

Manual JohneSSim Model

*A computer simulation model to
evaluate Johne's disease control program
on the epidemiological and economical consequences*

H. Groenendaal

Wageningen, July 2001

1. Table of Contents

1. Table of Contents.....	3
2. Acknowledgements.....	5
3. Introduction.....	6
3.1. About the model.....	6
3.2. How to use this manual?.....	7
3.3. Overview.....	7
3.3.1. Stochasticity.....	7
3.3.2. Time steps.....	7
3.3.3. General overview.....	7
4. Using the Johne's model.....	8
4.1. Before you begin.....	8
4.2. Result-files.....	8
4.2.1. Generate start herd (.stv-file).....	9
4.2.1.1. Generate a new InitialHerd.....	9
4.2.1.2. Generate new start herd-files (.stv-files).....	10
4.2.2. Simulate strategies (.rst files).....	11
4.3. Simulation of more profiles/statusses.....	12
4.4. Assimilation of results.....	12
4.5. Assimilation of more files at once.....	13
5. Main input menu.....	14
5.1. General.....	14
5.1.1. Time steps.....	14
5.1.2. Seeds.....	14
5.2. Dairy Herd simulation.....	14
5.2.1. General description.....	14
5.2.2. Related input in the model.....	14
5.2.2.1. Herd type and size.....	14
5.2.2.2. Herd size.....	15
5.2.2.3. Culling.....	15
5.2.2.4. Calving pattern and age first calving.....	17
5.3. Disease and management.....	17
5.3.1. Overview.....	17
5.3.2. Fetal infections.....	18
5.3.3. Infections around birth.....	18
5.3.4. Infections due to drinking mixed colostrum.....	18
5.3.5. Infections due to drinking colostrum of the own dam.....	18
5.3.6. Infections due to drinking pooled bulk milk.....	18
5.3.7. Infections due to suckling at more cows.....	19
5.3.8. Infections due to drinking rest milk.....	19
5.3.9. Infections due to environmental contamination.....	19
5.3.10. Infections due to introduction of an infected animal in the herd.....	20
5.3.11. Infection course of infected animals.....	20
5.4. Control of Johne's disease.....	21
5.4.1. Overview.....	21

5.4.2. 'Test and Cull' strategies	21
5.4.2.1. Test programs	21
5.4.2.2. Test attributes	22
5.4.3. Management strategies	24
5.4.3.1. Common management tools	24
5.4.3.2. Grouping animals	24
5.4.4. Vaccination strategies	25
5.5. The economics	26
5.5.1. The losses of Johne's disease	26
5.5.2. The costs of the control of Johne's disease	27
5.6. Assimilation of results	29
5.7. Assimilation of more files at once	29
6. VBA Code	30
6.1. Simulation Code	30
6.2. Assimilation Code	30

2. Acknowledgements

The JohneSSim model was developed within the Farm Management Group of the Wageningen University, under supervision of dr. ir. H.S. Horst, dr. ir. A.W. Jalvingh and dr. M. Nielen. The Animal Health Service in the Netherlands financially supported the development of the JohneSSim model.

Furthermore the members of the 'Johne's disease simulation model discussion group', are acknowledged for their important contribution in the JohneSSim model:

Dr. J. Verhoeff,	(Head section of Cattle Health, Animal Health Service),
H.J. van Weering	(Animal Health Service),
C.H.J. Kalis	(Animal Health Service),
J.A.M Muskens	(Animal Health Service),
ir. P. Wever	(Product-boards for Livestock, Meat and Eggs),
L. Meyer	(National farmers organization),
K. Frankena	(Wageningen Agric. Univ., Animal Health and Reproduction)
E. Piercy	(Ministry of Agriculture, Nature management and Fisheries)
F. van Zijderveld	(Dutch organization for agricultural research)
H. van Roermund	(Dutch organization for agricultural research)
L. Lobsteyn	(DVM, Sneek, The Netherlands),

The author also would like to thank the Center of Animal Health and Productivity, especially D.T. Galligan, VMD, MBA for his contribution in additions, made to the JohneSSim model. Also Dr. R.W. Sweeney and Dr. R.H. Whitlock (University of Pennsylvania) are acknowledged for their inputs in this study.

The JohneSSim model is owned by the Farm Management Group of the Wageningen University and may not be used without permission.

3. Introduction

3.1. About the model

JohneSSim is a computer simulation model, programmed in Excel and Visual Basics for Applications (VBA), designed to evaluate different Johne's disease control programs on their epidemiological and economical effects. From May 1998 until May 1999, January until June 2000, and May until July 2001, the Wageningen Farm Management Group has developed this dynamic and stochastic simulation model at the Wageningen Agricultural University. From June until December 1999 and June 2000 until April 2001 some modifications and extra options were built in to reflect the Pennsylvanian condition at the Center of Animal Health and Productivity, New Bolton Center, University of Pennsylvania.

More documentation about the calculation and application of the model can be obtained from several scientific papers and proceedings:

H. Groenendaal, M. Nielen, A.W. Jalvingh, H.S. Horst, D. T. Galligan and J. W. Hesselink. Johne's disease control: a simulation approach. Accepted by Preventive Veterinary Medicine

H. Groenendaal, M. Nielen, and J. W. Hesselink. Development of the Dutch Johne's disease control program supported by a simulation model. Submitted to Veterinary Quarterly

H. Groenendaal, M. Nielen, and J. W. Hesselink. Development of the Dutch Johne's disease control program supported by a simulation model. Proceedings SVEPM, March 2001

H. Groenendaal, and D. T. Galligan. Economic consequences of Johne's disease control programs on Pennsylvanian dairy herds. In: Proceedings of the International Conference on Dairy Science/ASAS, Indianapolis, July 2001

H. Groenendaal, and D. T. Galligan. Economic consequences of Johne's disease control programs on Pennsylvanian dairy herds. In: Proceedings of the 101st annual Penn Conference, Philadelphia, January 2001

H. Groenendaal, A.W. Jalvingh, M. Nielen, H.S. Horst. A simulation model to study the epidemiological and economic consequences of Johne's disease control programs. Proceedings 9th International Symposium ISVEE, August 6-11, 2000, Breckenridge, USA, 590-592

H. Groenendaal, A.W. Jalvingh, M. Nielen, H.S. Horst. A pilot study on paratuberculosis control on dairy farms in the Netherlands: evaluation of strategies using simulation. Proceedings (11) stuiedag Veterinaire Epidemiologie en Economie (VEEC), Bilthoven, December 9 1998. A.M. Henken en E.G. Evers (eds), 73-74

3.2. How to use this manual?

In this manual the JohneSSim model will be described in detail, to allow other users to use it. This user manual for the JohneSSim model is divided into two main parts.

First in chapter 4 a description how to use the model will be given. This includes the different steps that need to be taken to run the model to simulate a specific control program. It also includes some on how to use the model in the most efficient way.

Secondly, in chapter 5 the input data are described extensively. The input data is mainly concentrated in one (Excel) spreadsheet (“Inputs”), but additional input data in other sheets will also be covered.

3.3. Overview

3.3.1. Stochasticity

The simulation model is a so called ‘Monte Carlo’ model in which stochastic events find place randomly by using different probability distributions. The stochastic events are of influence on the outcomes for individual animals (e.g. become infected or not, be culled or not) hence for the outcomes on herd level. Because of the stochastic character of the model the variation of the results in the simulated process are taken into account (both good- and bad case scenarios are simulated). To get insight in the variation of the results, a big number of runs (iterations) of the simulation model. From those outcomes, estimates about the average and percentiles are calculated.

3.3.2. Time steps

The JohneSSim model works with time steps of *half a year* and the basic entity of the model is an animal. In other words, animals are followed in time and for each animal it e.g. age, production level or infection status will be simulated in time. Therefore, the model can be called a dynamic model; it contains time as a variable and can simulate the behavior of a system over time.

