

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



# Automating First- and Second-order Monte Carlo Simulations for Markov Models in TreeAge Pro

Benjamin P. Geisler, M.D., M.P.H.  
*United States of America*

## 1. Introduction

### 1.1 Note

This chapter draws heavily from a journal article (Geisler et al., 2009). The method described in there formed the basis for this book chapter and has been extended since.

### 1.2 Markov models

Markov models enable stochastic evaluation of linear and also non-linear decision problems that might otherwise be difficult to evaluate.

Named after the Russian mathematician Andrey Andreyevich Markov (1856-1922), Markov models enjoy a widespread appreciation in engineering, medicine, and other applied scientific disciplines.

For example, one application of this modelling approach lies in health technology assessment which aims to “forecast” clinical and economic consequences of adopting new diagnostic or treatment strategies. In this area, Markov models are typically used to estimate the long-term cost-effectiveness analysis of new medical interventions, i.e., weighing the incremental costs of a given intervention with its potential incremental health outcomes, and comparing their ratio to other incremental cost-effectiveness ratios (Weinstein & Stason, 1977).

However, Markov models are also useful to

- a. combine input parameters from different sources: short-term experimental studies (e.g., randomized controlled trials) with long-term observational studies, diagnostic with treatment strategies, costs with duration and quality-adjustment of life time, or any combination of these;
- b. extrapolate results to a longer time horizon, if possible ideally to lifetime;
- c. transfer or transpose data to a different patient cohort (e.g., a cohort with different baseline characteristics, a different compliance or adherence, or an important subgroup), to a different epidemiological situation (i.e., apply the model to other populations with other incidences and/or prevalences), to a different health care provider (who might follow a different standard of care), to a different payor (who might have different formula), or even to a different country or region.

Within a Markov model, often dubbed “state transition model”, the probability to be in one of the so-called Markov states always add up to one. The probability of changing states – or remaining in the same state, for that matter – depends in the classical Markov model solely

on the present state (Markovian property). This strict property can, however, be relaxed. Information *can* be maintained by “tracking” events as values in variables (tracking variables). Therefore, discrete events can be used to calculate state transition probabilities or rewards. The latter are used, for example, to calculate costs of health outcomes.

### 1.3 Monte Carlo simulations

As explained in more detail in other chapters of this book, there are two types of Monte Carlo simulations: first-order Monte Carlo simulations (sometimes referred to as “trials” or “microsimulations”) and second-order Monte Carlo simulations. These simulations both represent uncertainty that is not addressed in deterministic analysis (i.e., deterministic analysis does not use probabilities for parameters, just point estimates). However, first- and second-order Monte Carlo simulations represent different types of uncertainties that are consequently used for different purposes.

Second-order Monte Carlo simulation addresses parameter-uncertainty. A typical application in health care modelling is probabilistic sensitivity analysis of cost-effectiveness models. Probabilistic sensitivity analysis assesses the “joint effect of parameter uncertainty” (Briggs et al., 2002) on the result of the model, i.e. when randomly drawing from all distributions simultaneously. In cost-effectiveness analysis, the thus computed incremental costs and health outcomes are most commonly plotted on an X-Y scatter plot. Another very common depiction of the results of a probabilistic sensitivity analysis are cost-effectiveness acceptability curves where the “willingness to pay” (e.g., in dollars per quality-adjusted life year) is plotted against the proportion of runs that resulted in incremental cost-effectiveness ratios below this willingness to pay (Fenwick et al., 2004).

First-order Monte Carlo simulation addresses stochastic uncertainty. Their application is less straightforward.

However, in a more relevant application, first-order Monte Carlo simulations employing “tracker variables” enable one to overcome the Markov property. Another way to do this would be to save information by increasing the number of health states. However, “state explosion” makes model-building more prone to errors. Tracker variables offer an attractive way to incorporate discrete events into Markov models.

Moreover, first-order Monte Carlo simulations might be an interesting way to reflect heterogeneity in patient populations, a desirable goal.

Stochastic uncertainty from a practical point of view is considered “random noise from a decision maker’s point of view” but, as a means to an end, “can be overcome by increasing the sample size of the microsimulation” (Weinstein, 2006) to achieve the above stated goals.

The use of first-order Monte Carlo simulations in decision-analytic modelling has constantly increased over time (Hunink et al., 2001). This is largely due to the fact that drastically increased computer power has made it possible to perform microsimulations in reasonable amounts of time.

However, with a standard modelling package such as TreeAge Pro™, it can be difficult to perform multiple deterministic sensitivity analyses in model requiring first-order Monte Carlo simulations, since each single data point requires another microsimulation to be performed, and thus huge investments of time have to be spared by the researcher. The goal of this chapter is therefore to identify ways to automate deterministic one-way sensitivity analysis of models requiring the use of first-order Monte Carlo simulations.

### 1.4 TreeAge Pro™

Formerly named DATA Pro, TreeAge Pro™ (TreeAge Software, Inc., Waltham, MA, USA) is popular modelling software capable of computing expected values of Markov models as well as deterministic sensitivity, threshold and probabilistic sensitivity analysis via second-order Monte Carlo simulation. TreeAge Pro™ is also able to run microsimulations via first-order Monte Carlo simulation. Advanced value-of-information analysis features in TreeAge include expected value of partially perfect information analysis via two stacked second-order Monte Carlo simulations.

While value-of-information analysis based on microsimulation models can be run “out of the box” in three-dimensional models, it is not possible to run simple deterministic sensitivity analysis on models requiring microsimulations in TreeAge Pro™.

A relatively new analysis type, cost-effectiveness sensitivity curves, requires repeated runs of second-order Monte Carlo simulations (O’Day et al., 2010). This is also not natively supported by TreeAge Pro™.

