The Improved SimaPro 8 Calculation Engine

SimaPro 8

Table of Contents

Date: May 01, 2014
Version: 1.0

Author: Vincent Cleij

1 Introduction

In the LCA world, there is a tendency for databases and models to become larger and larger. The ecoinvent life cycle inventory database is a good example of this - the number of processes in the successive versions continues to increase. Important for calculating an LCA model using such libraries is the size of what we call the core library. This is the group of processes that are always loaded into the model, even if you only need one process from the library.

Table 1. Increase of the ecoinvent LCI database size (excluding processes added by PRé Consultants; all values are rounded).

Version	Total	Core size	Remark
1.3	2650	1600	Library Ecoinvent unit processes
2.2	4100	2000	Library Ecoinvent unit processes
3.0	10150	7750	Library Ecoinvent 3 - allocation, default - unit

When using ecoinvent 2, it was very rare for the number of processes in an LCA model to rise above 10,000, but with the introduction of ecoinvent 3, this will become a common situation. In a substantial number of cases, the number of processes in an LCA model using ecoinvent 3 will even be significantly higher than 10,000.

Comparing successive versions of the ecoinvent database also sheds light on another important difference: the average number of links per process also increases. Multiplying this number with the number of processes in the core library gives a good indication of the total number of links in an LCA model (or process network) using any version of the ecoinvent library.

Table 2, Average number of links in the core library processes (all values are rounded).

Version	Core size	Links per process	Total number of links
1.3	1600	9	15000
2.2	2000	10	20000
3.0	7750	31	250000

In Table 2, you can see that the total number of links in the core library increases from 20,000 to 250,000 with the update from ecoinvent version 2 to version 3. That is a 12.5 times increase! This is important for estimating the capacity and performance of the SimaPro calculation engine.

2 Limitations of the SimaPro 7 Calculation Engine

The SimaPro 7 calculation engine used one of two methods when calculating a process network, depending on the desired results.

1. Matrix inverse
1.1. Network function
1.2. Analyze function

2. Solver
2.1. Compare function
2.2. Uncertainty analysis function

The fifth function in SimaPro, the tree function, uses a completely different calculation method, but this will not be addressed in this whitepaper.

In general, a solver is better in terms of memory usage, performance and precision. A solver may still use a lot of memory, but it uses less than matrix inverse. However, if a user needs to know not only the end results of the whole network but also the contribution of each process, as in the case of the network and analyze functions, the solver cannot do these calculations and the matrix inverse method is used instead.

The main limitation of the SimaPro 7 calculation engine is its memory usage when using matrix inverse. The theoretical maximum amount of memory that SimaPro 7, a 32-bit Windows application, can use is 4 GB. In practice, the available amount depends on computer configuration. In Windows XP, for instance, the maximum amount of memory SimaPro can use is only about 1.6 GB, often less. The optimal system for SimaPro is configured with a 64-bit operating system like Windows 7 (64-bit) and 8 GB or more physical memory. With this configuration, SimaPro can use the full 4 GB.