3.3.3. General overview

By categorizing farms on their management, herd-size and Johne’s disease status (infected vs. free), many possible situations can be modeled. In the model different control strategies can be simulated during a certain number of years, by default a 20 years period. In general, control tools can be divided into two categories, (1) tests and cull strategies and (2) management strategies. Because of the ‘flexibility’ of the model many different strategies can be simulated relatively easy, by changing input data. By aggregating (assimilating) the results on farm level

of many different dairy farms, we can get insight in the results on e.g. 'farm type' level, 'infection status' level, State level or National level. The simulation model exists of four separated files, which all have to be used to run a simulation:

1. Herd dynamics
2. Disease dynamics
3. Control strategies
4. Economics

Because the four parts are separated, their input in the main screen will be described separately in chapter 5.

4. Using the Johne's model

4.1. Before you begin

To run JohneSSim you need:

- A Pentium or faster PC,
- Excel 97 or higher,
- The JohneSSim model (which exists out of 4 Excel-files)
 1. The Input-Simulation file,
 2. The Input-Simulation code,
 3. The Assimilation file,
 4. The Assimilation code,
- A printer (optional).

In summary, the next steps need to be taken to simulate **one** control strategy:

- a) Generalize a start herd (.stv-file) (contains variation between iterations);
- b) Simulate a farm without control (.rst-file);
- c) Simulate a farm with a certain control program (.rst-file);
- d) Run the sector-level model to get the results and economic benefits

In the next four paragraphs, all four steps will be described in more detail.

4.2. Result-files

The JohneSSim model calculates the epidemiological and economical effects of a specific control program, starting from the current (e.g. Dutch) farm situation. Therefore (1) a current farm situation has to be generated and (2) a strategy of no control has to be simulated to make an economic comparison.

(1). The file with the current farm situation (on $t=0$) has an extension ".stv", which stands for "stardveestapel" or "startherd". The .stv-files are generated by the model (see below for a description how).

(2). To calculate all economic parameters (e.g. reduction in losses of Johne's disease), a simulation without control has to be simulated too. Both the control strategies and the strategies without control are called "rst" (results) files.

4.2.1. Generate start herd (.stv-file)

To simulate a specific control program the next steps need to be taken:

- 1) Generate an InitialHerd
- 2) Generate a .stv (Start Herd) file

Both procedures are explained step by step below.

4.2.1.1. Generate a new InitialHerd

An InitialHerd of a specific size and type must be made. This herd is then stored in the sheet "InitialHerd_x_y". "x" Stands for the TypeOfHerd [there are four types, 1 = dairy, 2 = Suckling-Breed, 3 = Suckling-Mix and 4 = Suckling-Meat (C133): the main difference between the three kinds of Suckling herds in the model are the replacement policies]. "y" Stands for the Size of the herd, the model will make an InitialHerd with the herd-size (C134) that is specified in the input sheet (D2-D29, Sum in D30-32). The model will therefore write the InitialHerd to the sheet that is specified with the "Type"(C133) and "Size" (C134). For dairy herds, the default input is x = 1 (C133) and y = 1 (C134). The generation of the InitialHerd involves the next steps:

- Defining the number of animals in each age group (D2-D29, Sum in D30-32);
- Changing Cell H38, "New Herd?" to 1
(Don't forget to change this back to "0" before simulating);
- Adjust the minimal and maximum number of cows in the herd (G130 & G131), the minimal # of cow is a limit under which a herd is going to buy heifers, the maximum is the maximal # of cows (> 2 yr.) possible in the herd;
- Run the model for a few seconds by (goal = write new herd in the InitialHerd-sheet):
 - Go to "Tools", "Macro", "Macro" (or push Alt + F8),
 - Select the Macro "Main.Main",
 - Wait a few (about 10 seconds) seconds,
 - Stop the simulation by pressing the "Escape"-button and then "Stop",
- Now you can check the generated herd in the sheet 'Initial_Herd_x_y';
- If the Herd should include an infected (or more) animal(s), we have to manually:
 - (1). Change the 'Time infected' to 0;
 - (2). The 'Highly inf' to 5;
 - (3). The 'Clinical' to 2;
 - (4). We can do this for as many cows as we want to.
- Don't forget to set Cell H38 for "New Herd?" back to 0!!

The Herd, which is displaced (just made) in the sheet 'Initial_Herd_x_y will now be the starting herd to start a new simulation. This starting herd will be used to make ".stv-files". On default, the same Initial_Herd is used to generate different ".stv-files".

4.2.1.2. Generate new start herd-files (.stv-files)

In contrast to the Initial_Herd-sheet(s), .stv-files (flat files with just 0/1 code) are “farm-profile” and Johne’s status specific; a .stv-file represent a herd of a specific “farm-profile” and “Johne’s status”. A “farm-profile” represents a specific management (related to Johne’s disease) – in the original Dutch study 8 profiles were defined. In addition, different “Johne’s statuses” can be defined – in the original Dutch study, 3 statuses were defined, (1) infected and suspected, (2) infected and unsuspected and (3) free (both unsuspected and suspected). Each .stv-file is a combination between one profile and one status, so for the Dutch situations, 8 times 3 = 24 .stv-files were generated to represent the Dutch dairy industry.

One new .stv-file can be generated by:

- Change the “Save HASC” (Save Herd At Start Control) Cell (A179) to 1;
- Change the “Read HASC” (Read Herd At Start Control) Cell (B179) to 0;
- Now the model saves a start herd in a file with the file-name showed in C179, so in this Cell the name of the .stv-file should be written (take a logical name!);
- To generate a .stv-file, the herds should meet certain criteria. Therefore, “SaveResultsAlways, Cell D40 should be *False* when generating a .stv-file with specific characteristics. Two criteria can be used:

1. ‘Prev’ (specific test prevalence), Cell C183 = ‘Prev’:

During the generation of a starting herd, the model simulates a herd until the *test* prevalence (and NOT the real prevalence) is between a certain minimum and maximum (the test that is used is specified in Cell C184, default test = 1 = Elisa test). This test-prevalence should represent the desired start situations. In Cell D185 and E185, the FirstClassNr and LastClassNr are specified that determine the Prevalence Classes that are show in the table H174 – Z184 for dairy farms (‘Type 1’). The same ‘Prev’ table for ‘suckling herds’ (Type of farm = 2-4) is shown at the right side of the table described above, in Cells AA174-AS184. The next parameters are shown in this table:

- 1) Number of subclasses in a certain class, in Cells L174, L177, L180 and L183. The classes and sub-classes are made to enable the specification of narrow – ranges in which the test-prevalence should be to meet the criteria;
- 2) The minimum and maximum prevalence in the certain subclass, in e.g. Cells M174-N174, P174-Q174, etc. that all Subclasses 1, 2, etc. have to meet to be included in the .stv-file. For example, Cell M174 and N174 show 0 and 0, which means that the herds should have a test-prevalence of minimal 0 and maximal 0 to be included (these are the unsuspected herds). If an iteration is not selected, it will be not included in the .stv-file but deleted;
- 3) The number of iterations of each subclass, in e.g. Cell O174, O177, R174, etc.
- 4) Finally, in Cells K174-K184, the minimal and maximum number of time-steps (half years) is specified, between which the specified test-prevalence should be met.

“DoTests?” (Cell C201) musts always be FALSE if a .stv-file is made. The Backgroundtest is used to monitor the test-prevalence! In addition, Cell F184

(FarmMustBeInfected) is a variable that forces (if TRUE) each farm that is selected in the .stv-file to be infected (this can only be used when the criteria is 'Prev'). If this Cell is FALSE, then also farms without any infected but that have one or more test-positive (false positive in this case) cows can be selected. Because of the specificity, which is not equal to 1, a farm could potentially have no Johne's disease but still be test-positive. When this cell = 'True' a herd with a certain prevalence *must* have one or more infected animals to be selected in the stv-file.