### 1.5 Chapter objectives

This chapter will explain the techniques to automatically run Monte Carlo simulations, in TreeAge Pro, which potentially saves decision scientists valuable time and energy. The text will provide the open-source codes for Excel®- and Visual Basic®-based interfaces to run these analyses as well as to adapt and develop them further.

## 2. General materials and methods

### 2.1 Object-oriented programming

In general, the goals of Object-Oriented Programming are (1) to re-use programs and sub-programs; and (2) to ease the use by allowing programs to be accessed by other programs. In Object-Oriented Programming, *objects* are defined classes, methods, or other “building blocks”. That means that objects can be initiated and reused by other objects (created, accessed by other programs). Objects have certain behaviours (the things it can do, or features), and objects will have certain characteristics (attributes, fields, parameters, data, or properties). They can exist in a certain hierarchy where a daughter instance “inherits” all objects from the mother instance. This again eases the use of the coder as he/she does only need to define objects once and for all.

### 2.2 TreeAgeProLib

TreeAge Pro™ features graphical user interface, TreeAge Pro™ as we know it. However, starting with version 2007, TreeAge Pro™ also includes an object or “scripting” interface which provides access from other programs.

TreeAge Pro Suite/Excel installs an “add-in” to Excel that adds TreeAge Pro™ objects to be accessed by Excel (see below for preparations in Excel® and Visual Basic®).

“TreeAgeProLib” is an object library that implements a standard COM interface to create and access TreeAge Pro™ objects in any program, script, or macro can create or access TreeAge Pro™ objects through this interface.

An object interface enables automating of tasks that are common, repetitive, and time-consuming, e.g., (re)setting a tree’s variables from outside TreeAge Pro™, automatically running a set of analyses, or automatically exporting results. The object interfaces integrate

TreeAge Pro™ functions into other systems or applications like Excel® via Visual Basic®, into websites (via Microsoft®'s .ASP technology), or into models programmed for examples in C++.

The object browser is included in the Microsoft Visual Basic® for Excel® editor and shows object properties and methods. The objects available can be browsed in Visual Basic® by choosing Object Browser from the “View” menu or by simply pressing <F2>. After choosing the TreeAgeProLib, one can browse as well as search all objects and see descriptions and syntaxes (please see Figure 1).

