Team Amalgam

SE390 Research Plan

Joseph Hong, Chris Kleynhans, Ming-Ho Yee, Atulan Zaman
{yshong,cpkleynh,m5yee,a3zaman}@uwaterloo.ca

November 25, 2012

Abstract

SE390 Research Plan for Team Amalgam

Contents

1 Problem Definition **2**

2 Related Work **3**

3 Research Value **3**
3.1 Aerospace 4
3.2 Civil Engineering 4
3.3 Software Engineering 4

4 Goals **4**

5 Methodologies **5**

6 Risks & Technical Feasibility **5**

7 Costs **6**

8 Legal/Social/Ethical Issues **6**

References **6**

1 Problem Definition

Multi-objective optimization is a widely researched area of computer science that focuses on finding solution to problem definitions with respect to given objective realization constraints. Computing such problems is extremely resource intensive and the computation time grows exponentially with the number of optimization variables.

The nature of our work in scientific terms is called *exact, discrete multi-objective optimization*. Multi-objective optimization (MOO) is the process of computing the most optimized solution given a goal and a set of constraints. The reason it is called *multi-objective* is because multiple constraints are being computed for optimization at the same time as using the constraints, which means there could be more than one optimal solution that could satisfy the constraints that satisfy various optimization goals. Since optimization for different goals requires verification of many permutations of constraint combinations, the processing time for such problems rise exponentially with the number of dimensions required for optimizing.

One simple example of a multi-objective optimization problem is the satellite scheduling problem. In this problem [1], NASA needs to figure out the best possible scheduling routine for their satellites, each of which have different purpose and are of interest to different scientific communities. In this problem, the constraints for NASA are such things as resource limitations and launch ordering constraints and the objectives to solve for the values it relates to for the different science cohorts.

Exact in the definition of the problem indicates that all solutions computed by this algorithm is Pareto-optimal. Which means that each of the computed Pareto-optimal solutions satisfy the condition that no optimization goals can be made better off without compromising at least one other optimization goal.

Discrete indicates that our optimization algorithm only addresses discrete input data as constraints and optimization goals and does accept or produce continuous optimality conditions.

Moolloy [2] is a tool created in the MIT CSAIL labs that implements the algorithm described above, and it also has a GUI that lets the user specify the constrains and objective condition as well graphically view the Pareto-optimal solutions computed by the MOO algorithm. The problem with multi-objective optimization is that as the number of dimensions of the optimization goal increases, the Pareto-front of the problem also increases exponentially which causes a leap in the processing time. While using a un-optimized version of the *guided improvement algorithm* does give solutions;

however, its scalability is greatly handicapped by the how time consuming the computation becomes with a large problem space.

Therefore this work will focus on increasing the optimality as well as the scalability of Moolloy by addressing various relational logic optimization techniques without undermining the integrity of the MOO solutions.

2 Related Work

In 2009, Rayside, Estler, and Jackson [2] proposed the *guided improvement algorithm*. In their paper, they also conducted a literature survey on related work. Rayside et al. found that most multi-objective problems were concerned with continuous variables, as opposed to discrete, or combinatorial, variables.

Furthermore, most of the research on multi-objective optimization was focused on heuristic approaches, specific instead of general solvers, extensions of single-objective solvers, or problems with only two or three variables.

The *guided improvement algorithm* is an exact, discrete, general-purpose solver. Furthermore, it is not an extension of a single-objective approach. Rayside et al. identified a similar approach they call the *opportunistic improvement algorithm*, which was independently discovered by Gavanelli [3] and Lukasiewycz et al. [4]. Notably, the *guided improvement algorithm* produces intermediate Pareto-optimal results.

Only three publications have cited the *guided improvement algorithm* paper since it was published. One of them is about developing a user interface, while the other two concern applications of the algorithm. Thus, none of them are related to our work, which is strictly to optimize the algorithm.

A more recent paper by Dhaenens, Lemesre, and Talbi [5] proposes *K-PPM*, which can be parallelized. However, the algorithm does not produce intermediate Pareto-optimal results.

3 Research Value

There are many fields in which multi-objective optimization problems appear. By improving the performance and scalability of the algorithm we will enable its usage for problems with larger sample spaces. Three fields that we have identified that may benefit from an optimized algorithm are aerospace, civil engineering, and software engineering.

3.1 Aerospace

Every ten years NASA performs its decadal survey to determine which missions it will undertake for the next decade [1]. Multi-objective optimization can be used to determine a launch schedule that maximizes the scientific value the missions provide to different scientific communities while minimizing cost. Such a problem also requires constraints to be satisfied. For example, one mission may be dependent on another or a mission may need specific timing.

3.2 Civil Engineering

Professor Bryan Tolson has identified a number of problems he is researching that are multi-objective optimization problems. Currently, the problems are solved using heuristic methods or genetic algorithms. As discussed earlier, these methods do not guarantee that their solution is the best result. One of the problems he is interested in is determining the optimal materials to use for each of a landfill's lining layers, to minimize both seepage and cost.

3.3 Software Engineering

One applicable Software Engineering problem is that of Software product lines. In such a problem we wish to determine which modules we wish to include in the software for an embedded device. Each module can perform different functions and these functions may conflict with other modules. Additionally each module will have a different cost in terms of code size and different performance metrics. We wish to determine what would be an optimal set of modules for a device that requires certain functions.

4 Goals

The goal of this project is to reduce the computation time for Moolloy of large problem spaces with many optimization goals and to increase the scalability of Moolloy so that by the end of this project Moolloy can successfully computes solutions to optimality problems within a comfortable time bound, that are out of reach in the current version. Part of meeting this goal would mean creating a regression suite to make sure the results we are getting from optimization are still reliable and that might a comparison metric for the new results with the original results during our regular build system.

5 Methodologies

The methodologies that we are considering as a logical starting point are the following, which are results of previous works by researchers in this field and also by Derek Rayside and his collaborators. As we proceed with the project it possible we might find some of these optimization ideas not very useful and we might come up with other techniques that might be more relevant.

- Parallel Decomposition
- Sequential Decomposition
- Input Space Reduction
- Duality
- Empirical Profiling
- Improve Search Guidance / Speculative Execution
- Workflow Feedback
- Incremental SAT Solving

6 Risks & Technical Feasibility

One of the risks of this project is that we might not be able find ideal use cases to test our optimization ideas in which they apply. Because different optimality problems have various possible routes for optimization it could both be difficult to find the perfect optimization scheme, that addresses most models as well as finding models that are ideal for our algorithms. This is why we are working with multiple collaborators to increase our problem space. That includes the graduate students in WATFORM who work on solving relevant problem sets and also Prof. Bryan Tolson from Department of Civil Engineering who has multiple problems that might be more helpful to solve using MOO.

Another unlikely risk for this project is that none of our algorithms provide satisfactory results in terms of performance benchmark, in which case it would still have research value however, little research impact since there would not be a lot of benefit to users from this.

Time is always a risk in research project such as this, because we might not be able to explore all our optimization ideas. Therefore we are paying close attention to which algorithms we want to explore first on the basis of how effective they might seem from initial analysis.

7 Costs

8 Legal/Social/Ethical Issues

References

[1] NASA. (2011). Decadal Survey, [Online]. Available: `{http://science.nasa.gov/earth-science/decadal-surveys/}`.

[2] D. Rayside, H.-C. Estler, D. Jackson, "The Guided Improvement Algorithm for Exact, General-Purpose, Many-Objective Combinatorial Optimization," MIT CSAIL, Tech. Rep., 2009.

[3] M. Gavanelli, "An Algorithm for Multi-Criteria Optimization in CSPs," in *Proceedings of the 15th European Conference on Artificial Intelligence*, Frank van Harmelen, Ed., Lyon, France: IOS Press, 2002, pp. 136–140.

[4] M. Lukasiewycz, M. Glaß, C. Haubelt, J. Teich, "Solving multi-objective pseudo-boolean problems," in *Proceedings of the 10th international conference on Theory and applications of satisfiability testing*, ser. SAT'07, Lisbon, Portugal: Springer-Verlag, 2007, pp. 56–69.

[5] C. Dhaenens, J. Lemesre, E.G. Talbi, "K-PPM: A new exact method to solve multi-objective combinatorial optimization problems," *European Journal of Operational Research*, vol. 200, no. 1, pp. 45–53, 2010.