2. '**Time**' (specific time), Cell C183 = 'Time'.

This is used if we want to simulate a control strategy that start e.g. 10 years after the initial infection. In this case, we don't have any control over the test prevalence that the farm has at the moment it is selected in the .stv-file. In this case, Cell D40 must be 'True' because we select every herd after x half years.

- Number of iterations:

a) If '**Prev**' then the maximum number of iterations (Cell D39) must always exceed the number of needed iterations (so normally D39 = 1000 or 2000 if 100 iterations are needed) because not all iterations will meet the criteria to be included in the .stv-file.

b) If '**Time**' then the maximum number of iterations (D39, then default = 100) is how many we get. Cell D40 (see above) needs to be 'True' in this case.

4.2.2. Simulate strategies (.rst files)

To start the simulation of a farm with any strategy (with or without control) the next steps should be taken:

1. We should now use the .stv-file (flat files with just 0/1 code) as a starting situation. The exception is if we simulate a free herd. For the free herds we can (1) use a .stv file as described above or (2) we start with an InitialHerd, which is displayed in the InitialHerd-sheets. The second option is possible with free herds, because a free herd does not have to meet certain test-prevalence criteria, but just has to be free.
2. To use a .stv-file as the start situation, in cell C179 the name of the specific StartHerd (e.g. StartHerd_x.stv) must be written. This .stv-file must correspond with a .stv-file that is already made. In Cell C178 the name of the strategy that is simulated, must be specified (e.g. Strategy_x.rst). Furthermore, it is important that cell B179 (ReadHASC) will be changed back to 1 and A179 (SaveHASC) to 0.
3. The procedures to simulate a farm with or without a specific control program are the same. The only different is that a specific control program needs to start at $t = 1$. The different options, available to modify and simulate a certain program are described in paragraph "Control of Johne's Disease".
4. For all herds we are simulating, we have to make sure that the management ("related to the spread of Johne's disease") reflects the situation on the farm **at the moment of starting** (in the .stv-file) of the control program. **After** the start a specific control program (or without control) we can change the management (this is a potential control-strategy) by modifying

the control parameters. In that way, the management-related disease routes will be changed (e.g. start giving the calves milk-replacer will eliminate the infection route 'whole milk').

4.3. Simulation of more profiles/statusses

To make the generation of different .stv files or the simulation of several scenarios faster and easier, the next advices are given:

- 1) If saving a Input-Simulation file, give it a logical name (e.g. StartHerd_1.xls or TypeOne_StatusA_Testing.xls),
- 2) Give the .stv-files a logical name too, which is the same (except for the extension) as the .xls file.
- 3) After saving all Input (.xls) files, which must be saved in one folder, open the Simulation code-file,
- 4) Put all the file names (excluding the .xls extension) in a column, starting in Cell A6 in the file "Simulation_Code",
- 5) Make sure that, when pressing the open-button, the Input (.xls) files are seen. In this way, the model can find these files.
- 6) Push the button "Start" and the model will start opening all subsequent Input (.xls) files, simulate them, save the .stv-files, close and start the next one.
- 7) You can always check the status of the simulation (how far is the model?) by pushing the 'Esc' button on your p.c.
- 8) The same procedure can be followed for .rst. files, but make sure that first the .stv-files are generated!

4.4. Assimilation of results

In the Assimilation model, the result-files (.rst-files) of one or different simulated farm-types or -situations can be assimilated and can be read. Also the Assimilation part of the model exists out of two files, an Input-Output file and a file with the code. After running the assimilation procedure, we obtain a summary of the simulation in one single sheet, the 'OutPutSectorTable-sheet'. More detailed results of the simulated herds are available in other sheets 'of the Input_Output file'.

To read the output files (.rst files) from the simulation model, their names (*including* the .rst extension) need to be in the 'InputSector' sheet. In the Column starting with the cell C6 we need to write the .rst-files with a certain control program. In the Column starting with the cell D6 we need to fill in the files without control but with the same 'farm type' etc. This enables the model to compare the economic losses without and with a certain control program. Furthermore, in cell C3 we need to write how many different farm types (how many .rst. files) we want to assimilate starting with the farm type in Row 6.

The **number of iterations**, used from each of the management types, is shown in cell F6 and under. The number of iterations of a certain farm-type should be equal or higher than the number of iterations (default 100) in the simulation model. By assimilating a certain number

of each of the farm-types (for instance FarmOne.rst - 100 iterations, FarmTwo.rst - 400 iterations and FarmThree.rst - 500 iterations) we get the results were a certain percentage of all farms has a specific management type. In this example 10% of the farms is of the 'farm type' 'One', 40% of the type 'Two' and 50% of the type 'Three'. Different infection statuses (e.g. free and infected) and farm profiles can be combined in this way.

In the 'InputSector' sheet, Cell H1 in the Assimilation-file we can specify what percentage of management costs we want to include in the Cost Benefit Analyses. In addition, in Cell H2 we can specify if we want to include the costs and benefits of 'heifer contract rearing' in the Cost Benefit Analyses (True or False).

4.5. Assimilation of more files at once

With the Assimilation_Code, also more then one assimilation at a time can be done. This is done in the same way as simulation of more then one strategy at a time (see paragraph 4.3). In the sheet "Batch" of the file "Assimilation_Code", we fill in the names of the different Assimilation_Input_Output files (without .xls extension) that the model needs to run. We do this in Cell A6 and under. Make sure again that when the model wants to open those files, it can find the files (when pressing the open-button, the Assimilation_Input_Output files can be seen immediately). In addition, all the .rst files that are needed (the results!) need to be in the same file.

Finally, you can now also see how far the model is (how many Assimilation_Input_Output files the model has finished already).

5. Main input menu

5.1. General

The input describe below has to be used for both (1) making .stv files and (2) making .rst files. The input is used to simulate a dairy herd including young-stock for an x number of years under specific (depending on input) conditions.

5.1.1. Time steps

The number of time-steps, during which the model simulates a farm depend on one of the following inputs:

1. TimeAfterStartControl (Cell D37); this cell is used if we start the simulation with a specific Start Herd (we read the .stv-file). This parameter stands for the number of years
2. Time Periods (half years) (Cell D38); this cell determines how many half years we simulate a herd if we are creating a Start Herd (.stv-file) or if we simulate a strategy, but we did not start with a Start Herd but start with the herd, shown in sheet 'Initial_Herd_x' (like we can do with for example free herds, as explained in paragraph 4.2.1).

5.1.2. Seeds

There are several 'seeds', which are used in the model. The seeds are used as a start number for the random number generator (see for the function of seeds the book of Vose (2001). By changing the seeds, the sequence of random numbers will change. The default values are:

- 1) Herd-Seed (Cell H37) = 0; used for the herd dynamics, e.g. involuntary culling,
- 2) IterSeed (Cell H38) = 1; used for iterations, e.g. 'closed' or 'open' farm,
- 3) Seed (Cell H39) = 1; seed that is not (yet) used at the moment.

5.2. Dairy Herd simulation

5.2.1. General description

The basic unit of the JohneSSim model is one dairy herd. The model works with a standard herd of x cows plus additional youngstock. Heifers do calf at an age of two or three years (Cells D136-G136); the calving interval is always one year.

5.2.2. Related input in the model

5.2.2.1.Herd type and size

In the model it is easy to switch from different herd type and sizes (different InitialHerd-sheets, which are generated before, see 4.2.1.1) because of the option 'Type of Herd' and 'Size of Herd' in respectively Cell C133 and C134. Both inputs determine which data will be used for:

- The start herd if we read it from an 'Initial_Herd_x'-sheet;
- The minimal and maximal herd size.

The 'Type of the Herd' will determine the next parameters (everywhere, there are four inputs in a row for the same parameter):

- Age at first calving etc;
- Culling procedure (see culling);
- Milk fed to the young calves etc.