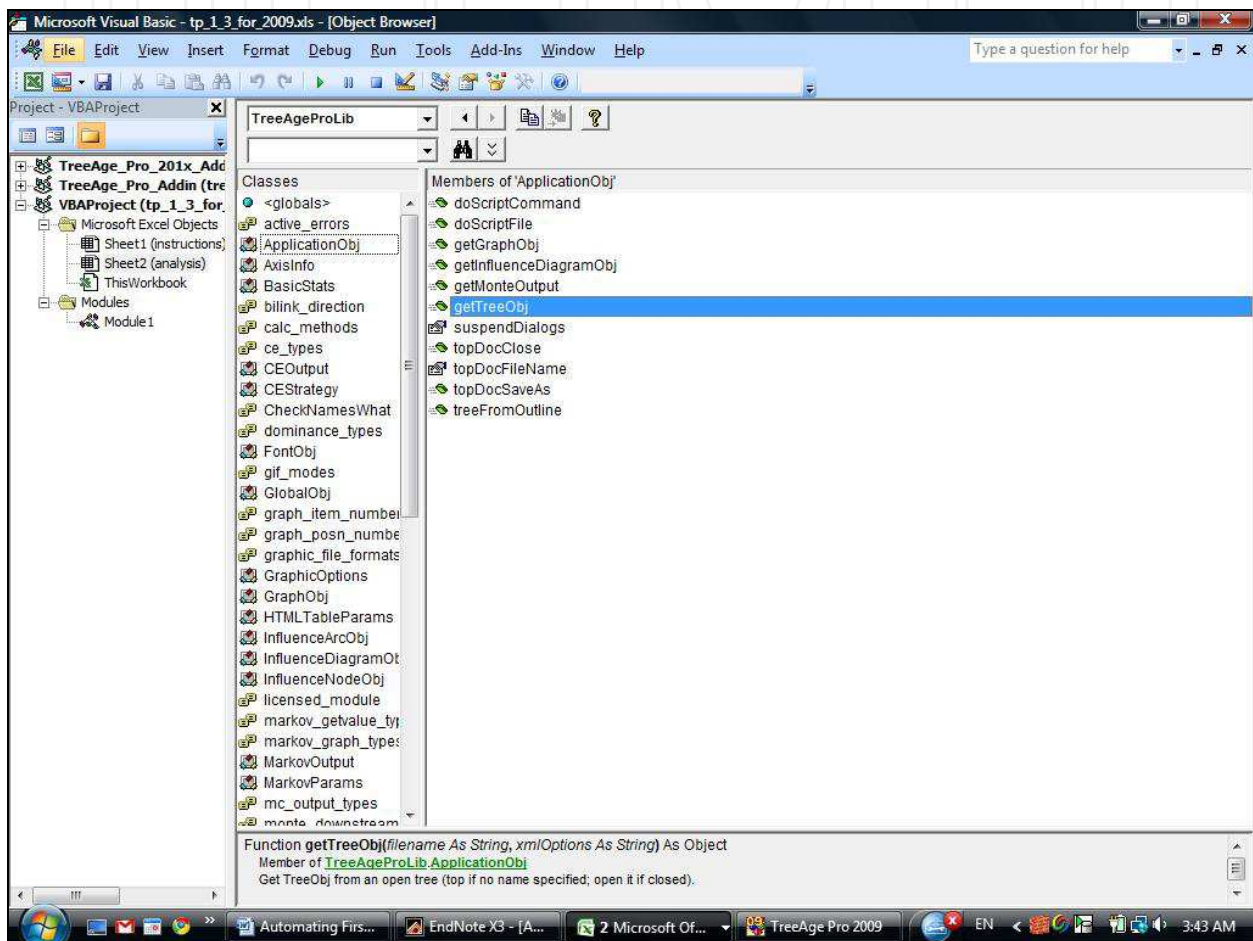


Fig. 1. Browsing objects from TreeAgeProLib object library in Visual Basic®

### 2.3 Software requirements

The object interface requires the Excel® Module (included in Tree Age Pro™ Excel and Tree Age Pro™ Suite). Tree Age Pro™ Suite 2009 includes both the Healthcare and the Excel® packages which are necessary for our purposes. All analyses have been conducted in TreeAge Pro™ Suite 2009 (from now on simply referred to as TreeAge Pro). The model will also open and the Visual Basic®/Excel® interface will function in earlier versions. This has been successfully tested in TreeAge Pro™ 2007 and 2008. However, the interface will not work with adaptations in the soon-to-be-released TreeAge Pro™ 2011. The reason for this is that TreeAge Pro™ 2011 uses a completely different platform, ECLIPSE™ integrated development environment, based on the programming language Java.

## 2.4 Preparations in Excel® and Visual Basic®

Please note that Excel® needs to be able to execute macros. You can tell Excel® whether it shall accept macros at Tools > Options > Security > Macro security. In order to execute our macro, you need to put the macro security level to medium (or low). You will then be prompted about whether you want to allow macros when opening the spreadsheet file.

The Visual Basic® editor can be opened in Excel® by pressing Alt + F11. In Microsoft Office® Excel® versions 2007 and 2010, the editor can be reached via the developer tab in the “ribbon”.

First, the TreeAge Pro Suite add-on in Excel® needs to be enabled. This adds TreeAge objects that can then be accessed by Excel®. Click on Tools > References and make sure that the reference “TreeAge Pro 2009.0.0 Type Library” is enabled (please see Figure 2).

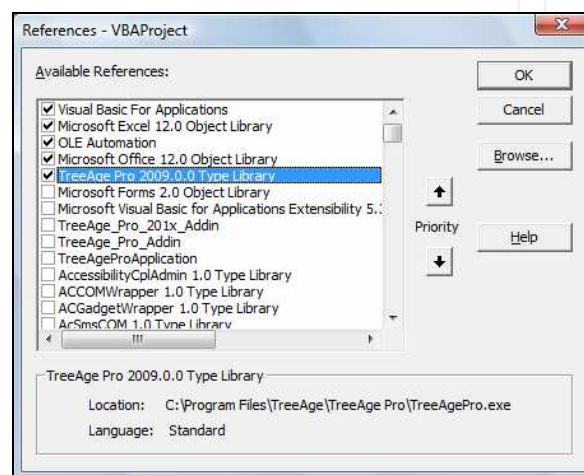


Fig. 2. References in the Visual Basic® Editor

In the tree view on the left, click then on “Module1” (under VBAProject (filename) > Modules) if it is not already opened in the editor.

## 3. Automating deterministic one-way sensitivity analysis in Markov Models requiring first-order Monte Carlo simulations

Next we tackle the generic script that was constructed which enables the performance of automated deterministic one-way sensitivity analyses in Excel employing microsimulation models.

We use object-oriented programming to access the TreeAgeProLib objects of the TreeAge Pro Excel® add-in to perform a specified number of first-order Monte Carlo simulations in Excel®.

The code in Visual Basic® is given below in the next subsection 3.1. In the subsequent section 4 we will then discuss an easy way to access the script.

### 3.1 Visual Basic® code

The goal of the script is to run a file in TreeAge repeatedly while varying one parameter during each run, and subsequently output the results.

More specifically, the coding objectives are to

1. open the tree;
2. loop for the purposes of running model for each value that the sensitivity analysis requires;

3. repeatedly manipulate variable in root;
4. repeatedly run the Monte Carlo simulation; and
5. output effectiveness, costs, incremental cost-effectiveness ratio in Excel®.

We will use the following TreeAgeProLib object:

- *ApplicationObj*, used to open the tree
- *TreeObj*, to access the tree
- *NodeObj*, to navigate to the route and change the parameter of interest
- *MonteParams*, to set up the Monte Carlo simulation
- *MonteOutput*, to access the results of the Monte Carlo simulation
- *BasicStats*, to access the costs and effectiveness results

Given below is a sample code that needs to be put into the Visual Basic® for Excel® editor.

Please note that every line that starts with an apostrophe (') and in the colour green is just a comment line and not necessary for the program.

Please also note that terms in italics need to be replaced with either the name of the appropriate worksheet or the appropriate cell number for the input parameters.

```
Sub one_way_w_two_strategies()
```

```
'the locations and name of our model; "" means that the currently opened tree is being used
```

```
Const pkgname = ""
```

```
'number of microsimulations
```

```
Dim runs As Double
```

```
'start row and current row
```

```
Dim srow As Integer
```

```
Dim currow As Double
```

```
'variable of interest
```

```
Dim var As String
```

```
'startvalue, number and value of interval (between the values)
```

```
Dim nInter As Integer
```

```
Dim vInter As Double
```

```
Dim sval As Double
```

```
'seeding
```

```
Dim seedN As Long
```

```
Dim seedB As Integer
```

```
'declaration of variables for use within the loop
```

```
Dim i As Integer
```

```
Dim cur As Double
```

```
Dim cur_string As String
```

```
Dim root As NodeObj
```

```
Dim mcParams As New MonteParams
```

```
Dim mcOutput As MonteOutput
```

```
Dim cstats_nt As BasicStats
```

```
Dim cstats_t As BasicStats
```

```
Dim estats_nt As BasicStats
```

```
Dim estats_t As BasicStats
```

```
Dim err2 As Long
```

```

Dim icer As Double
Dim c_nt As Double
Dim c_t As Double
Dim e_nt As Double
Dim e_t As Double
Dim c_i As Double
Dim e_i As Double
Dim curcolumn As Integer
Const save = 0
'first a message
    Application.ScreenUpdating = False
    ' turns off screen updating
    Application.DisplayStatusBar = True
    ' makes sure that the statusbar is visible
    Application.StatusBar = "Please wait while the tree is being opened..."
'definition of number of microsimulations
ActiveWorkbook.Sheets("name of worksheet ").Select
Range("appropriate cell for number of runs").Select
runs = ActiveCell.Value
'definition of start row
srow = 20
currow = srow
'definition of variable of interest
ActiveWorkbook.Sheets("name of worksheet ").Select
Range("name of worksheet for variable of interest").Select
var = ActiveCell.Value
'seeding
Range("name of worksheet for seed number").Select
seedN = ActiveCell.Value
Range("name of worksheet for seed behavior").Select
seedB = ActiveCell.Value
'the following code lines serve the purpose to read the start value, number of intervals and
value of intervals from the spreadsheet; if not the spreadsheet from the website is being
used, the cell values might have to be adjusted
Range("name of worksheet for starting value").Select
sval = ActiveCell.Value
Range("name of worksheet for number of intervals").Select
nInter = ActiveCell.Value
Range("name of worksheet for the value of each interval").Select
vInter = ActiveCell.Value
'start of the loop
For i = 1 To nInter
    'definition of the current value of the variable of interest; determined by the loop
    number (i)
    cur = sval + ((i - 1) * vInter)
    cur_string = Str$(cur)