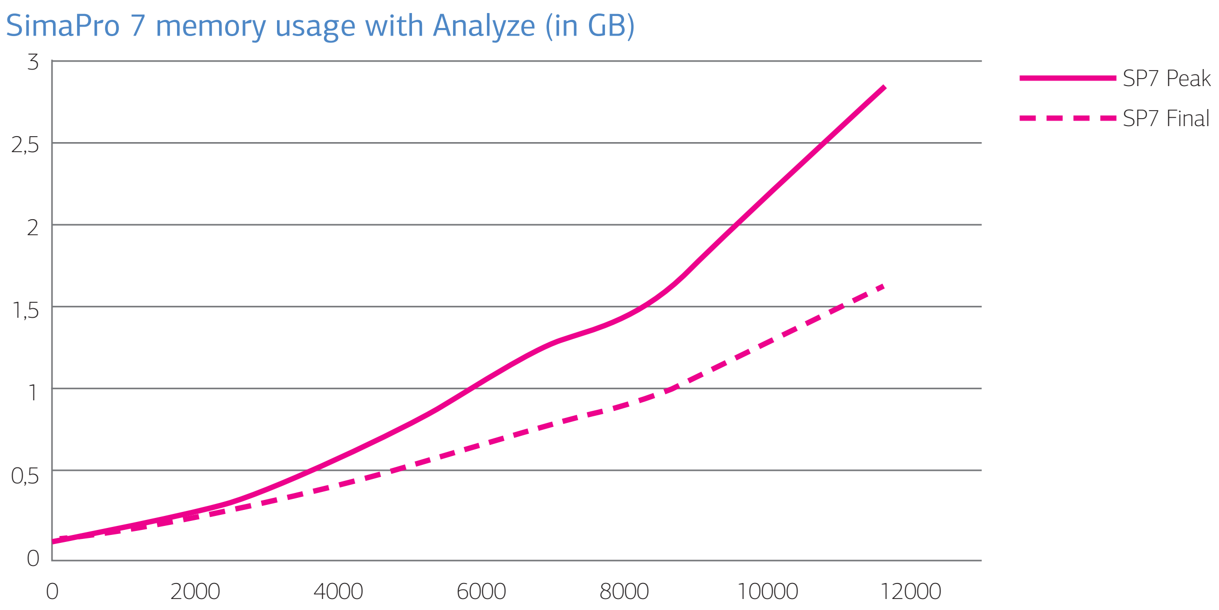


Figure 1. Memory usage of the SimaPro 7 analyze function. Actual memory usage can vary depending on aspects such as the number of libraries selected and the number and type of windows open in the application.

Figure 1 shows SimaPro 7 memory usage for the analyze function, which is similar to network function memory usage. Note that the peak memory usage is much larger than memory usage after the calculation operation is completed. This peak usage determines whether SimaPro can perform the calculation function.

The largest process network SimaPro 7 can calculate when executing the analyze or network function is around 13,000 processes, in an ideal situation only.

The main conclusion of this is:
When using SimaPro 7, we run into problems easily if the number of processes in the LCA models we use is 10,000 or more. Even with fewer processes in the model, it can already be problematic, depending on PC configuration and other factors.

Which leads to this follow-up conclusion:
We need to solve this problem for SimaPro to be used for large LCA models containing 10,000 processes or more. This size can be reached easily when using the latest libraries such as ecoinvent 3.

3 Investigating A Switch To 64-bit Software

The problem is simply a limit to the network size SimaPro 7 can handle. We investigated various solutions, and will discuss the two most promising ones here.

Switching to a 64-bit application initially sounded like an attractive solution, since a 64-bit application can use a very large amount of memory (2^64 bytes). When looking into this solution in more detail, however, it turns out not to work. Most computers that run SimaPro have a limited amount of physical memory, in most cases 4 to 8 GB. Increasing the theoretical amount of memory SimaPro can use would result in only a modest increase in the actual amount of memory SimaPro can use. Raising the system requirements of SimaPro to 16 or 32 GB RAM to significantly increase actual memory availability would not be acceptable for many of our users.

Another obstacle is that, even with enough memory available to complete the calculation, executing a matrix inverse operation for a large LCA model would result in very long calculation times. Tests showed that calculation times could easily increase to several minutes or more. When LCA models become even larger, which we expect to happen with more elaborate databases becoming available, calculation times would become unacceptably long. The logical-sounding solution of using 64-bit software to overcome the memory limitations of SimaPro 7 turns out not to be practically usable.

Instead of increasing the amount of available memory, we had to look for solutions that can complete the necessary calculations using significantly less memory.

4 Introducing The Iterative Method

In cooperation with ModelIt (1), a small company specialising in mathematical solutions for computer software, we developed a new calculation method that calculates results in a series of equal steps. For that reason, it is called the iterative method.

Without going into the mathematical and software implementation details, this new calculation method has a number of important benefits compared to matrix inverse.

1. Low memory usage
1.1. Memory usage increases proportionally with the number of links in the process network
2. High precision
3. Good overall performance
4. Better error detection

The only real disadvantage is that performance depends on the type of data in the process network. Theoretically, in certain unlikely circumstances, calculation time could increase significantly. In practice, this almost never happens.

4.1 Memory Usage

With the iterative method, a process network is transformed into a so-called memory model which stores the necessary data in a highly optimized structure for retrieval, which enables the code to execute the iterative algorithm very efficiently.

The size of this memory model is proportional with the number of links in the process network. This is an improvement compared to matrix inverse, were memory usage increases quadratically with the number of processes in the network.

Even though the size of a process network is measured in the number of links rather than the number of processes, memory usage of the iterative method still increases proportionally to network size. Memory usage no longer increases quadratically with the size of LCA models, regardless how you measure this size.

Tests showed that the iterative method allows SimaPro 8 to calculate process networks consisting of 500,000 ecoinvent 2 unit processes before it hits the limits of its new calculation engine. In a model using ecoinvent 3 processes, with roughly three times more links per process than ecoinvent 2, SimaPro 8 can handle LCA models that include up to 250,000 processes.

Memory usage SimaPro 7 versus SimaPro 8 (in GB)

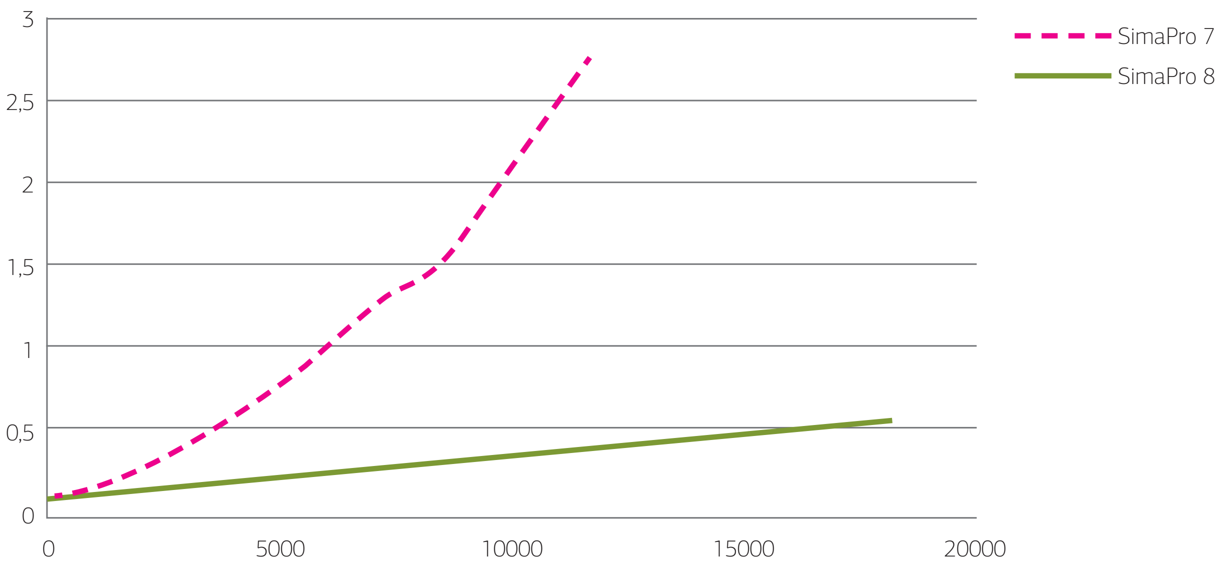


Figure 2. Comparing SimaPro 7 and SimaPro 8 memory usage.

Figure 2 shows memory usage for SimaPro 7 and SimaPro 8. You can see that SimaPro 8 memory usage increases proportionally with the size of the process network, signified by a straight line, and is much lower than SimaPro 7 memory usage.

Conclusion

1. The iterative method solves the main problem with the SimaPro calculation engine and allows it to process large LCA models.
2. Because memory usage increases proportionally with model size, the new calculation engine means that SimaPro 8 can keep up with the increasing size of LCA models for a long time without running into problems.

4.2 Precision

Slightly oversimplified, the iterative method produces more precise results with each iteration executed.

Because of this, we could theoretically reach any required precision simply by increasing the number of iterations. Of course, there is a limit to the precision that can be reached, depending on the type of values the software uses. In SimaPro, all calculations are done with 64-bit floating point values, also known as double precision values, which means the maximum precision possible is about 15 digits. For LCA calculations, this is much higher than necessary, allowing us to set the precision to any value we could reasonably want.

In SimaPro 8, all calculations with the iterative method produce results with a precision of 8 digits or more, except when calculating cumulative indicator values as part of the network function or when calculating intermediate results for the uncertainty analysis function. In these cases, we use a 4-digit precision to reduce calculation time. With these highly calculation-intensive functions, a better performance is often more important and a high precision is not necessary.

If we compare the precision of the SimaPro 7 and SimaPro 8 results, we see important differences.

Table 3. Precision of different calculation methods in SimaPro 7 and 8.

SimaPro version	Method	Precision	Remarks
7	Matrix inversion	Low	About 3 - 4 digits
7	Solver	High	8 digits
8	Iterative	High	- Standard 8 digits - Cumulative indicators: 4 digits - Uncertainty analysis: 4 digits

In Table 3, you can see that matrix inverse produces results with a relatively low precision. Because the largest values have the highest precision, in most cases this is not a problem. Only when small values with a relative low precision have a high impact on the impact assessment results, because of large multiplication factors, the lack of precision could start introducing significant errors.

Conclusion

With the introduction of the iterative method, all results can reach the required precision, leaving occasionally occurring significant deviations a problem of the past.

4.3 Performance

Measuring performance is always tricky - many factors play a role, and if you compare the performance of two functions you must be very clear about what you are comparing and whether that is relevant for the end-user.

Depending on the size of the process network, the iterative method can outperform matrix inverse. With small process networks, matrix inverse and the iterative method have a comparable performance. With increasing network sizes, the iterative method starts becoming faster. With really large networks, matrix inverse takes many minutes while the iterative method still only needs a second or less.

In SimaPro, performance is measured not only by calculation speed but also by the time it takes to load process data of an LCA model into memory. Depending on computer and network configuration, loading data contributes substantially to total execution time. In a multi-user configuration with the database server located on a remote server, loading time is relatively long. On average, it takes about four times longer to load process data with a network configuration than with a single-user stand-alone configuration with both database and software on the local hard disk.

SimaPro 8 uses a database cache buffer. After a process is loaded from the database, it is stored in this local data buffer. The next time SimaPro needs the same process, it is retrieved from this fast buffer instead of the slow hard drive or even slower network server, decreasing loading times dramatically. Loading a process network consisting of 7750 ecoinvent 3 processes takes about 2 minutes the first time the data is loaded from a remote database server, and only takes 10 seconds when it is loaded from the local cache buffer the second time it is needed.

Table 4. Relative performance scores of SimaPro 8 compared to a baseline from SimaPro 7.

Function	SimaPro 7 ecoinvent 2	SimaPro 8 ecoinvent 2	SimaPro 8 ecoinvent 3
Compare	1	3.6	0.5
Analyze	1	3.0	0.5
Network	1	2.0	0.3
Uncertainty analysis (Monte Carlo)	1	80	3.3

Table 4 compares the performance of SimaPro 8 and SimaPro 7. First we compare performance based on the same process data (ecoinvent 2). Loading times will be equal, because SimaPro 7 and SimaPro 8 use the same database management system. Total execution time in SimaPro 8 is roughly one third of total execution time in SimaPro 7, and these speed gains all arise from the faster iterative method of SimaPro 8. In case of Monte Carlo calculations, a number of other improvements also contribute to the increased performance in SimaPro 8.

However, users will most likely not only update the software from SimaPro 7 to SimaPro 8, but will also start using ecoinvent 3 instead of ecoinvent 2. Therefore, we give a more realistic comparison in the last column. Except when doing an uncertainty analysis (Monte Carlo), doing an LCA based on ecoinvent 3 processes in SimaPro 8 does take around 50% longer than doing an LCA using ecoinvent 2 processes in SimaPro 7. This decrease in performance, however, only occurs because ecoinvent 3 LCAs are so much larger that the loading times bias the results.

Conclusion

In-memory calculation functions of SimaPro 8 are faster than those of SimaPro 7, but this performance gain is neutralized because users will use ecoinvent 3 instead of 2, and that requires a larger core library to load before the in-memory calculations can begin.

4.4 Error Detection

When calculating an LCA model, several problems can arise that have to do with errors in the model itself. If a process contains a reference product with amount = 0, it is not possible to calculate the model because at one point the system will try and fail to do a division by zero. In SimaPro, the process editor protects again this error by checking reference amounts before storing them in the database.

But another type of modelling error can go undetected when using matrix inverse or a solver. Loosely described, this error occurs when a process uses more of its own product than it produces. An example will clarify this further.

Assume:

1. Process P1 produces 1 kg Steel, i.e. its reference flow is 1 kg Steel.
2. Process P1 has an input flow of 1.5 kg Steel.

It will be obvious this is not possible in real life, but let's track what happens mathematically in the model. Assuming the demand amount for Steel from process P1 is 1 kg, the following equation holds for product Steel:

1 + 1.5*P1 = 1*P1 =>
0.5*P1 = -1 =>
P1 = -2

So we must produce -2 kg Steel to get 1 kg Steel out of this system, which is nonsense.

Solving the equation yields results that are mathematically correct but must nonetheless be rejected - negative amounts are not possible in real life.

Solving the equations with matrix inverse or a solver would yield exactly the same result. If this problem occurs in a small part of a large process network, the resulting values might all become unreliable – the negative process contributions are, of course, incorrect, but many of the positive ones will also be incorrect if one of the processes with negative results was also used in their calculation.

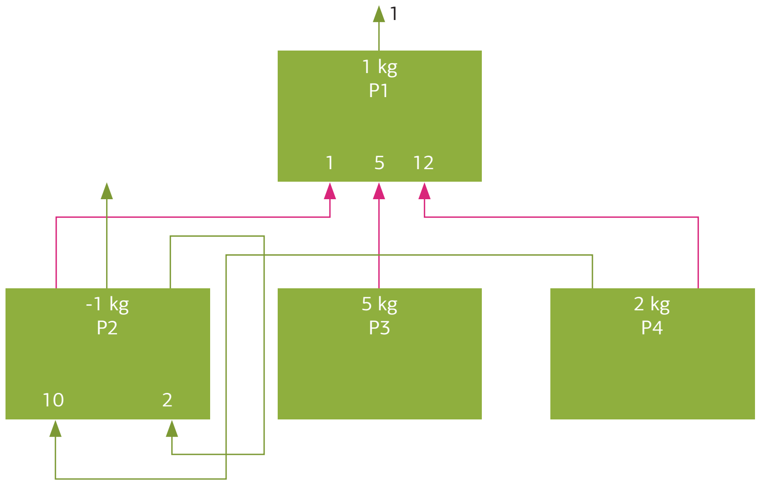


Figure 3. An example of a process network with a modelling error in P2 that also affects other parts of the network.

Figure 3 shows a small process network to illustrate the problem. In this example, the reference amounts of all products are set to 1 unit. Process P2 uses 2 units of its own reference product to produce 1 unit. This error results in a negative calculated amount for P1. But P2 also uses P4 to produce its reference product. Therefore, the calculated amount of P4, even though it is positive, will also be incorrect.

Calculating this process network in SimaPro 8, with the iterative method, protects against errors like this. The iterative method will raise an error indicating there is a problem in the network. It will not give incorrect results without warning like matrix inverse, but will simply refuse to calculate the network once it detects that it cannot be calculated properly.

Even though SimaPro 8 generates an error message in cases like this, it cannot pinpoint exactly which processes and flows of the LCA model are incorrect: this kind of problem can be the result of a combination of many processes.

Conclusion

Unlike matrix inverse, the iterative method will detect an improperly modelled process network in which a process uses, whether directly or indirectly, more of its own product than it produces.

5 Conclusion

With the introduction of the iterative calculation method in SimaPro 8, we managed to solve the network size limit problem that arose as a result of the ever-increasing size of LCA models. In addition, the iterative method also achieves a number of other useful improvements.

6 References

1. ModelIt www.modelit.nla

About SimaPro

The world's leading LCA software chosen by industry, research institutes, and consultants in more than 80 countries. SimaPro provides you with a professional tool to collect, analyze and monitor the sustainability performance of products and services. Learn more about SimaPro

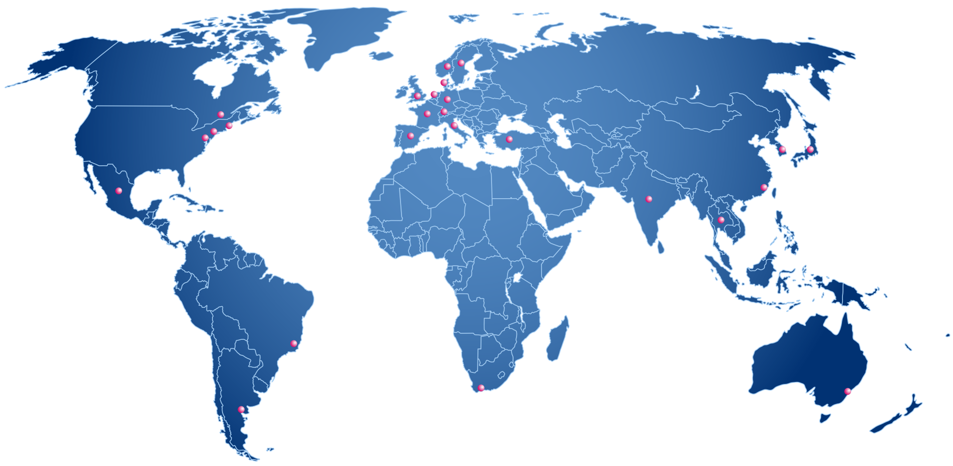
About PRé

At the forefront of sustainability for more than twenty years; focused on Life Cycle Thinking. We are home to world's leading LCA software SimaPro and will help you put the metrics behind your sustainability goals with SimaPro Software and Sustainability Consulting, tailored to your business needs. Your goal is to make a change in the world by becoming a truly sustainable organization. All of our efforts are meant to guide you on that path. Learn more about PRé

SimaPro Global Partner Network | Expert Knowledge and Support

SimaPro thanks its worldwide leadership in LCA software to its Global Partner Network, which extends over 23 countries and is active in more than 80 nations around the world. The SimaPro Global Partner Network is comprised of LCA scientists and sustainability experts. All members of the network provide expert regional support and regular training courses. This partnership is based on expert knowledge and collaboration, the facilitation of large international and multi-client projects and the desire to encourage sustainability adoption worldwide.

Find a partner in your region: http://www.pre-sustainability.com/global-partner-network



Please contact us
for further information:

The Netherlands
PRé Consultants bv
Printerweg 18
3821 AD Amersfoort
The Netherlands

Phone: +31 33 4540 4010
consultancy@pre-sustainability.com

United States
PRé North America Inc.
20 F Street NW
7th Floor
Washington, DC 20001
USA

Phone: +1 202 507 6231
PReNA@pre-sustainability.com

We look forward to being your partner in putting the metrics behind sustainability.

pre-sustainability.com