If the 'Type of the Herd' = 1, the most left input will be used in the simulation, if type = 2 the second, etc. The type of herd can be e.g. a dairy herd or a specific 'suckling herd' (three different 'suckling herd types' were specified in the Netherlands).

5.2.2.2.Herd size

Start

The herd size in the **start** of the simulation is determined by one of the next two factors:

- If we generalize a start herd file (.stv-file) the start herd is used, which is shown in the sheet 'Initial_Herd_x'. In other words, the simulation model starts with this herd (when generating a start herd file, the herd-size will not increase (ignore Cell G132)).
- If we simulate a certain control program and therefor use a start herd (.stv-file), the herd size is determined by the herd size in the .stv-file.

During

However, in both cases the herd size **during** the simulation is determined by the minimum and maximum herd size, shown in Cell G130 and G131. In addition, both parameters (min, max) can increase with a percentage (G132) per year.

5.2.2.3.Culling

The culling in the simulation model can be divided in (I) 'involuntary' and (II) 'voluntary' culling:

- I. The '**involuntary culling**' percentages (R2-U29 for different ages – vertically- and different herd types – horizontally -) for adult cows are used first in the culling procedure in the model. The percentage calves born alive is shown in Cell N35 and N36, different for first calving heifers and older cows. The percentage of heifer calves (default 50%) is shown in Cell N37 and the percentage of heifers we keep is shown in Cell N38. At a default age of 2 years the surplus of heifers will be sold.
- II. After the involuntary culling has taken place, the '**voluntary culling**' is depending on the RPO-values (the total extra profit to be expected from trying to keep the cow until its optimal life-span, compared with immediate replacement, taking into account the risk of premature removal of retained animals) of the cows. The voluntary culling will

be used to cull low-producing and older animals until the maximal culling percentage (Cell G127) is reached or until the number of animals \leq "maximal number of animals in herd". This percentage is a yearly percentage, but can be checked every half a year or every year (depending on Cell G126), the default is every year (so the input in Cell G216 is "2" half years!). The first time step assumes that the time step before the percentage shown in Cell G128 of G127 is culled (e.g. 50% of the maximal culling rate of 30% -default- per year). *Three* different RPO-values can be used in the model, depending on the situation:

- Dutch RPO values without alternative use of e.g. labor (I39 = False);
- Dutch RPO values with alternative use of e.g. labor (I39 = True);
- U.S. RPO values (I40 = True, if this is TRUE, it overrules the Dutch RPO values).

The voluntary culling takes place after all involuntary culling of the last $\frac{1}{2}$ year has been determined. In other words, voluntary culling will only take place, if there are still too many animals left after the involuntary culling procedures.

The simulation selects the RPO of each individual cow, depending on her (1) lactation number, (2) month in lactation culled and (3) milk production. Voluntary culling will take place in the model, depending on their RPO value in a certain month of the lactation, shown in Cell M47 and culling will take place in the month, shown in Cell N47. On default, Cell N47 = 7, because that is around the months that the RPO is minimum, assuming a 12-month calving interval.

The losses, of premature culling, can be divided into two categories:

- Losses when culling a clinical diseased cow (involuntary). Those losses are equal to the RPO value of the cow;
- Losses when culling an infected diseased cow because of a low milk production (voluntary). If a cow is culled because of a low production, the losses are equal to the RPO value of the cow with her 'normal production' minus the RPO value with her 'actual production'. Because her 'actual production' can be reduced due to Johne's, these losses must be included in the losses due to Johne's disease;

III. A second voluntary culling procedure has been added to reflect the culling procedures on '**suckling farms**' in the Netherlands. This procedure will take place after the involuntary culling and keeps the herd size constant. The general steps of this culling procedure are:

- Newborn calves are randomly assigned a 'excellent breeding' or 'normal-breeding' label. In other words, the model decides randomly for each young calf if it will be an 'excellent breeding' calf or not.
- All heifer calves are kept at least for one calving;
- Each time step (half year) first the involuntary culling takes place;
- Then the model voluntary culls all 'normal-breeding' cows, starting with the oldest until the herd size is constant again. These 'normal-breeding' are in this way slaughter for an optimal price and the month of slaughter is shown in Cell D154-G154.
- 'Excellent-breeding' cows are kept until they are involuntarily culled.

This procedure will be used in the model if Cell D144-G144 is True (depending on the 'Type of the Herd' which Cell). The rest of the input is shown in Cell D146-G152. In the latter table, the percentages of 'excellent-breeding' animals is shown, both for the starting herd (D146-G151) and for the calves during the simulating (D152-G152). The RPO values of the 'excellent-breeding' suckling cows are shown in Cell F354-I368. These RPO values are only used to determine the losses if culling a clinical cow or the costs of culling a test positive cow.

Last but not least, the **maximal age** of cows in the model is shown in Cell D137-G137.

5.2.2.4. Calving pattern and age first calving

Calving pattern

The calving pattern in the herd can be 'spread' (every half year cows are calving) or 'concentrated' (every second half year cows are calving). The default pattern is 'spread' and to make the pattern concentrated, the Cells D142-G142 should be 'True' to make the pattern 'concentrated'.

Age first calving

A second related input is the age at first calving (Cells D136-G136). The default age is 2 year (input = 5) but also 3 year (with suckling cows, input = 7) is possible.

5.3. Disease and management

5.3.1. Overview

In the model, several infection routes do simulate the transmission of Johne's disease within a herd. In addition, infected animals can also be introduced in the herd. The next infection routes are available in the JohneSSim model:

1. Fetal infections;
2. Infections around birth;
3. Infections due to drinking mixed colostrum;
4. Infections due to drinking colostrum of the own dam;
5. Infections due to drinking pooled bulk milk;
6. Infections due to drinking rest milk;
7. Infections due to suckling milk from other cows;
8. Infections due to environmental contamination;
9. Infections due to introduction of an infected animal in the herd.

The infection routes, which are present on the different dairy farm, are different because of different farm- and management types. The input, which is related to the disease transmission routes, is depending on the management on the dairy farm. If e.g. the farmer feeds whole milk to his calves, this route must be enabled. The input of the different infection routes is

explained in the next paragraphs. Also the disease dynamics of infected animals is explained in relation to the input of the model.

5.3.2. Fetal infections

The input, related to fetal infections, is shown in Cell C51 to F51. The infection probability is depending on the infection status of the mother of the calf. The values shown in the 'Input' sheet can only be changed by another input, not by a control program (in contrast to the rest of the infection routes).

5.3.3. Infections around birth

The input of the infections around birth is shown in (1) Cell D55 and D56 and further and (2) in Cell C61-E66. Both tables show the infection probabilities under different infection situations on a dairy farm.

- 1) The first Table (D55-56 and further) shows the infection probabilities of a calf with a low infectious dam (C). Depending on how many time steps (half years) the dam is before she will become highly infectious (D) the infection probability increases.
- 2) The second Table (C61-E66) shows the infection probabilities in all other situations. The probability depends on both the infection status of the dam and the rest of the herd (dam status in rows, herd situation in columns).

Important is that if a farmer improves hygiene and management around birth, we should NOT change the parameters in both Tables but we need to change the parameters of the "control of Johne's", see below for more information about this.

5.3.4. Infections due to drinking mixed colostrum

This infection probability is depending on several input variables, which are shown in Cell C70-E71. In addition, Cell F83 and F85 determine the management on the farm. Cell F83 shows the number of calves that get colostrum from one cow (because of pooling colostrum) and Cell F85 shows if the farmer actually feeds colostrum. So, Cell F85 needs to be changed if we want another 'colostrum management' on the farm before we start a control program.

If mixed colostrum is fed to calves, also the colostrum of the own mother is fed to the calf.