```



```

'initiation of application and tree
Dim appObj As TreeAgeProLib.ApplicationObj
Dim tree As TreeAgeProLib.TreeObj
Set appObj = New TreeAgeProLib.ApplicationObj
Set tree = appObj.getTreeObj(pkgname, "")
Application.Wait Now + TimeValue("00:00:01")
Application.StatusBar = "The tree is opened. Now the MC simulation will be run..."
'manipulation the root and setting the variable of interest to the current value
Dim ok As Long
ok = tree.setVariableValue(defined_at_root, var, cur_string)
'monte carlo microsimulation/trials (1st order monte carlo simulation) and threads
mcParams.trials = runs
mcParams.sampleType = 0
mcParams.seedValue = seedN
mcParams.seedBehavior = seedB
Application.StatusBar = False
Set mcOutput = tree.MonteCarlo(mcParams)
While mcOutput.TimeElapsed = 0
    Application.Wait Now + TimeValue("00:00:02")
Wend
'check
If mcOutput.valid = 0 Then
    MsgBox ("McOutput invalid")
    Exit Sub
End If
' costs and efficiency by strategy
Set cstats_nt = mcOutput.GetStatsByStrategy(ce_cost, 1, 0.1)
Set estats_nt = mcOutput.GetStatsByStrategy(ce_eff, 1, 0.1)
Set cstats_t = mcOutput.GetStatsByStrategy(ce_cost, 2, 0.1)
Set estats_t = mcOutput.GetStatsByStrategy(ce_eff, 2, 0.1)
'calculate icer and output
c_nt = cstats_nt.mean
c_t = cstats_t.mean
e_nt = estats_nt.mean
e_t = estats_t.mean
c_i = c_t - c_nt
e_i = e_t - e_nt
If e_i <> 0 Then icer = c_i / e_i Else icer = 0
If icer < 0 Then icer = 0
'definition of the current row
currow = srow + (i - 1)
'output
curcolumn = 1
ActiveWorkbook.Sheets("name of worksheet").Cells(currow, curcolumn).Value = cur
curcolumn = curcolumn + 1
ActiveWorkbook.Sheets("name of worksheet").Cells(currow, curcolumn).Value = icer

```

```

curcolumn = curcolumn + 1
ActiveWorkbook.Sheets("name of worksheet ").Cells(currow, curcolumn).Value = c_i
curcolumn = curcolumn + 1
ActiveWorkbook.Sheets("name of worksheet ").Cells(currow, curcolumn).Value = e_i
curcolumn = curcolumn + 1
ActiveWorkbook.Sheets("name of worksheet ").Cells(currow, curcolumn).Value = c_nt
curcolumn = curcolumn + 1
ActiveWorkbook.Sheets("name of worksheet ").Cells(currow, curcolumn).Value = c_t
curcolumn = curcolumn + 1
ActiveWorkbook.Sheets("name of worksheet ").Cells(currow, curcolumn).Value = e_nt
curcolumn = curcolumn + 1
ActiveWorkbook.Sheets("name of worksheet").Cells(currow, curcolumn).Value = e_t
    'reset of all variables
    Set mcOutput = Nothing
    Set tkt = Nothing
    Set tree = Nothing
    Set root = Nothing
    Set mcParams = Nothing
    Set mcOutput = Nothing
    Set cstats_nt = Nothing
    Set cstats_t = Nothing
    Set estats_nt = Nothing
    Set estats_t = Nothing
    err2 = 0
    'closure of the Monte Carlo simulation window
    appObj.topDocClose (0)
Next
End Sub

```

---

### 3.2 Excel® spreadsheet

In addition to the script, a spreadsheet was constructed in Excel® (please see Figure 3). The workbook allows one to make use of the script without programming knowledge.

The script and spreadsheet (as well as the script that comes with it) is generic for all TreeAge decision trees with two strategies and Markov nodes

The spreadsheet, which includes the script as a macro, can be downloaded from the author's website [www.hcval.com](http://www.hcval.com).

## 4. Automating cost-effectiveness sensitivity curves in Markov Models requiring second-order Monte Carlo simulations

Analogous to the previous section, let us explore if there are applications that make it worthwhile to also automate second-order Monte Carlo simulations and how they can be realized.

One possible application is expected value of partially perfect information analysis, which requires either two stacks of second-order Monte Carlo simulations, or even, if first-order Monte Carlo simulation is required to derived expected values, three levels of analysis.

