PC
Next i
```


| A_NotesIn (48.6 mln notes) | A_RecircUnfit (7.8 mln notes) | A_RecircFit (1.1 mln notes) |

A_NotesToSortByNCB (57.5 mln notes)
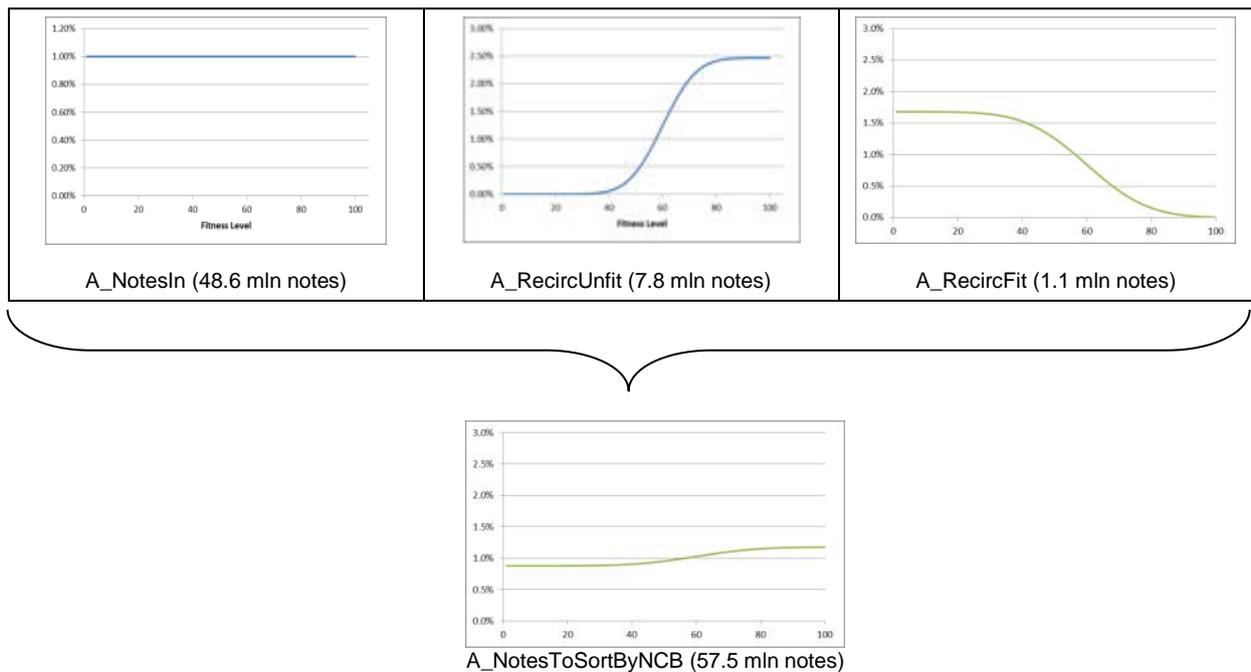
13. <u>Determine the NCB sorting threshold to sort at the indicated sorting policy</u>

The question to answer now is at which NCB sensor threshold the defined policy of 8% False Fit in the note re-issued by the NCB [S2] can be achieved. For this, the next step (see 14.) is carried

---

[15] All profiles are normalised before this step.

out 100 times with the NCB sensor threshold varying from 1 to 100. The False Unfit rate is determined for each 'virtual' sorting step and the final NCB sensor threshold is determined via a linear interpolation algorithm. The figure below shows the False Fit rate when sorting A_NotesToSortByNCB at an NCB sorting threshold of 1 to 100 and highlights the NCB sorting threshold (49.50) where the relative False Fit rate of 8% is achieved.