5.3.5. Infections due to drinking colostrum of the own dam

For this infection route, the same input variables (C70-E71) are used as for the pooled colostrum. However, if $F85 = 0$, no colostrum will be mixed but only colostrum from the own dam will be administered (if $F85 = 1$, mixed colostrum will be administered).

5.3.6. Infections due to drinking pooled bulk milk

Three situations with respect to milk feeding on a farm are available. The three situations can be changed in Cell F74-76, only one of the three can be equal to 1, the other two must be 0!

- 1) If C74 = 1 then the farmer both feeds pooled bulk milk **and** rest milk;
- 2) If C75 = 1 then the farmer both feeds milk replacer **and** rest milk;
- 3) If C76 = 1 then the farmer feeds only milk replacer (no infections via milk).

Bulk milk can only be infected if one or more infectious cows spread a large amount of *M. paratuberculosis* bacteria in their milk. The probability on infected bulk milk depends on the probability of each highly infectious cow to have large amounts of *M. paratuberculosis* in the milk, which is shown in Cell F84 (this is a probability per cow). The infection probability if the pooled bulk milk is infected is equal to the value shown in Cell F77 (default 95%). The infection probability due to drinking rest milk is explained in the next paragraph.

5.3.7. Infections due to suckling at more cows

This infection route is of importance if we simulate a 'suckling herd' where calves stay at their dams for some months. To enable this infection route, one of the Cells D139-G139 must be 'True'. In this case, some calves (x percent, input in Cell D140-G140) will drink milk at other calves. If a calf drinks milk at other cows (suckling herds), the average number of cows where it drinks is shown in Cell D141-G141.

5.3.8. Infections due to drinking rest milk

For this infection route, a number of parameters are important. The infection route is only in use if the farmer feeds rest milk (Cell F74 or F75 = 1).

The infection probability depends on:

- 1) The probability that a cows gives rest milk per lactation (Cell F78);
- 2) The number of calves that get milk from one cow with rest milk (Cell F79).

Furthermore, the number of infectious cows is important for the infection probability, but this number is determined by the simulation model (depends on the results).

5.3.9. Infections due to environmental contamination

For this infection route, the input is shown in Cell D100 – K103 and the Reed Frost method is used (see paper for more information and references). An important input parameter for this infection route is 'the number of effective contacts (k) between calves and adult cows'. This number can depend on (1) the season and system in which the calves are kept (barn or pasture) and (2) the age of the calves. For young (0-6 months) and old (7-12 months) k is shown in Cell D100 – E102.

Furthermore, the susceptibility influences the effectivity of contacts of older cows. Two approaches can be used for the decline in the susceptibility, a linear (Cell N97 = True) or an

exponential (Cell N101 = True). The rate of decline can be regulated by respectively Cell N98 and Cell N102 (they determine the rate of decrease).

5.3.10. Infections due to introduction of an infected animal in the herd

Introductions of infected animals can only take place on 'open- farms'. It is randomly determined if a specific farm (iteration) is 'open' or 'closed'. The percentage of 'open' and 'closed' herds at the start is shown in Cell E339. Because this percentage can change in time, the increase (-/- if decrease) is shown in Cell F339. Once a farm (iteration) is closed, it will be closed the rest of the simulation (rest of the default 20 years).

If a farm (iteration) is 'open' there is a probability on an introduction of a Johne's disease infected calf/heifer/cow (with probabilities of respectively D344, D345 and D346). The probability that each of those animals is infected can change (increase or decrease) in time. The start probability is shown in Cell E340, the increase (-/- if decrease) is shown in Cell F340. However, for calves, heifers and cows, the probability can be differently, and therefore they are multiplied by a factor, which is shown in Cell F344-F346.

For different reasons, a shortage of heifers can occur. In that case, the heifers we buy can also be infected. The probability that each heifer is infected is shown in Cell E341 and F341.

5.3.11. Infection course of infected animals

An infected animal can become low- or highly infectious or clinical infected sooner or later. The process, which determines for each animal the age when this will happen, is explained in this paragraph. The next steps are important for each *infected animal*:

- 1) After an infection, first the age that the animal will become '**highly infectious**' will be determined randomly. A triangular distribution is used, the minimal, most likely and maximal ages are shown in table of B92-E95. The parameters of this distribution depend on the age when the animal is infected (A92-95). There are four ages distinguished, (1) before birth = fetal, (2) around birth, (3) from week 1 until month 6 and (4) from month 7 – 12.
- 2) A certain number of lactation (Cell D88) before an animal becomes highly infectious, it becomes '**low infectious**'. The minimal age to become 'low infectious' is shown in Cell D87. In this 'low infectious' period, the animal spreads *M. paratuberculosis* only x month (Cell F111 and G11) per half year, depending if it is the first or second half-year after calving. If a 'low infectious' animal spreads *M. paratuberculosis* in it's faces, the infectiousity, relative to 'highly infectious animals' is shown in Cell F89.
- 3) After an animal has become highly infectious, it can become **clinical**. The length of the period between becoming highly infectious and clinical is also modeled with a triangular distribution. The min., most likely and max. length is shown in Cells B96-D96 (in months!).

- 4) Once an animal is clinical infected, it will not stay a long period in the herd. The length of this clinical infected period is modeled with a triangular distribution too, the parameters are shown in the Cells F114-H114 and if a control strategy is used F115-H115.
- 5) In addition to this, farmers can already **cull the last x** calves of each cow that becomes clinical. The number x is shown in Cell F116 and reflects the situation *before* starting a control program. In Row 219 the number of calves culled *during* control is shown.

5.4. Control of Johne's disease

5.4.1. Overview

In general there are three categories of control tools against Johne's disease in the model:

- 1) 'Test and cull' strategies;
- 2) Management strategies;
- 3) Vaccination strategies.

Test and cull strategies can vary in many ways. Tests, frequency, culling, costs etc. can all be different. The test attributes will be shown in the next paragraph. Test strategies can be simulated in two different ways, both will be explained below. Furthermore, the management strategies and the options related to this are shown and last but not least, the options related to vaccination are shown in the last paragraph.

5.4.2. 'Test and Cull' strategies

In all cases when test are used during a control program, Cell C201 must be 'True'. To initialize the different test, the first and last test-code in each category should be shown in Cell C202-D206.

There are five different tests in the model:

- 1) Background test;
- 2) Start test;
- 3) Test during (same as confirmation test);
- 4) Surveillance test;
- 5) Confirmation test.

The data of all the tests is showed below in the 'Input'-sheet.

The Background test is only used to determine the test prevalence of a herd to check if the test prevalence meets the minimum and maximum, which is required to start a control program. The other four tests can be used on different times during the program, depending on the status (level) of the farm and the frequency of testing. This is explained further in the text below.

5.4.2.1. Test programs

There are two procedures to simulate test strategies:

- 1) Using the **frequency of the tests** (test = depending if farms is suspected or unsuspected)

If this test program is simulated, always the 'Start test' will be used first. The year in which the first (Start test) is done has to be specified. Two different ways are available:

- (1) If Cell F201 = True, then we can fill in the Min, Most Likely and Max 'first year' (Cell D203-G203) in which an iteration starts testing with the Start Test (this is stochastically modeled with a triangular distribution).
- (2) The second way is if Cell J201 = TRUE. Now we can exactly specify what percentage of farms will start in which year (Cell I230 and further to the right). To start all herds with the Start Test in year 0, we can fill in 100% at year 0.

Which one of the two is used, depends on the number in Cell C202 and C203. After the 'Start test' the 'Test During' will be used if one of the animals was positive (confirmed) so the herd is suspected. The 'Surveillance Test' will be used if there are no positive animals anymore in the herd.

- 2) Using a **test-track** (test = totally depending on ParaStatusAccent of the herd)

The USA test program is different from the Dutch program because test are used in different orders and in addition to this, not all animal are tested sometimes. If we want to simulate one of the three "Testing-Tracks", Cell E287, I287 or M287 must be 'True'. This overrules the programs, which are simulated with the frequency (see above, number 1).