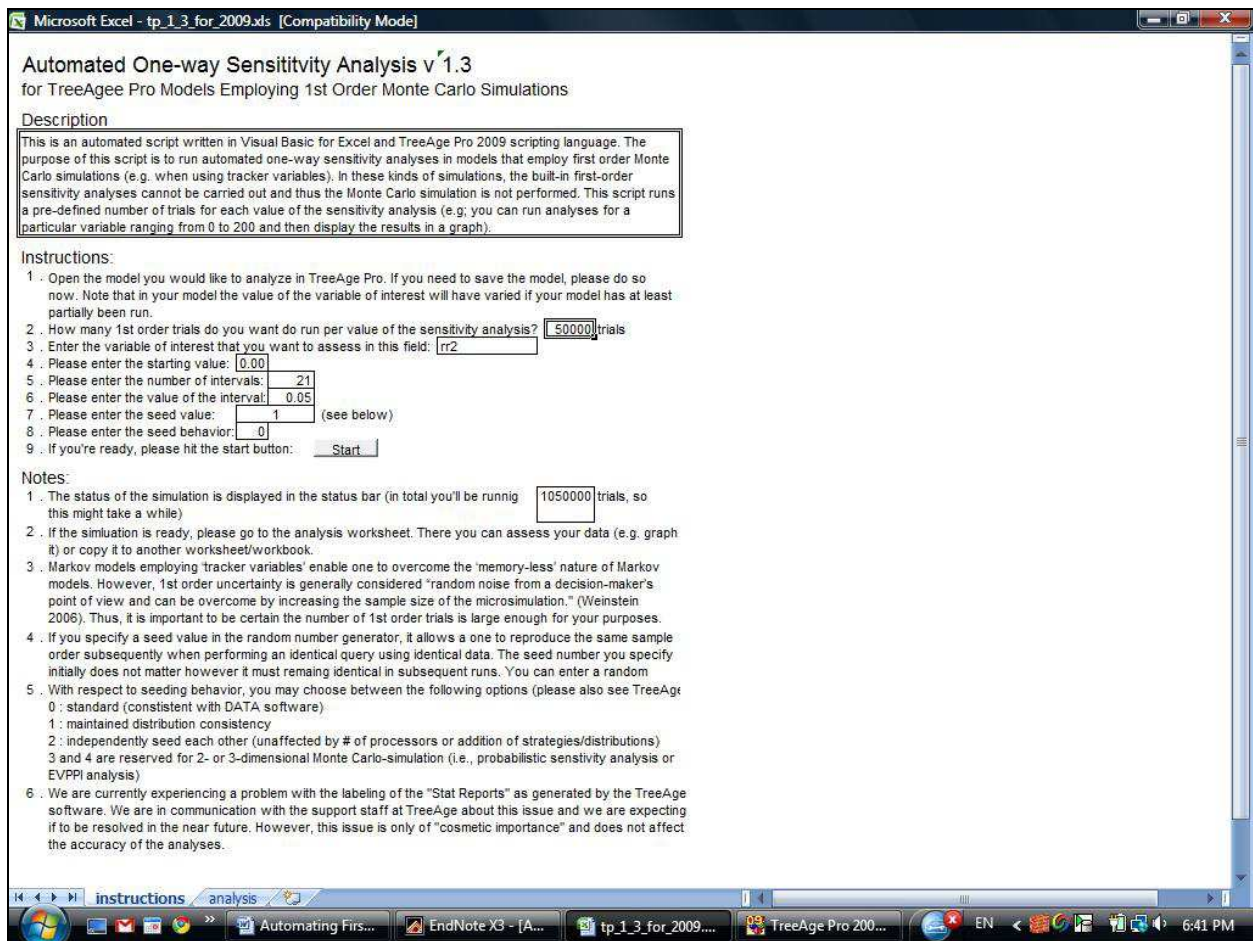


Fig. 3. Generic Excel® spreadsheet

However, both are natively supported in TreeAge Pro and these features generally perform well.

Another possible application are cost-effectiveness sensitivity curves (O'Day et al., 2010). Cost-effectiveness sensitivity curves are a combination of one-way sensitivity analysis and probabilistic sensitivity analysis. For each curve, one willingness-to-pay level needs to be specified, let us say by convention \$50,000/QALY and \$100,000/QALY. One parameter of interest is varied and, for example, plotted on the x-axis. For each value that this parameter can take in the range specified on the x-axis, a separate second-order Monte Carlo simulation has to be run. In this type of analysis, we plot the proportion or percentage of runs that were at or below the specified willingness to pay threshold on the y-axis.

It would be desirable to run these kinds of analyses either in normal deterministic Markov models or in those requiring First-order Monte Carlo simulations (because tracker variables are used). Unfortunately, the methods to extract the necessary values from second-order Monte Carlo simulations are currently not specified for the object in question, MonteOutput. Hopefully, in the future this will be supported by TreeAge Pro.

## 5. Example

Consider a simple example: a Markov model with two states, alive and dead, that nevertheless represents a chronic diseases that has a higher probability of recurrence and a worse quality of life (morbidity) as well as a higher risk of death (mortality) with each

recurrence. Additionally, the treatment gets more and more expensive as the disease progresses.

Please see Figures 4 through 6 for representations of the model.

