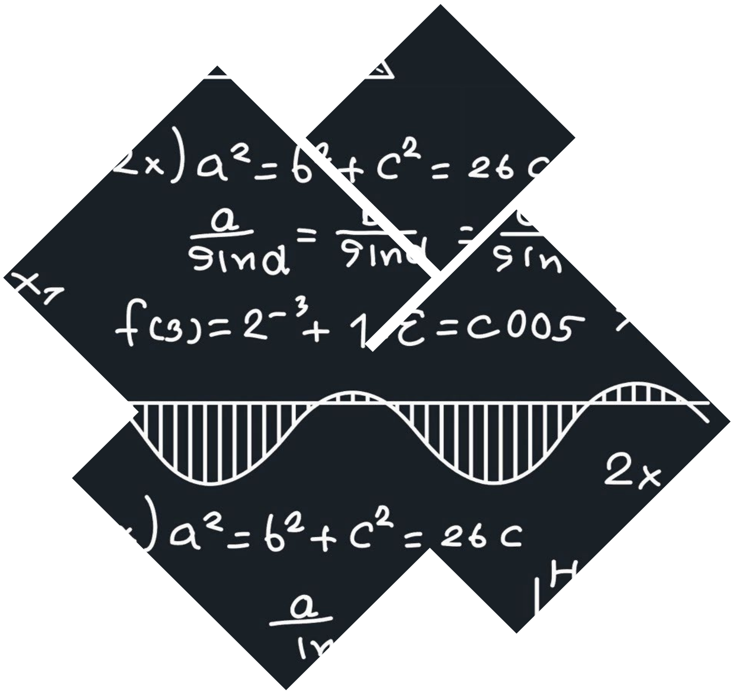
White Paper

Integer Linear Programming with MemComputing

Within minutes, MemComputing found feasible solutions to the f2000 problem, which is an intractable hard problem from MIPLIB; the feasibility of which had remained unknown since 2010…

MemComputing Inc.



EXECUTIVE SUMMARY

Integer linear programming (ILP) encompasses a very important class of optimization problems that are of great interest to both academia and industry. Several algorithms are available that attempt to explore the solution space of this class efficiently, while requiring a reasonable compute time. However, **these algorithms still face considerable challenges when confronted with particularly hard problems,** such as those found in the MIPLIB library.

We propose a radically different **non-algorithmic approach** to ILP based on our novel physics-inspired computing paradigm known as memcomputing. This paradigm is based on digital memcomputing machines (DMM) that are scalable in nature. DMMs are built using a new circuit architecture that also contains memory. DMMs can be built in hardware or their equations of motion can efficiently be emulated in software on traditional computers.

Originally, we introduced DMMs that represented circuits of self-organizing logic gates. These specifically satisfied traditional binary logic problems. In order to solve for the linear inequalities representing a general ILP problem, we have developed a new architecture of self-organizing algebraic circuits. These circuits self-organize dynamically to satisfy the correct (algebraic) linear inequalities. As with standard DMMs, these machines can be built in hardware or emulated in software. The computations and results discussed in this paper are based upon our software emulation of the self-organizing algebraic solution, known as the MemCPU™ Coprocessor (available now as a cloud based Software as a Service Solution). **In this paper we show that we find feasible solutions to an intractable ILP problem, (f2000 from MIPLIB) that had previously been unsolved since presented as a challenge in 2010.**

2 MemComputing: An Innovative Approach

The memcomputing approach to ILP problems is based on the **Self-Organizing Algebraic Gates (SOAGs)** concept. SOAG is a novel circuit design developed at MemComputing, Inc. and is inspired by the previous work on **Self-Organizing Logic Gates (SOLGs).** Both SOLGs and SOAGs are building blocks for practical realizations of **Universal Memcomputing Machines (UMM)**, in particular their digital (hence scalable) sub-set: **Digital Memcomputing Machines** (DMMs).