To simulate different test programs, the input shown in Cells D290 – R325 can be changed. For each track, three different inputs can be distinguished. First, (the first column) the number of years a herd stays in a specific ParaStatusAccent before the next testing. Secondly, (second column) the number of the test (the tests, which are used are the Surveillance tests!!!) that is used to go to a next status. Thirdly (in the third column) the number of cows that needs to be tested with the test. If this is smaller than the number of cows that can be tested in the herd, on default random animals will be picked out. Two other options exist (test only the oldest or youngest) however, they can only be changed in the Simulation_Code. This can be done by change in the Module "Control" the parameter "ActiveControl.Track.WhatAnimalsToTest(ParaStatus)" from "Random" to respectively "Oldest" or "Youngest".

In addition, for the three different Test-tracks there are three inputs (Cells P290-R335), which are used always if one of the three tracks in used. The first column is the number of animals that is allowed to be test positive to be able to still go the ParaStatusAccent that is shown in the next column (Column Q). The third column shows the ParaStatusAccent in which a herd will come if the number of test positive animals > the number of animals that is allowed (Column P).

5.4.2.2. Test attributes

The tests, used in the model have many different attributes. The attributes determine e.g. the performance of the test, when the test should be used and what the costs are. All attributes are explained below:

- 1) **Test Code (row 209):** this is just for simulation purposes, so the model knows which test to use (each test now has a unique number in the code).
- 2) **Sensitivity (row 210-213):** the sensitivity is specified for animal with different infection statuses in the model.
- 3) **Specificity (row 214-215):** the specificity is specified for both A (younger than 1 year) and B animals (> 1 year), both none-infected animals.
- 4) **Minimal age of testing (row 216):** this is the minimal age that an animal can be tested, mostly 2 or 3 years (in the model respectively 5 or 7 half years!).
- 5) **Maximal age of testing (row 217):** this is the maximal age that an animal can be tested.
- 6) **Cull if positive (row 218):** this determines if an animal will be culled directly after a positive test or not (maybe a confirmation test is necessary, see row 223).
- 7) **Cull number of Calves (row 219):** this is the number of calves that will be culled too if the dam is culled because of a positive test result. The costs associated with culling calves (before first time calving) are shown in Cells G248-L251.
- 8) **Month until result (row 220):** this is the number of month we have to wait before the test results will become available and we can cull the cow.
- 9) **Frequency (row 221):** this is the frequency of the test per year, so if we do the test once a year it is 1, if we do the test once in the two years it is $\frac{1}{2}$.
- 10) **Confirmation test number (row 222):** this is the number of the confirmation test (Test Code) if a confirmation test is needed.
- 11) **Cull on confirmation (row 223):** will an animal be culled if the confirmation test is positive (True or False).
- 12) **Background (row 224):** are we dealing with a background test or not?
- 13) **Start test (row 225):** are we dealing with a test, which we use at the start or not?
- 14) **Surveillance test (row 226):** are we dealing with a surveillance test or not?
- 15) **Pooled test (row 227):** are we dealing with a test where we pool the samples?
- 16) **Number of animals in one pool (row 228):** number of animals pooled together.
- 17) **Costs per test (row 229):** what are the costs per test (exclusive vet costs)?
- 18) **Visits per test (row 230):** how many times does the vet need to come for this test to test all animals?
- 19) **Vet costs per sample (row 231):** what are the veterinary costs per sample?
- 20) **Test only part of the animals (row 232):** test only a proportion of the cows or all? This is only of importance when using the frequency of the test (see paragraph 5.4.2) because if we use the test-tracks we use the Cells F290-F325!
- 21) **Test number of animals (row 234):** how many animals need to be tested?
- 22) **Whole Herd Confirmation test (row 235):** if one or more animals are positive or confirmed positive and this Cell = True, then we will use Confirmation test number 4 (always 4!!).

5.4.3. Management strategies

The input for the 'common management control tools' is shown in Cells C188-C200. In addition, input data about 'grouping animals' (used as a control strategy) are shown in Cells D165-F171. In the next paragraphs, these two management strategies will be explained in more detail.

5.4.3.1. Common management tools

In general, the management tools discussed in this paragraph do have a direct influence on the different infection routes. The following management tools are available:

1. Having a better **hygiene and management around a calf's birth** (e.g. separating the calf very quick from its dam). Changing the Cells C192-C194 from 1 to a fraction between 0 and 1 (percentage) reflects this. The original infection probability will be multiplied by this fraction (default is 0.1 which is equal to a 90% reduction).
2. Give **only colostrum from the own dam** (stop mixing colostrum). Changing Cell C188 from 1 to 0 initializes this control tool.
3. Give **colostrum replacer** instead of colostrum from the own dam. Changing Cell C189 from 1 to 0 initializes this control tool.
4. If we simulate a suckling herd, changing Cell D189 to 0 means that the calves **stop suckling at more cows** (e.g. the calves are separated from the dams and therefore they are not suckling anymore).
5. Stop giving bulk milk. Only **rest milk and milk replacer** will be given instead. Changing Cell C190 from 1 to 0 initializes this control tool.
6. Stop giving bulk- and rest milk. Now we will only give **milk replacer** instead. Changing Cell C191 from 1 to 0 initializes this control tool.
7. Have a **better hygiene during raising of young calves and separation** from older animals. For this, we have to change the Cells C169-D171 to a fraction between 0 and 1 (default is 0.1, which is a 90% reduction). Changing Cells C168 and C169 will only have an influence on the costs of better hygiene and separation; if the cells F196-G196 are 'True', the model includes costs for new housing of calves.

5.4.3.2. Grouping animals

Making two groups of animals, based on a test result or age, can potentially also influence the spread of Johne's disease within the herd. On default we only keep heifer calves from the 'none suspected' group as replacement heifers, and therefore 'grouping animals' could potentially be used to reduce the transmission of Johne's.

First of all, to start grouping animals, we need to change Cells D165-G165 to 2 (2 groups instead of 1). There are *two criteria* in the model, which can be used *to group animals*:

1. **'Age'** (Cells D166-G166): In this case, Cells D169-G169 ('Age to make the groups') will be the age when the animals will go to the older = 'suspected' group. Another option is to cull all the animals older than a certain age. To enable this option, we have to change 'CullGroupTwoAnimals' in Cells D170-G170 to 'True'.
2. **'Test'** (Cells D166-G166): In this case, Cells D167-G167 ('Teststart to make the groups') and Cells D168-G168 ('Testduring to make the groups') will determine which tests will be used to separate the animals in two groups.

In addition, to control Johne's disease more effectively, we could decide not to use replacement heifers from the 'suspected' (test positive or older) animals. In this case, we can sell all the heifer calves at an age 'Age of culling group-2 calves', shown in Cells D170-G170.

5.4.4. Vaccination strategies

The input data for vaccination are shown on two places. First of all, the main data are shown in Cells D156-G162. In addition to this, data on the effect of vaccination on the disease dynamics of vaccinated animals is shown in Cells F92-I96.

The next factors are important for a vaccination strategy:

- 1) Vaccination starts directly when the control program starts (only the newborn calves are vaccinated) if Cell D156-G156 is 'True'.
- 2) There are two criteria to stop the vaccination:
 - A **'Time'** criteria; in this case, we stop vaccinating after a certain number of years (Cell D158-G158).
 - A **'Prev'** (prevalence) criteria; in this case we'll stop vaccinating when a certain prevalence is reached (Cell D159-G159) and we use Background-test number 1 or 2 (Cell D160-G160) to determine this test prevalence.
- 3) The costs of vaccination per vaccinated calf are displayed in Cell D161-G161, and in addition to these costs, the costs of the veterinarian per vaccinated calf are in Cell D162-G162.