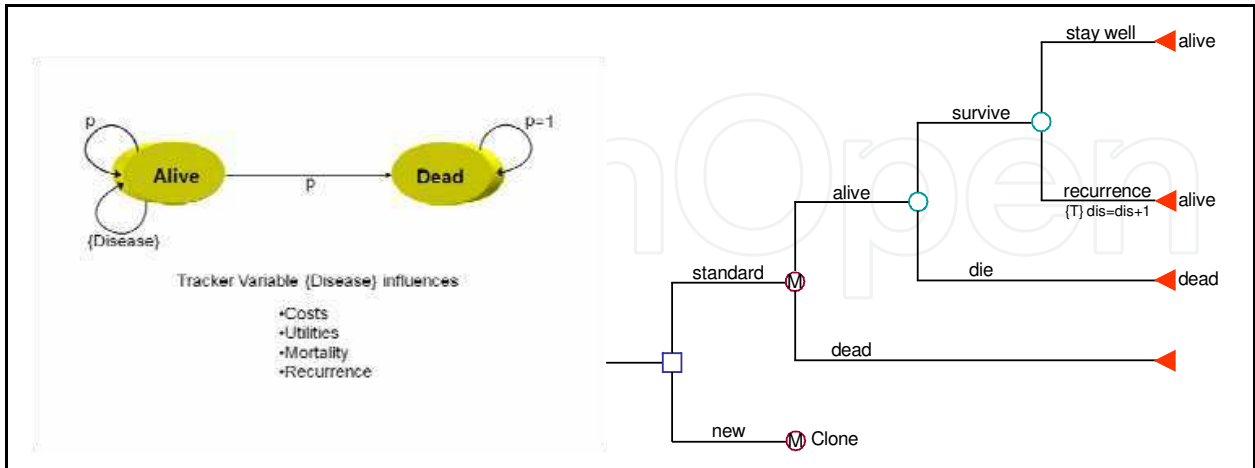


Fig. 4. Illustrative Markov Model requiring a tracker variable (bubble diagram and decision tree with Markov states)

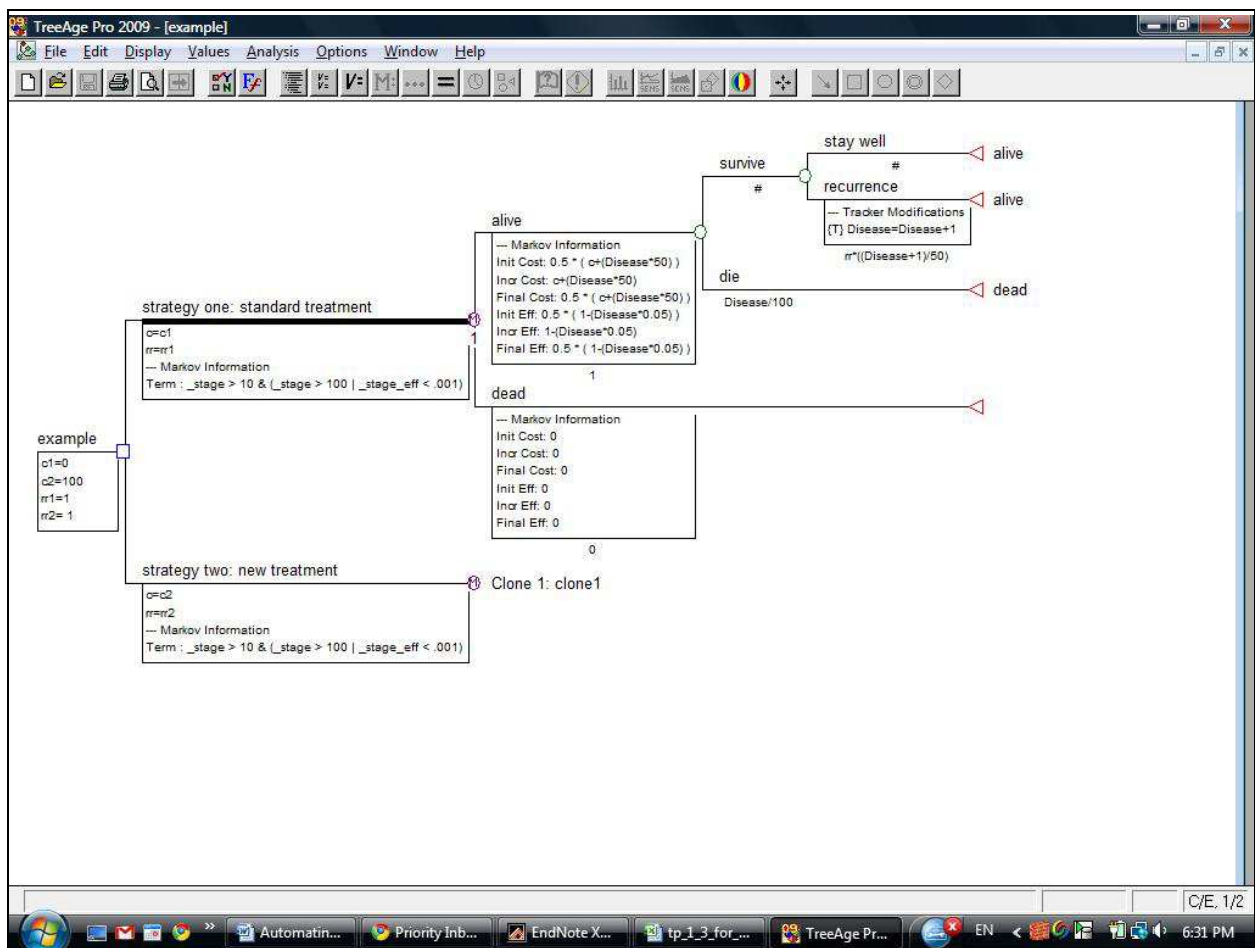


Fig. 5. Illustrative Markov Model requiring a tracker variable (screenshot)

Although this is such a minimalistic model, it is possible to add complexity by other means: e.g., a third state, recurrence, could be modelled within the alive state when a tracker variable, let us call it {Disease}, assumes a certain value. Every time a patient gets a recurrence, the value of the tracker variable goes up by one. The tracker variable then in turn influences the transition probability of the recurrence, the transition probability to death (mortality), costs (severer disease might be more expensive), and utility (multiple recurrences might be associated with lower health-related quality of life). Please see Table 1 for the actual variable definitions in this example.