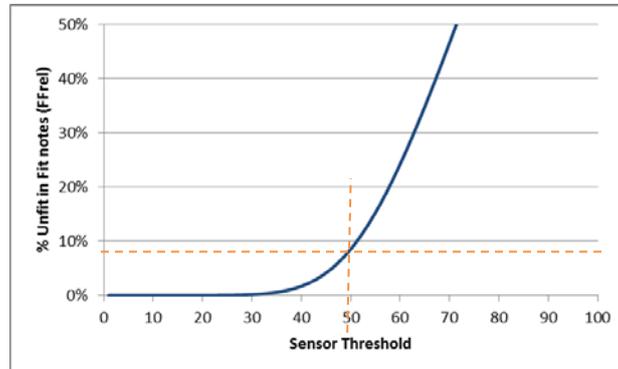
NCBSortThreshold = 49.50

14. Carry out the sorting at the NCB

Similar to step 8, A_NotesToSortByNCB is subjected to a sorting at the NCB using a profile based on an NCBSortThreshold of 49.50 and an NCBStDev[F5] of 10):

```
For I = 1 to 99
        A_ NCBSensProfile (i) = NormDist (i, 49.50, 10,TRUE)
Next i
A_ NCBSensProfile (100) = 1 'N.B.: notes at level 100 (i.e. defective) are always sorted to unfit
```

Input

A_NotesToSortByNCB

A_NCBSensProfile (Mean=49.50 StDev=10)
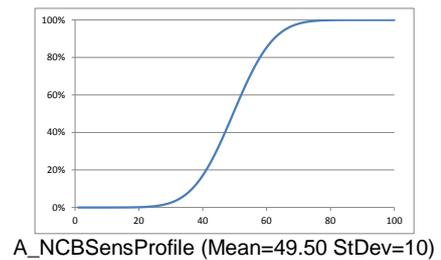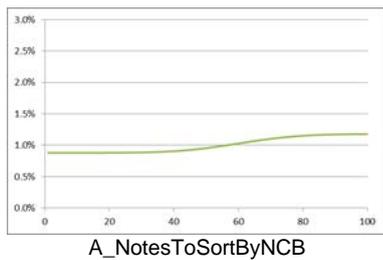
Result
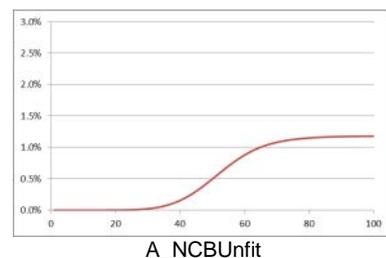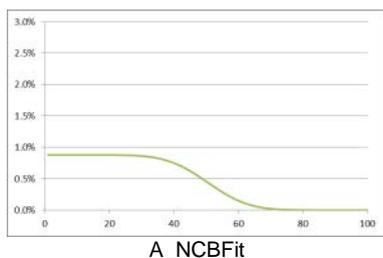
A_NCBFit

A_NCBUnfit

The overall percentage of NCB-Fit and -Unfit notes is again calculated by summing up (integrating) A_NCBUnfit and A_NCBFit respectively, resulting in:

NCBFit   = 44.0%

NCBUnfit = 56.0%

The amount of False Fit and Unfit is calculated using the two resulting profiles and checking which of the notes are actually fit/unfit against FitUnfitThreshold[F5] using the following code:

```
For i = 1 to FitUnfitThreshold
        A_NCBCorrectFit(i) = A_NCBFit(i)
        A_NCBFalseUnfit(i) = A_NCBUnfit(i)
Next i

For i = (FitUnfitThreshold+1) to 100
        A_NCBFalseFit(i) = A_NCBFit(i)
        A_NCBCorrectUnfit(i) = A_NCBUnfit(i)
Next i
```

Resulting in:
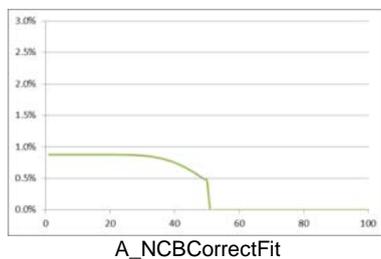
A_NCBCorrectFit
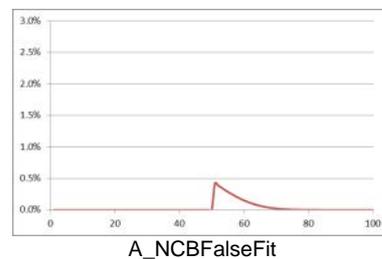
A_NCBFalseFit
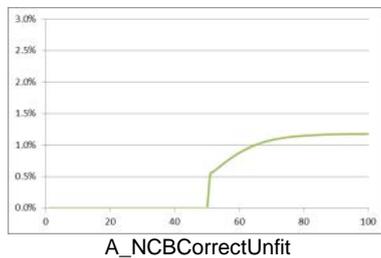
A_NCBCorrectUnfit
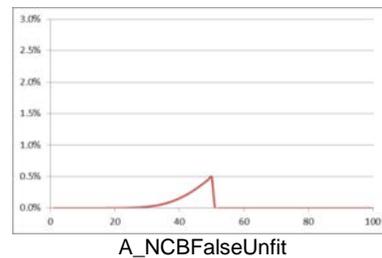
A_NCBFalseUnfit

The overall percentage of the four categories is again calculated by summing up the four arrays, resulting in:

NCBCorrectFit	= 40.5%
NCBCorrectUnfit	= 51.8%
NCBFalseFit	= 3.5%
NCBFalseUnfit	= 4.2%

The above four note categories (totalling 100%) are the percentages in relation to the absolute NCB sorting volume. False Fit (and sometimes False Unfit) rates are within the Eurosystem also used relative to the total of the Fit and Unfit notes only. These relative figures are calculated as follows:

NCBFalseFitRel   = NCBFalseFit / NCBFit = 3.5% / 44.0% = 8.0 % *(N.B.: The figure that was specified in [S2])*

NCBFalseUnfitRel   = NCBFalseUnfit / NCBUnfit = 4.2% / 56.0% = 7.4%

The amount of the six different note categories created by the NCB sorting per cycle is calculated by multiplying the above calculated percentages with the total sorting volume:

NCBFitPC = NCBFit * TotalNotesSortedAtNCBPC = 44.0% * 57.5 mln = <u>25.3 mln</u>
NCBUnfitPC = NCBUnfit * TotalNotesSortedAtNCBPC = 56.0% * 57.5 mln = <u>32.2 mln</u>

NCBCorrectFitPC = NCBCorrectFit * TotalNotesSortedAtNCBPC = 40.5% * 57.5 mln = <u>23.3 mln</u>
NCBCorrectUnfitPC = NCBCorrectUnfit * TotalNotesSortedAtNCBPC = 51.8% * 57.5 mln = <u>29.8 mln</u>
NCBFalseFitPC = NCBFalseFit * TotalNotesSortedAtNCBPC = 3.5% * 57.5 mln = <u>2.0 mln</u>
NCBFalseUnfitPC = NCBFalseUnfit * TotalNotesSortedAtNCBPC = 4.2% * 57.5 mln = <u>2.4 mln</u>

15. <u>Create the profile of notes in circulation after sorting one cycle</u>

The note profile after the recirculation and NCB sorting is now generated by making the weighted average of

1. notes which were not processed in this cycle,

2. recirculated fit notes,

3. notes entering the cash cycle specified by [C4] and [C5],

4. NCB fit notes (which are returned to circulation) and

5. an amount of new notes corresponding to

    o NCB unfit notes, plus

    o any changes due to a note increase/decrease as specified in [C3], plus

    o any compensations for notes leaving the cash cycle specified in [C6]

using the below formulas:

NotesUnchangedPC = NotesNotProcessedPC – NotesLeavingPC = 932.2 mln – 2.4 mln = <u>929.8 mln</u>

NewNotesAddedPC = NCBUnfitPC + NoteChangePC + NotesLeavingPC – ExternalNotesIncomingPC =

    = 32.2 mln + 1.8 mln + 2.4 mln – 4.8 mln = <u>31.7 mln</u>

```
For i = 1 To 100
        A_NoteProfileAfterSorting(i) = (NotesUnchangedPC * A_NotesIn(i) +