SOLGS Properties in a nutshell

- Input and output related to the targeted problem by associating logical 0s or 1s to voltages that are below or above a threshold.
- Potential to form Self-Organizing Logic Circuits (SOLCs) demonstrating long-range order and topological robustness.
- Proper design brings significant advantages including avoidance of persistent chaotic & oscillatory behavior.
- SOLCs are digital realizations of UMMs because they accept inputs & return outputs that are digital. In this way, the required precision in writing inputs and reading outputs is finite and independent of the size of the problem at hand. However, the transition function of these machines (namely the function that maps input to output) is physical (analog) and takes full advantage of the collective state of the system to process information.
- DMMs scale easily because they don't require precision that increases with the problem size. Rather, they can handle, ideally, unbounded problem sizes.
- High efficiency in a variety of combinatorial optimization problems such as maximum satisfiability (MAXSAT), quadratic unconstrained binary optimization (QUBO), spin-glasses, and pre-training of deep-belief networks.
- Time-to-solution observed to increase linearly as the size of the targeted problem increases, where standard approaches show an exponential explosion in time-to-solution.

SOAGs Properties in a nutshell

SOAGs share the same principles and scalability advantages of SOLGs but their circuit is designed to self-organize toward an algebraic relation rather than a boolean relation as for SOLGs.

SOAGs have been designed to satisfy linear relations between boolean variables as a particular case of algebraic relations (see Fig. 1). Further extensions of this design will include mixed integer and continuous variables, as well as nonlinear algebraic relations. By connecting together SOAGs, we then assemble a Self-Organizing Algebraic Circuit (SOAC). The SOAC collectively self-organizes in order to satisfy the relations embedded in the self organizing logic gates.

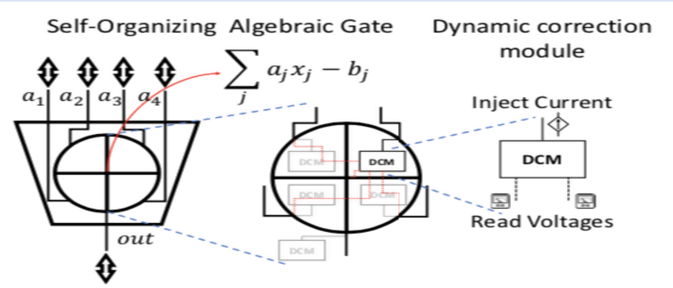


FIG. 1. Sketch of a Self-Organizing Algebraic Gate. All terminals allow a superposition of incoming and outgoing signals from the surrounding circuit. The central unit processes the signals in order to satisfy a liner algebraic relation consistent with the requirement of the "out" terminal. The self-organization is enforced by the Dynamic Correction Modules that read voltages from all terminals and inject a current to the appropriate terminal as long as the algebraic relation is not satisfied.

Like the SOLCs, the ultimate physical electronic circuit representing an SOAC contains active and passive elements, with and without memory (internal state variables).

The corresponding electrical circuit can be built with available Complementary Metal Oxide Semiconductor (CMOS) technology. Since its components are non-quantum, the ordinary differential equations describing it can be efficiently simulated in software on our modern computers.

3 DEMONSTRATED PERFORMANCE (1/2)

MemCPU: the first to solve the f2000 problem

What is MIPLIB ?
The Mixed Integer Programming LIBrary

In response to the needs of researchers for access to real-world mixed integer programs a group of researchers created MIPLIB in 1992, a library of both pure and mixed integer programs (https://miplib.zib.de/).

This was updated by researchers in 1996, 2003, 2010 and again in 2017 by a larger initiative from academia and industry. The library is managed by researchers from the Zeus Institute of Berlin (ZIB).

Since its introduction, **MIPLIB has become a standard test set used to compare the performance of mixed integer optimizers**. Its availability has provided an important stimulus for researchers in this very active area.

Introduction to the f2000 problem
The f2000 problem of the MIPLIB library was first introduced in 2010. The problem belongs to the class of hard random problems with an "open" status. The "open" status in MIPLIB identifies problems that have not been solved or even determined as feasible to solve.

The researchers from ZIB selected the f2000 problem from the pseudo-boolean competition of 2010, which was held as part of the 2010 Satisfiability (SAT) conference.

Since then, f2000 has remained in the pseudo-boolean competition and has also remained an open/unsolved problem in the MIPLIB library.

Many groups from both the SAT and MIP communities have tried to find feasible solutions for this problem, using both complete and incomplete solvers; however, none have presented any evidence of success.

First to solve the F2000 problem

The feasibility of f2000 is classified as "unknown" by MIPLIB.

Running this problem using the MemCPU Coprocessor, MemComputing Inc. found the first feasible solution to the problem to this previously open problem. And MemComputing **found the solution within 60 seconds.**

Continuing to run for longer times results in **more solutions with objectives of increasing quality**. So, not only was MemComputing the first to discover a feasible solution, it found many feasible solutions!

This impressive result shows the uniqueness and power of the memcomputing approach.

3 DEMONSTRATED PERFORMANCE (2/2)

Scaling with problem size - Open Pit Mining Problem

The f2000 problem is a great example of MemComputing's ability to solve intractable and unsolved problems. It implies scalability, but there are other problems in MIPLIB that are specifically designed to demonstrate scale in size and complexity.

The rMine benchmark is a series of problems that model an open pit mining problem. MIPLIB contains five versions of the rMine benchmark where the numbers of variables and problem complexity increases.

As documented in the scientific publication on these tests, MemComputing Integer Linear Programming (https://arxiv.org/abs/1808.09999v1), we ran the problem head to head against a commercially available best in class solver for mixed integer programming.

The study shows that for the smaller problem instances, the commercial solver converges very quickly to a solution within 1% of the global minimum. However, by increasing the number of variables and thusly increasing the problem complexity, the commercial solver only came within 10,000+ % of the global minimum. The MemCPU Coprocessor converged slightly slower on the small instances in order to reach a solution within 1% of the global minimum. However, as the problem complexity increased, the MemCPU Coprocessor maintained its scaling and continued to provide a solution within 1% of the global minimum, 5 orders of magnitude more accurate than the commercial solver.

The key aspect to MemComputing is that it is designed to work better the larger and more complex a problem becomes. On smaller problem instances, it is more economical to use current commercial solvers. However, as the problems become more complex and these solvers begin to take hours, days and even weeks to solve; MemComputing can often provide equivalent or even better solutions in minutes or seconds. In a world where time is money, the value of the MemCPU Coprocessor becomes more and more apparent the more compute intensive a problem becomes.

The Open-pit mining Problem

An open pit mine can be described by a three-dimensional array of blocks, each of which is assigned a number of values defining its characteristics.

Scheduling an open pit consists in finding a sequence in which the blocks should be removed from the mine in order to maximize the total discounted profit from the mine subject to a variety of technical and economic constraints.

This problem is industrially relevant and has been heavily studied by the MIPLIB organizers themselves

4 CONCLUSION

The use of digital (hence scalable) memcomputing machines to tackle the important problem class of integer linear programming problems proves very compelling. The MemCPU Coprocessor represents a new paradigm in computing industry's most complex and compute intensive ILP problems.

We have simulated the corresponding equations of motion of these circuits; that is, we have emulated the MemComputing solution in software. With the software emulation we have found solutions to a variety of benchmark ILP problems as reported in the MIPLIB library. **In particular, we found feasible solutions for a problem that had eluded many (the MIPLIB f2000 problem). We found a solution within 1 minute, while many have spent 100's if not 1000's of compute hours and failed to find a solution.** We then showed how MemComputing scales and is able to accurately solve problems with the expected accuracy but orders of magnitude faster than today's best in class solutions.

Please note that the version of the MemCPU Coprocessor used in this evaluation was the initial prototype. A commercialized version of the MemCPU Coprocessor is available as a cloud based SaaS solution. Additional enhancements and optimizations have been built into the commercial version that have improved the performance over what is shown here/

- **Solved previously unsolvable problem!**
- **The more complex and compute intensive a problem, the better that MemComputing performs.**
- **Available today as a SaaS Solution**

MemComputing, Inc.'s disruptive coprocessor technology is accelerating the time to find feasible solutions to the most challenging operations research problems in all industries. Using physics principles, this novel software architecture is based on the logic and reasoning functions of the human brain.

MemComputing enables companies to analyze huge amounts of data and make informed decisions quickly, bringing efficiencies to areas of operations research such as Big Data analytics, scheduling of resources, routing of vehicles, network and cellular traffic, genetic assembly and sequencing, portfolio optimization, drug discovery and oil and gas exploration.

The company was formed by the inventors of MemComputing, PhD Physicists Massimiliano Di Ventra & Fabio Traversa and successful serial entrepreneur, John A. Beane.

MemComputing Inc.
4250 Executive Square,
Suite 200,
La Jolla, CA 92037

http://www.memcpu.com
info@memcomputing.com