Parameter	Strategy 1 - standard -	Strategy 2 - new -
Costs	Disease*50	100+Disease*50
Utilities	1-(Disease*0.05)	
Recurrence	$\frac{\text{Disease}+1}{25}$	$\frac{\text{RR}*(\text{Disease}+1)}{25}$
Mortality	$\frac{\text{Disease}}{100}$	

Table 1. Input parameters for the example

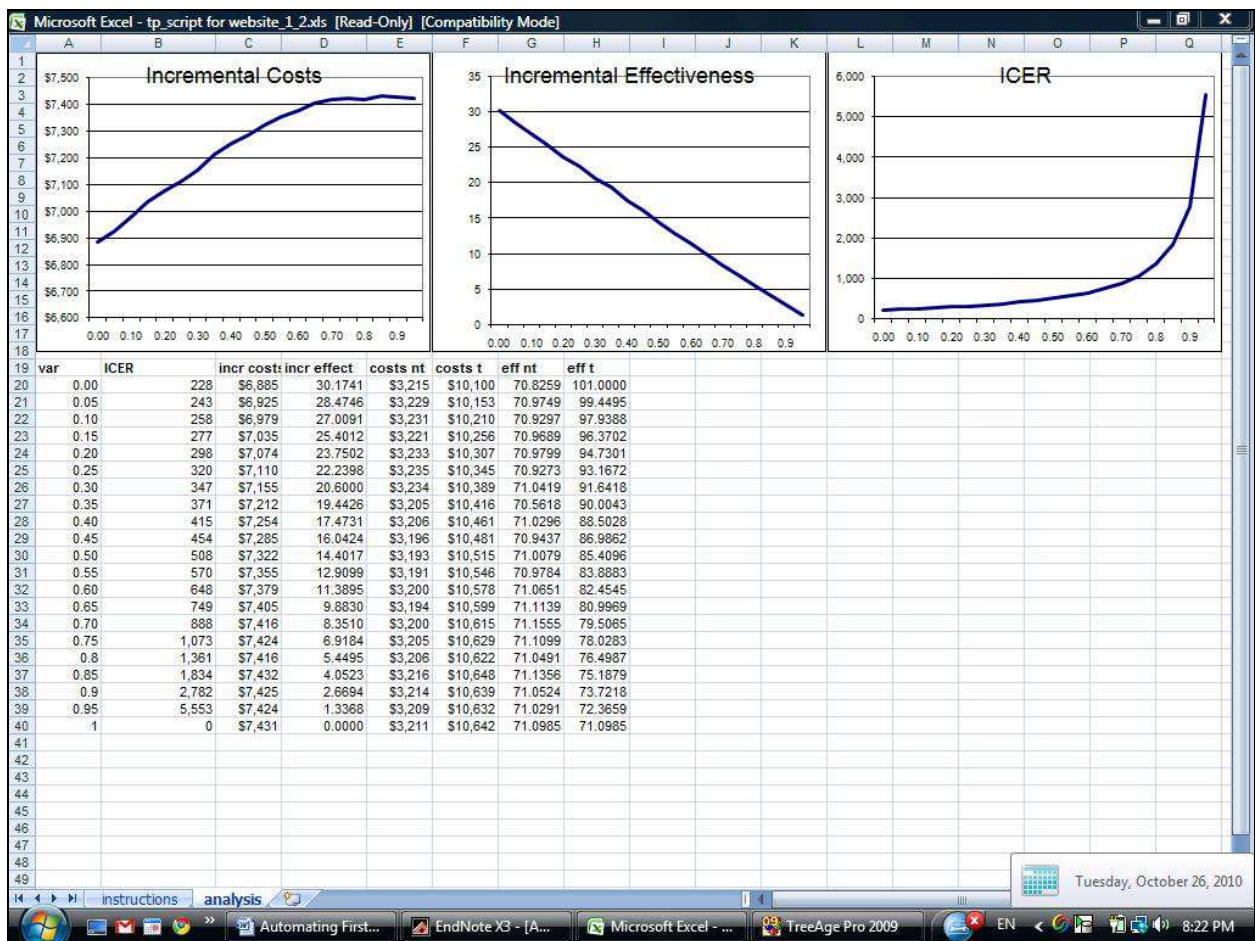


Fig. 6. Resulting deterministic sensitivity analysis from the example in Excel®

In this example, we chose to perform a deterministic one-way sensitivity analysis on  $rr_2$ , i.e., we vary the relative risk that describes how a new treatment (strategy 2) influences the risk of recurrence. Since we are using a tracker variable, we need to perform first-order Monte Carlo simulations *for the entire range of values* that we chose to analyze.

A screenshot of the results in Excel® is provided in Figure 6. As can be seen, the lines are not smooth or (in case of a linear function) straight. This is the result of the model relying on first-order Monte Carlo simulations. However, with increasing sample size, the smoothness of the lines should approach the level of deterministic analysis.

## 6. Other applications

A relatively new analysis type, cost-effectiveness sensitivity curves, requires repeated runs of second-order Monte Carlo simulations (O'Day et al., 2010). This is neither natively supported by TreeAge Pro, nor – as explained above – can this analysis type be run via an automated script from Excel®/Visual Basic®.

Other, potentially interesting applications are automated sensitivity analysis of cohort models, automated deterministic two- or multi-way-sensitivity analyses of models depending on first-order Monte Carlo simulations, automated model calibration, and automated second-order Monte Carlo simulations such as expected value of partial information analyses with varying sets of constant variables.

The online documentation is available at:

<http://server.treeage.com/ObjDocs/TP/TOC/ref.php3> and includes descriptions of all objects and their properties and methods as well as code samples.

Please check the website [www.hcval.com](http://www.hcval.com) for updates.

## 7. Conclusions

Linking TreeAge Pro and Microsoft Excel® via Microsoft Visual Basic® allows the automatic performance of multiple deterministic sensitivity analysis in first-order-Monte Carlo simulations.