Vaccination does (in the current version of the model) only have an influence on the disease dynamics of infected animals. If a calf is vaccinated and infected, the age that it will become 'highly infectious' (D) will be higher than when the calf is not vaccinated. The data on the distribution of the age when an animal will become 'highly infectious' are shown in Cells F92-I96.

At the moment the model does not assume that vaccinated calves have a lower probability on infection. This can be added in the model, but studies show controversial data about this.

5.5. The economics

The economics in the JohneSSim model are:

1. The losses of Johne's disease;
2. The costs of the control of Johne's disease.

In the next paragraph (5.6. Assimilation of farms) the calculation of the reduction of the losses due to Johne's disease (benefits) will be explained. The losses of Johne's and costs of control will be explained below.

5.5.1. The losses of Johne's disease

1.1. Milk production reduction:

The reduction of the milk production due to Johne's disease (percentage of the normal production) is shown in Cell I47-L47, separated for different infection statuses. The average milk production in the herd is shown in Cell D34. The number of months that cows are in lactation is shown in Cell F80-I80 and the relative production in each month is shown in Cell P63-P86. In addition to this, each animal in the herd has a certain 'relative milk production (%)', which is randomly assigned at birth from a normal distribution (Average = 100%, SD = 10%). The average 'relative milk production' of newborn calves is shown in Cell D44 and of bought animals is shown in Cell D45. Finally, the losses per liter or lbs. of milk reduction are shown in Cell G236 (Price – variable or feeding costs).

1.2. Test of clinical cows:

If an infected animal becomes clinical infected, some costs can occur because of testing the cow or a veterinarian visiting. The average costs per clinical cow are in Cell G239 (not for all clinical cows, the vet needs to come).

1.3. Reduced slaughter value:

Infected animals can have a reduction in their slaughter value, depending on their infection status. The percentages are shown in Cells L237-O240. The normal slaughter values are shown on the right of this table, in Cells Q237-AD240.

1.4. Premature culling:

The losses due to premature culling have two reasons:

- Losses due to involuntary culling clinical cows;
- Losses due to voluntary culling infected cows with a lower milk production.

Depending on which RPO values are used, the losses will be calculated automatically (see voluntary culling).

All losses due to Johne's disease are calculated on a half yearly basis and in the 'Assimilation of farms' they are added to yearly losses. The total losses are then also discounted (standard with a 5% compound discount rate).

5.5.2. The costs of the control of Johne's disease

The costs of the control of Johne's disease are categorized in:

- Costs of tests,
- Costs of Animal Health Service accompaniment,
- Costs of culling cows (because positive test result),
- Costs of changing management,
- Costs of vaccination.

Tax

With all costs, tax are also included. Because of different tax percentages, used for different costs, there are two different tax percentages in the model. The first percentage (G277) is used for test and other GD (Animal Health Service) costs, and the second percentage (G278) is used for all other costs.

- *Costs of testing:*
The costs of testing all or part of the animals are calculated automatically if it is not a background test. In addition to these costs, the costs of a visit of a veterinarian have to be added, shown in Cell G240 and the costs of sending a the tests (per sending) are shown in Cell242.
- *Costs of Animal Health Service accompaniment*
If a control program starts (in the Netherlands), for which the Animal Health Service (called GD) accompanies the farm, yearly costs of this consultation of the GD are shown in Cell G241 and Cell G243.
- *Costs of culling cows:*
The costs of culling cows if they are test positive or if they are too old (if we decide to cull the oldest cows as a control tool) are equal to the RPO value of the cow. These costs will be calculated automatically (see also voluntary culling).
The costs of culling calves and heifers (if their dams are positive or if their dam is in a 'suspected' group) before their first time calving are shown in Cells G248-L251.
- *Costs of changing management:*
There are many different costs, associated with a change in the management. The next costs are included in the model:
 - **Costs of better hygiene and management around birth:**
Those costs are shown in Cell G254-G257. Cell G254 is the costs of 'cleaning apparatus', which are needed for a better hygiene. In addition, the costs of 'soap etc' are shown in G255. Thirdly, the extra labor hours due to cleaning the calving barn are shown in G256 and extra hours during each birth are shown in G257.
 - **Costs of giving colostrum replacer:**
Those costs are shown in Cell G258-G261. Cell G258 shown the amount of colostrum (in liters) used for each calf and Cell G259 shows the costs per liter of colostrum replacer. In addition, Cell G260 shows the number of extra labor hours

for giving colostrum replacer and Cell G261 shows the costs of transportation of colostrum replacer per calf (in the Netherlands the case).

➤ **Costs of giving milk replacer:**

Those costs are shown in Cell G262-G267.

1. Cell G262 shown the liters of milk replacer per kg of milk powder,
2. Cell G263 shows the costs of one kg of milk powder,
3. Cell G264 shows the number of liters of milk/milk replacer per calf,
4. Cell G265 shows the extra labor hours due to feeding milk replacer instead of bulk milk and rest milk,
5. Cell G266 shows the extra labor hours due to feeding milk replacer instead of rest milk and milk replacer,
6. Cell G267 shows the percentage of rest milk if a farmer feeds both bulk- and rest milk.

➤ **Costs of better hygiene and management during raising of calves:**

1. Costs of a 'hygiene barrier', shown in Cell G270,
2. Costs of 'better roughage, straw and others', shown in Cell G271-H273 for respectively young (0-6 months) and old (7-12 months) calves,
3. Costs of 'new housing' to separate calves and adult cows more effectively. These costs are only included if Cell C195-D195 are 'True' and Cells G274-H274 are also 'True'. In addition, the costs of 'new housing' are shown in G275-N276, separately for young and old calves and big and small herds.

• **Costs of labor:**

In several places in the model, extra labor can be associated with certain changes in management. The costs of labor per hour are shown in Cell G246.

• **Costs of vaccination:**

Those costs are already explained at 'Vaccination' and they will be added to the costs of control for each calf that is vaccinated.

• **Costs of having two groups of cows:**

The costs associated with having two groups of cows are (1) extra labor and (2) costs per cow because of e.g. 'extra housing'.

1. In Cell G280-J281 the number of extra labor hours (for big and small herds) are shown,
2. In Cell G282 the costs per cow are shown.

The costs of 'changing the management' can be included for $x\%$ ($0\% < x < 100\%$) to reflect the situation that not all costs of changing the management are associated with Johne's disease, but also will give economic benefits on other factors. E.g. if we build a new barn to separate calves better from older cows, a reduction of the losses of other disease can also be expected. The input for this percentage is in Cell H1, sheet 'InputSector' of the 'Assimilation file'.

5.6. Assimilation of results

After simulating different herd types and/or farms with different infection statuses, we can read to the separate results or the assimilated results after running the 'Assimilation model'. For this we need to take the next steps:

1. First we need to have the .rst-files (both no control and control) in the same folder and if we press the 'open'-button, this folder should be opened first.
2. Secondly, we have to open the 'Assimilation file', on the 'InputSector' sheet.
3. Then we put the names of the .rst-file in Cell C6 and under (with control) and in Cell D6 and under (without control). In this way, the model calculates the benefits of control (reduction of losses). If we only put the .rst-files without control in the C-column, we get the losses without control (there are no benefits in this situation).
4. In addition, we can change Cell H1 to include only part of the management costs or we can change Cell H2 to include ('True') or exclude ('False') the costs and benefits of 'heifer contract rearing'.

To save the results per strategy, we save each strategy in a 'Assimilation' file, change the input and run another farm type or strategy (advice: use find and replace, 'Ctrl H').

In addition, we can select specific Iterations (if Cell H6 = TRUE) on either the ParaStatus or the ParaStatusAccent (determined by input in Cell H7). We can also select on ParaStatusUSA, but then we have to change Cell H8 to TRUE.

Which Status we select on, is determined by the input in the Cells H11 – I11 and the Assimilation_Code will search for iterations with a Status between those two inputs between the years, shown in the Cells H10 – I10.

5.7. Assimilation of more files at once

In the same way as is explained in paragraph 4.3, we can assimilate many files at once (once pressing the button "Start"). This is described in more detail in paragraph 4.3.

6. VBA Code

6.1. Simulation Code

The main explanation of the Simulation Code is found **in** the Simulation Code itself. This allows easier and better access to the explanations. In addition, a flow diagram of the code is given in Appendix I, to show the overall-structure of the model/simulation.

6.2. Assimilation Code

Again, the main explanation of the Assimilation Code can be found **in** the Assimilation Code itself. In addition, also a flow diagram is given in Appendix II, to show the overall-structure of the code.

Appendix I. Flow-chart of the Visual Basic for Applications **Simulation** (VBA) Code

1
2
3 BatchRun (start of simulation of more profile*status files)
4 For all files do:
5 Main.Main (simulation of one profile*status file)
6 InitializeGeneral (give names and input-data to variables/parameters in the model)
7 InitializeInputVar (most of the input variables)
8 InitializeInputControl (input related to "control")
9 InitializeTest (read data of all tests)
10 InitializeSettings (general parameters and opens stv file)
11 InitializePrevalenceCriteria (reads the criteria)
12 OpenResultsFile (opens.rst file)
13 CreateInitialHerd (Herd on a sheet)
14 CreateNewInitialHerd (Make this herd)
15 InitAnimalSeeds
16 GetInitialHerdFromSheet (Read this herd)
17 InitAnimalSeeds
18 If SaveHASC (If making a .stv file:)
19 InitializeIteration (Start one iteration)
20 For Tstep = 1 to TotNrOfTSteps (Do all Tsteps)
21 FollowHerdOneTimeStep (to make a .stv herd) (Follow the herd one
22 Tstep)
23 SetResultsToZero (to make a .stv herd) (Make all results first
24 zero)
25 MakeLossesZeroAgain (Make losses zero too)
26 UpdateControlAndDoTesting (Check if time for any control?)
27 BackgroundTest (What is test-prevalence?)
28 UpdateActiveControl
29 StartControl (Time to Start control)
30 CheckToStartMngtChangesNow (change)
31 CountYearlyGDCosts (costs Animal Health
32 Serv)
33 CountYearlyManagementCosts (costs Mngt)
34 CountCostsTwoGroups (costs of two groups)
35 DetermineTestAnimals (which animal test if only part)
36 MakeGroupsByAge (grouping based on age)
37 CheckVaccination (Vaccination yes/no)
38 CarryOutTest (depending on test characteristics)
39 CountVetTestCost (costs Veterinarian)
40 TestOneAnimal (test animals)
41 CountVetTestCost
42 CountJustTestCost
43 TestEffectAnimal (effect of test)
44 TestEffectCalves (cull calf of positive)
45 CountTestedAnimals
46 CountNumberOfAnimalsPositive
47 PutAnimalInGroup
48 CarryOutPooledTest (Do pooled test)
49 CullingAndReplacement (All replacement in the herd)
50 InvCulling (Involuntary culling)
51 CountCulling
52 CullCalvesOfClinicalCow
53 CountVetAndTreatmentLosses

54 CountIdleProdFactorsLosses
55 CullingAfterTest
56 CountReducedSlaughterValueLosses
57 CountFutureIncomeLosses
58 VolCullingDoAll (Voluntary culling)
59 VolCullingSuckling (Suckling herd)
60 CullVolSucklingNow
61 BuyHeifersNow
62 VolCullingDairy (Dairy herd- RPO values)
63 HeiferBuyOrSell (herd herd-size purpose)
64 IntroductionInfectedAnimal (Infection from outside?)
65 MakeInfectedAnimal (If yes, infect animal)
66 CountInfection
67 CountAnimalsPurchased
68 TestOneAnimal (test introduced
69 animal)
70 CountInfectiousAnimalsDuringLoopNew(count infectious
71 animals)
72 CountInfectiousAnimalsDuringNew
73 DetermineCowsWithInfectedAntibioticMilk
74 CountCowsWithInfectedAntibioticMilk
75 CalculateInfectionProbabilities (calculate infection
76 probabilities)
77 For Animal = 1 to AllAnimals
78 DstatChange (Disease status changes)
79 InfectionMilkNew (infection via milk?)
80 CountInfection
81 InfectionRF (infection environment – Reed
82 Frost)
83 CountInfection
84 CowsWithInfectiousColostrum (colostrum)
85 CountCowsWithInfectiousColostrum
86 MilkProd (Do everything with milk)
87 CountProduction
88 CountManagementCostRaisingCalf (costs Mngt.)
89 Next Animal = 1
90 For Animal = 1 to AllAnimals
91 CreateCalf (all calvings)
92 Next Animal = 1
93 CountManagementCostRaisingCalf (costs Mngt.)
94 InfectionsMixedColostrumDoAll (infections mixed colostrum)
95 CalculateInfProbabilityMixedColostrum
96 InfectionsMixedColostrumLoop
97 InfMixedColostrum
98 CountInfection
99 InfectionsSucklingMoreCowsDoAll (infections suckling calves)
100 CalculateInfProbabilitySucklingMoreCows
101 InfectionsSucklingMoreCowsLoop
102 InfectionsSucklingMoreCows
103 CountInfection
104 CountAnimalLoop
105 CountAnimal (count animals per age/disease group)
106 CheckParaStatus (Dutch and USA status)

```

107             SaveResultsAll (Save all results)
108                 CalculateResults (calculate some end-results)
109                 TotalLossesAndCosts (sum up some results)
110                 CalculateSensitivityTests (estimate the overall
111                 sensitivity)
112                 WriteResultsToFile (write to .stv-file or .rst file)
113             Next Tstep
114             WriteAnimalData
115             For Tstep = 1 to TotNrOfTSteps
116                 FollowHerdOneTimeStep (to make a .rst herd)
117                 * Start at FollowHerdOneTimeStep (see above, same steps)
118             Next Tstep
119             WriteAnimalData
120         Else (so,,if ReadHASC)
121             InitializerIteration
122             For Tstep = 1 to TotNrOfTSteps
123                 FollowHerdOneTimeStep (to make a .rst herd)
124                 * Start at FollowHerdOneTimeStep (see above, same steps)
125             Next Tstep
126             WriteAnimalData
127         End if
128         CloseResultsFile (close the profile * status input file)
129     End of Main module
130     Next file (go to next profile*status input file)
131 End of BatchRunModule (Simulations are ready)

```


Appendix II. Flow-chart of the Visual Basic for Applications **Assimilation** (VBA) Code

132
133
134
135 GenerateResultsSector
136 DetermineNumberOfIterations
137 OpenResultsFile
138 SetResultsSetToZero
139 ObtainYearResultsSingleSet
140 DetermineNumberOfIterations
141 OpenResultsFile
142 SetResultsSetToZero
143 ReadBasicResultsIntoResYear
144 OpenResultsFile
145 CalculationsSingleSet
146 CalculateSensitivity
147 CalculateResYearPerc
148 CalculateReductionLosses
149 ObtainYearResultsSingleSet (Stv file WITHOUT control!)
150 DetermineNumberOfIterations
151 OpenResultsFile
152 SetResultsSetToZero
153 ReadBasicResultsIntoResYear
154 OpenResultsFile
155 DiscountEconResults
156 SumOverYears
157 AddAllIterations (Add iterations to Sector)
158 AddSelectedIterations (Select part of iterations to Sector)
159 CalculateResYearPerc
160 CalculateSectorEconomics
161 GenerateStatistics
162 IndexSort
163 EconomicParameters
164 IndexSort (twice)
165 MakeTable (several times)
166 MakeSmallTable (several times)
167 MakeDiscountedTotalsTable
168 MakeSmallTableAdditionData (several times)
169 MakeParaStatusTable (several times)
170 MakeTableSensitivity
171 End Sub
172
173
174