                ExternalNotesIncomingPC * A_ExternalNotesIn(i) +
                RecircFitNotSortedByNCB_PC * A_RecircFit(i) + _
                FitNotesAddedPC * A_NCBFit(i) + _
                NewNotesAddedPC * A_NewNotes(i)) /
                (NotesUnchangedPC + ExternalNotesIncomingPC + RecircFitNotSortedByNCB_PC +
                FitNotesAddedPC + NewNotesAddedPC)
Next i
```

Resulting in:

Input

	A_NotesIn	A_RecircFit	A_NCBFit	A_NewNotes	A_ExtNotesIncoming
Note profile					
Weight	NotesUnchangedPC (929.8mln)	RecircFitNotSortedByNCB_PC (10.3mln)	NCBFitPC (25.3mln)	NCBUnfitPC + NoteChangePC + NotesLeavingPC - ExtNotesIncomingPC (31.7 mln)	ExtNotesIncomingPC (4.8 mln) [C5[20%UnfitInCirc]]
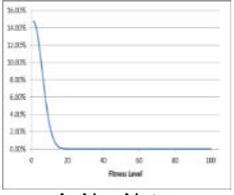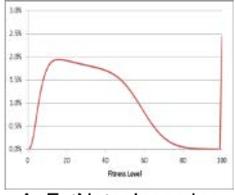
Result
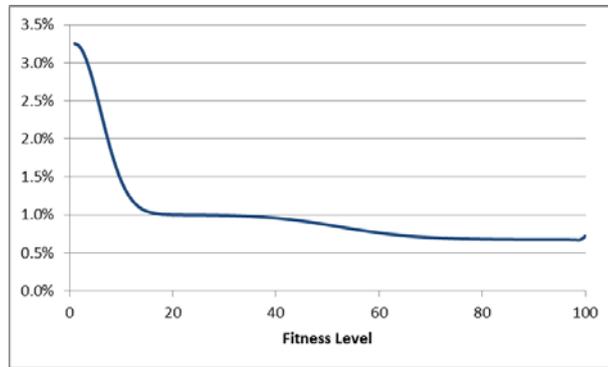
A_NotesAfterSorting

(1001.9 mln)

16. <u>Calculate the ageing per cycle</u>

The average amount of steps a note is ageing due to soiling is calculated, taking into account the ageing method selected in NoteLifeCalculation[T3] (see also the example on page 16). In this example 'Real' is selected, therefore the average fitness level of new notes (AvgFitLevelNewNotes) is determined first. Subsequently the amount of ageing steps is calculated based on ModelCycleDuration[T2], NoteLifeTime[N2] and the NCBFitUnfitThreshold [N5] using the below code:

If NoteLifeMethod = "Ideal" Then AgePerCycle = ModelCycleDuration / 30 / NoteLifeTime * (NCBFitUnfitThreshold - 1)[16]

```
If NoteLifeMethod = "Real" Then
      For i = 1 To 100
                AvgFitLevelNewNotes =AvgFitLevelNewNotes + A_NewNotes(i) * i
      Next i
      AgePerCycle = ModelCycleDuration / 30 / NoteLifeTime * (NCBFitUnfitThreshold - AvgFitLevelNewNotes)
End If
```

The calculation results in:

$$AgePerCycle = ModelCycleDuration / 30 / NoteLifeTime * (NCBFitUnfitThreshold - AvgFitLevelNewNotes)$$
$$= 7 / 30 / 25 * (50\text{-}4.68^{[17]}) = \underline{0.42}$$

17. <u>Create the ageing distribution</u>

    Based on AgePerCycle the ageing distribution selected in [N1] is calculated. As 'Poisson' is selected, calculation is done using the built-in Excel formula:

```
For I = 1 to 100
          A_AgeingProfile(i)= POISSON.DIST(i, 1.81, FALSE)
      Next i
```

    Resulting in
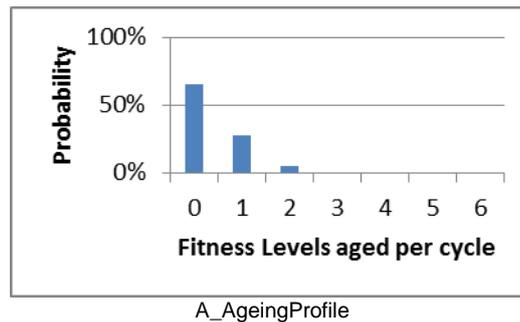
A_AgeingProfile

18. <u>Age the resulting note profile</u>

    A_AgeingProfile is now applied to every single value in A_NotesAfterSorting, apart from those that already are at level 100 (cannot get any more soiled), using the below code:

```
For i = 1 To 99
      For j = 0 To 99
      If i + j <= 99 Then
                A_NotesAfterAgeing(i + j) = A_NotesAfterAgeing (i + j) + A_NotesAfterSorting(i) * A_AgeingProfile(j + 1)
      End If
      Next j
Next i
```

---

[16] Not used in this example.

[17] The average fitness level of A_NewNotes, see earlier for the profile.

In a last step A_NotesAfterAgeing is normalised again with all notes which would have fitness levels >100 being put in the 100 bin via:

$$\text{A\_NotesAfterAgeing (100)} = \text{A\_NotesAfterSorting (100)} + \text{Sum\_Array(A\_NotesAfterSorting, 1, 99)}^{18} - \text{Sum\_Array(A\_NotesAfterAgeing, 1, 99)}$$

19. <u>Apply the defects</u>

The probability of a defect is modelled as a function of the fitness level based on the profile defined by [N4] and is in this example set to 'Increase from 1' as shown below:
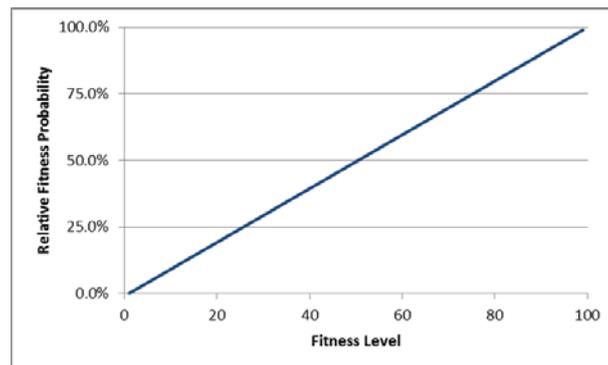
A_DefectProbability

Applying this profile to A_NotesAfterAgeing is done in a 2 step approach. In a first step the profile is applied without any factor and the number of defects added in this cycle is compared to DefectsPC (i.e. the number of defects which are specified, see Step 7) in order to obtain the factor at which A_DefectProbability needs to be applied. In a second step A_DefectProbability is applied to A_NotesAfterAgeing using this factor.

```
A_ResultBackup() = A_NotesAfterAgeing()
     For i = 1 To 99
        old_Result = A_ResultBackup(i)
        A_ResultBackup(i) = A_ResultBackup(i) * (1 - A_DefectProbability(i))
        NewDefectsTotal = NewDefects + old_Result - A_ResultBackup(i)
     Next i

     'Now we can determine the factor to apply to reach the desired amount of defects per cycle
        DefectFactor = DefectsPC / NewDefectsTotal     (0.82%/0.79%=1.034)

        NewCDefects = 0
     'Now we apply the changes to the actual profile
        For i = 1 To 99
           old_Result = A_NotesAfterAgeing(i)
           A_NotesAfterAgeing (i) = A_NotesAfterAgeing(i) * (1 - A_DefectProbability(i) * DefectFactor)
           NewCDefects = NewCDefects + old_Result – A_ NotesAfterAgeing (i)
        Next i
```

---

[18] A function calculating the sum of a one dimensional array from entry 1 to entry 99.

'Add the catastrophic notes to the 100 bin
    A_ NotesAfterAgeing (100) = A_ NotesAfterAgeing (100) + NewCDefects

Applying the ageing and defects results in:

A_NotesAfterAgeing (red curve)
(A_NotesAfterSorting shown in blue as reference)
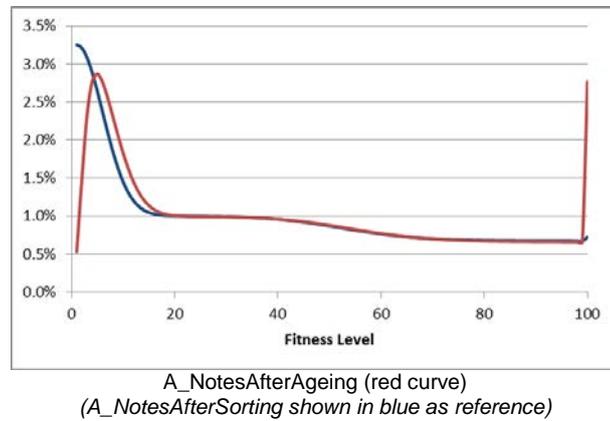
20. Check steady-state and either

    a. start next cycle

       The correlation coefficient between A_NotesAfterAgeing and A_NotesIn is calculated
       using the CORREL Excel function and if no steady state is reached, the model is run
       again from step 1 with A_ NotesAfterAgeing being used as new starting value for
       A_NotesIn. (In this specific example steady state is reached after 373 cycles).

    b. or, output final results

       In case steady state has been reached, the calculation is stopped and the result charts
       are created and figures outputted. Whilst some of the outputs should be self-explanatory
       based on the above explanations, the calculations for some others are shown below:

       AnnualReplacementVolume   = NCBUnfit / CycleDuration * 365
       AnnualFalseUnfitVolume   = NCBFalseUnfit / CycleDuration * 365
       AnnualReplacementCosts   = NCBUnfit * BNPriceNewPerNote / CycleDuration * 365

       ReturnTimeNCB   = NotesInActiveCirc / (TotalNotesSortedAtNCB_PC * CycleDuration)
       ReturnTimeRecirc   = NotesInActiveCirc / (NotesRecircPC * CycleDuration)
       ReturnTimeNCBorRecirc = NotesInActiveCirc / (NotesRecircPC + TotalNotesSortedAtNCBPC) * CycleDuration)

7. Batch functions to run multiple model cases

When modelling a cash cycle it can be useful to see how the model would react if multiple of the input parameters change. While the sheet 'Main' allows the comparison of two cases with one or multiple different parameters, extended functionality to carry out a sensitivity analysis of a case is possible using the sheets 'WhatIf' and 'Variability'. These sheets do not provide the detailed graphical analysis of the 'Main' sheet (including the two results sheets), but instead provide the key-results of multiple cases in direct comparison.

You can copy any cash cycle cases which have been set-up in the 'Main' sheet to the 'WhatIf' and 'Variability' sheet using the buttons provided there (see also Section 2.4).

N.B.: The algorithm is identical when run from any sheets of the circulation model. The only difference is in the different input masks and the display of the results.

7.1 Batch functions on 'WhatIf' sheet

The cash cycle is defined on the left of the sheet using the same input mask as on the 'Main' sheet. In column K you can then specify the change to each single input parameter. In case any cells are left empty, the respective case will be skipped in the calculation.

In row 15 the tolerances for the colouring of the changes (conditional-formattings) can be adjusted.

After pressing 'Update Calculations' (in row 12), the base case as well as all variations are calculated and the key results shown. The results of the different cases are coloured in relation to their deviation from the base case.

Due to the time needed and also the complexity you can only calculate the different cases at a fixed sorting threshold (sorting policy S1, see Section 4.7.1)


Figure 13: Result section on the 'WhatIf' sheet'

Also, changes to the new note profile can only be specified via changes to the numeric values [P2] and [P3]; the specification via any arbitrary profile [P4] is not supported on this sheet.

N.B.: The three standard new note profiles provided on the 'Main' sheet can be set via the following values:

Profile Name	Mean [P2]	StDev [P3]
Perfect	1	0.01
Some Variation	1	5
Large Variation	5	10

7.2 Batch functions on 'Variability' sheet

Also here the model case is defined on the left. In the range J6:K7 relative changes to the specified NCB and CH sorting volume can be specified. In cell J9 the number of iterations/ steps can then be specified (3, 5, 7, 9 or 11).

After pressing the below 'Update Calculations' button all variations in the NCB and CH sorting volume are calculated and the key figures output in the tables to the right.

The variability analysis is useful once a cash cycle is known to the changes or shifts in NCB and CH sorting involvement.


Figure 14: Input (left) and Result section (right) on the 'Variability' sheet'

7.3 Modelling via the User Defined Function (UDF)

The sheet 'UDF' demonstrates how you can set up any number of cases with varying inputs using the circulation model as a standard function in Excel (The 'UDF' worksheet is locked, but can be simply unlocked by the button in the ribbon, or by the standard Excel function).

The function must be entered as an Array function (see https://support.office.com/en-us/article/Create-an-array-formula-e43e12e0-afc6-4a12-bc7f-48361075954d). When changing any part of the Array Function DO NOT use the 'Insert Function Option' in Excel, but change the formula directly in the entry field as otherwise Excel might crash. Also take note that using too many UDFs will take time for Excel

to recalculate the formulas on startup. Due to speed constraints, the UDF only supports modelling at a constant NCB sensor threshold.

A general recommendation when using UDFs to simulate cases is to always create a new file which you only use for this specific UDF modelling. Having many UDFs in a file will make it slower as Excel needs from time to time recalculate all these functions which can take a substantial amount of time and also reduce the performance of the circulation modelling using the other (batch) functions.

8.   Known limitations of the model

Sorting for constant unfit in circulation (circulation quality), sorting policy [S4]

When the starting note profile (specified in [C1]) is very different from the steady state of the model, the model might oscillate and never reach a steady state. To avoid this, apply this sorting policy only on a starting note profile close to the desired one. Having a higher return frequency to the NCB (by e.g. increasing the NCB sorting volume) also helps for this sorting policy to become more stable.

9.   Tips & Tricks

General

- You can abort the calculation any time by pressing 'ESC'.

- Steady State (SS) is reached quicker if your starting note profile is close to the final results. Selecting a profile for [C1] which is close to your final result will reduce the number of iterations required to reach SS.

- The status bar displays the current calculation cycle and the correlation factor. Once steady state conditions for a case are reached this is indicated by a beep.

- You can manually unprotect all sheets and have free access to the VBA modules for background checking, studying … .

On the 'Main' sheet

- Setting 'Show Calculation Steps' to 'No' speeds up calculation about 20x.

- You can change the Y-axis scale of the fitness profile (the chart called 'Final steady state fitness profile in circulation' on the 'Main' sheet) manually by clicking the button left of it. This can be useful to show the whole profile in cases where you have a high number of defect notes (the dotted lines in the final profile chart show the starting condition).

END OF DOCUMENT