The automatic deterministic sensitivity analysis in models requiring first-order-Monte Carlo simulations can be run conveniently via Excel® spreadsheets, with no programming knowledge necessary. A sample worksheet in Excel® for all TreeAge decision trees with two strategies and Markov nodes allows for use of the script without programming knowledge and can be downloaded from the website [www.hcval.com](http://www.hcval.com).

However, it is currently impossible to automate second-order Monte Carlo simulations in any meaningful ways. An example where this would be helpful is the automatic calculation of cost-effectiveness sensitivity curves based on multiple probabilistic sensitivity analyses based on second-order Monte Carlo simulations (O'Day et al., 2010).

## 8. Acknowledgements

The origin of this work was in a decision-analytic model that the author created together with Alexander Göhler, MD PhD MSc MPH, Uwe Siebert, MD MPH MSc ScD, G. Scott Gazelle, MD MPH PhD, and David J. Cohen, MD MSc. The author is indebted to these individuals for accepting him as a research fellow to perform this research, and for co-authoring a journal article on the method that this decision model used and that formed the basis for this book chapter (Geisler et al., 2009). The book chapter draws heavily on the work accomplished in those previous projects.

The author would also like to thank TreeAge Software, Inc. Support for help with new features which were not yet documented. The author is not affiliated with TreeAge Software, Inc.

Software and trademarks are the property of the respective copyright holders. In particular, "TreeAge Pro" and "DATA" and the TreeAge Software logo are TreeAge Software, Inc. trademarks. ECLIPSE is a trademark of Eclipse Foundation, Inc. Java is a registered trademark of Oracle and/or its affiliates. Microsoft® is a registered trademark of Microsoft Corporation in the United States and other countries. Microsoft Office® is a registered trademark of Microsoft Corporation in the United States and other countries. Microsoft Excel® is a registered trademark of Microsoft Corporation in the United States and other countries. These trademarks and trade names are the property of their respective owners. Other trademarks or trade names may be the property of their respective owners.

## 9. References

- Briggs, A.H., Goeree, R., Blackhouse, G. & O'Brien, B.J. (2002). Probabilistic analysis of cost-effectiveness models: choosing between treatment strategies for gastroesophageal reflux disease. *Med Decis Making*, 22, 4, (Jul-Aug 2002) 290-308, ISSN 0272-989X.
- Fenwick, E., O'Brien, B.J. & Briggs, A. (2004). Cost-effectiveness acceptability curves--facts, fallacies and frequently asked questions. *Health Econ*, 13, 5, (May 2004) 405-415, ISSN 1057-9230.
- Geisler, B.P., Siebert, U., Gazelle, G.S., Cohen, D.J. & Gohler, A. (2009). Deterministic sensitivity analysis for first-order Monte Carlo simulations: a technical note. *Value Health*, 12, 1, (Jan 2009) 96-97, ISSN 1098-3015.
- Hunink, M.G.M., Glasziou, P.P., Siegel, J.E., Weeks, J.C., Pliskin, J.S., Elstein, A.S. & Weinstein, M.C. (2001). *Decision making in health and medicine : integrating evidence and values*. Cambridge ; New York, Cambridge University Press.
- O'Day, K., Meissner, B. & Bramley, T. (2010). The Cost-effectiveness Sensitivity Curve: Quantifying the Effect of Individual Parameter Uncertainty in a Probabilistic Model. ISPOR Fifteenth Annual International Meeting. International Society for Pharmacoeconomics and Outcomes Research. Atlanta, GA, Value in Health. 13; 3: A1-219.
- Weinstein, M.C. (2006). Recent developments in decision-analytic modelling for economic evaluation. *Pharmacoeconomics*, 24, 11, 2006) 1043-1053, ISSN 1170-7690.
- Weinstein, M.C. & Stason, W.B. (1977). Foundations of cost-effectiveness analysis for health and medical practices. *N Engl J Med*, 296, 13, (Mar 31 1977) 716-721, ISSN 0028-4793.



## **Applications of Monte Carlo Method in Science and Engineering**

Edited by Prof. Shaul Mordechai

ISBN 978-953-307-691-1

Hard cover, 950 pages

**Publisher** InTech

**Published online** 28, February, 2011

**Published in print edition** February, 2011

In this book, Applications of Monte Carlo Method in Science and Engineering, we further expose the broad range of applications of Monte Carlo simulation in the fields of Quantum Physics, Statistical Physics, Reliability, Medical Physics, Polycrystalline Materials, Ising Model, Chemistry, Agriculture, Food Processing, X-ray Imaging, Electron Dynamics in Doped Semiconductors, Metallurgy, Remote Sensing and much more diverse topics. The book chapters included in this volume clearly reflect the current scientific importance of Monte Carlo techniques in various fields of research.

### **How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Benjamin P. Geisler (2011). Automating First- and Second-order Monte Carlo Simulations for Markov Models in TreeAge Pro, Applications of Monte Carlo Method in Science and Engineering, Prof. Shaul Mordechai (Ed.), ISBN: 978-953-307-691-1, InTech, Available from: <http://www.intechopen.com/books/applications-of-monte-carlo-method-in-science-and-engineering/automating-first-and-second-order-monte-carlo-simulations-for-markov-models-in-treeage-pro>

**INTECH**  
open science | open minds

### **InTech Europe**

University Campus STeP Ri  
Slavka Krautzeka 83/A  
51000 Rijeka, Croatia  
Phone: +385 (51) 770 447  
Fax: +385 (51) 686 166  
[www.intechopen.com](http://www.intechopen.com)

### **InTech China**

Unit 405, Office Block, Hotel Equatorial Shanghai  
No.65, Yan An Road (West), Shanghai, 200040, China  
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元  
Phone: +86-21-62489820  
Fax: +86-21-62489821



© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen