The Optimizing Web: Leveraging Agent Technology for Sustainability

Aditya Ghose

Decision Systems Laboratory
School of Computer Science and Software Engineering
University of Wollongong, NSW 2522 Australia
aditya@uow.edu.au

Abstract. This paper describes how agent technology might be leveraged to deliver a critical solution to the carbon mitigation challenge - in the context of the Optimizing Web project.

1 Introduction

There is widespread recognition of the climate change crisis, and the need to develop scientific, technological and managerial responses. Current thinking on climate change responses emphasizes the development of alternative energy sources, the development of smart automotive technology and the introduction of macro-economic levers (e.g., carbon taxes, emission trading schemes) to alter energy consumption behaviour at the level of both enterprises and individuals. While these are laudable objectives, these initiatives might be somewhat premature. We do not yet have the ability to ensure *efficient utilization* of existing infrastructure - it stands to reason that this must be a necessary first step before we make massive investments in novel technological bases for energy generation or transportation infrastructure.

The notion of *efficient resource utilization* is inextricably tied to the notion of *optimization* - in particular, the ability to optimize energy use - yet this has been largely ignored in the current discourse. The connection between optimization and carbon mitigation is compelling: optimization enables efficient resource utilization, thus lowering energy consumption and the carbon footprint. The global industrial/technological infrastructure, including transportation systems, manufacturing plants, human habitat and so on, is typically operated in an ad-hoc and significantly sub-optimal fashion. This remains the case despite the availability of sophisticated optimization technology for almost the past seven decades (present day operations research techniques trace their roots to the pioneering work of George Dantzig in the early 1940s that resulted in the original optimization algorithm - linear programming).

Most applications of optimization technology have been in piecemeal, monolithic optimization systems. Yet the climate change crisis requires optimization on a large-scale, and in a manner that permits entities in a massive planetary supply chain (it can be instructive to view the global network of human activity as such) to collaborate to achieve the commonly agreed upon carbon mitigation objective. Traditional stand-alone ”batch” optimization technology cannot be deployed in this setting for a variety of reasons. It is impractical to conceive of a centralized “global optimizer”'. It is equally impractical to expect business entities to reveal what if often highly sensitive information about their supply chains to central optimizer. Finally, the scale of the problem is too large to be feasibly addressed. The problem, then, is to support decentralized, distributed, collaborative optimization on a global scale. The nearest point of departure for such an enterprise is the literature on agent-based, distributed optimization.

The climate change crisis has presented the community of researchers interested in agent technology with a historic opportunity. For the first time ever, we have a globally agreed-upon objective function: the carbon footprint minimization objective. This opens up the possibility for devising large-scale, agent-based, *collaborative optimization architectures*, where large numbers of agent-based optimizers solve local optiimization problems, while collaborating to improve the cumulative system performance relative to a shared objective function. The Optimizing Web project (see www.optimizing-web.org) seeks to design and validate the conceptual underpinnings of an infrastructure that would support very large scale collaborative optimization across a potentially global collection of optimizers. The Optimizing Web project grew out of the University of Wollongong Carbon-Centric Computing Initiative (see www.ccci.uow.edu.au) which has the broader agenda of exploring ways in which the full spectrum of computing technologies might contribute to solutions to the climate change crisis. The Optimizing Web vision is to provide ubiquitous collaborative optimization services, at the level of individual devices, vehicles within transportation systems, units within organizations or manufacturing plants - as well aggregations of all of these. The Optimizing Web would be a systems of systems, and would provide a protocol (or a set of protocols) for local optimizers to inter-operate to optimize the global carbon footprint minimization objective, while making appropriate trade-offs in relation to their local objectives. While the modelling and solution of ”local” optimization has been the focus of attention for the operations research (OR) community for several decades, this project addresses the question of how large collections of optimization problems (with associated solvers), with possibly intersecting signatures (sets of common variables), might be made to inter-operate to optimize a shared function (the carbon footprint minimization objective).

There are three specific challenges for the agents community: the design of agent-based *optimization architectures*, the development of the next generation of agent-based distributed optimization protocols and the integration of optimization with distributed agent-based planning. We address the first of these in some detail below.

2 Agent-based optimization architectures

Fundamental to the optimizing web is the notion of an optimization architecture, i.e., a collection of appropriately configured inter-operating optimizers. It specifies the constituent optimizers, their signatures (the decision variables whose values are determined by the optimizer in question), their parameters (the variables whose values constrain the optimizer), and the nature of the inter-agent coordination messages exchanged. The architecture is agnostic to the internals of individual optimizers. We might design an optimization architecture from scratch, or we might engineer one around pre-existing, legacy optimizers. Both approaches present challenges. Key to understanding optimization architectures is an appreciation of the competing pulls of *local* objectives and system-wide (societal or global) objectives, and the implications of resolving them in different ways. Consider an agent-based traffic planning setting (Srivastav, 2011), where individual road users get routing advice from decision-support agents executing on hand-held devices (smartphones, PDAs etc.). Our empirical studies confirm the intuition that locally sub-optimal decisions at the level of individual road users can contribute to improving the system-wide objective (of reducing the cumulative carbon footprint, for example). Sometimes, an individual road-user on a short-hop journey will need to be incentivized to take a longer route, in the interests of reducing the cumulative carbon footprint of road users on longer journeys who would have been obliged to take significantly longer routes to avoid the congestion that the short-hop users might have contributed to. Similarly, our empirical work on designing optimal resource allocation mechanisms in clinical settings [1] suggests that making patients incur a small wait-time (within clinically acceptable limits) achieves far better system-wide efficiencies than a "first-come-first-served"' logic. Foremost amongst these is the notion of objective alignment (or consistency). An objective function determines the optimization behaviour of an agent, i.e., the choices it makes amongst possible alternative solutions. Objective alignment helps ensure that optimizers use objectives that are aligned with the global carbon footprint minimization objective. Given a set of objective functions, we need to be able to determine if these jointly determine a consistent set of optimization behaviours. Consider the pair of objective functions minimize x and minimize -x. If the set of feasible solutions (i.e., solutions that satisfy all of the applicable constraints) is non-singleton, then an agent will not be able to satisfy both objectives (since they, in effect, "pull in opposite directions"). If there is exactly one feasible solution, however, the pair of objectives is in fact aligned. Similarly, the objectives minimize x and minimize x2 are not aligned in general, but may be aligned if x is restricted to be positive. Definitions of objective alignment did not exist in the literature, until our preliminary work in [2], where we view an objective function as a preference relation defined over the set of feasible solutions. Alignment then reduces to the absence of contradictory preferences. While this approach provides the conceptual framework for understanding objective alignment, it does not immediately lead to practical tools for checking alignment. A major challenge is the fact that alignment cannot be determined on the basis of the objectives alone, but is also contingent on the set of applicable constraints, and hence the set of feasible solutions (as the examples above illustrate). Additionally, exhaustively enumerating the preference relation on the set of feasible solutions is impractical, yet there are no easy approaches to performing alignment checking analytically. A compromise is to perform approximate checking using heuristic techniques, with no guarantee of completeness. The methodological bases for designing optimization architectures need to be defined. Ensuring that the objectives within an optimization architecture are aligned with the global carbon footprint minimization objective by design also requires the ability to decompose objectives (for instance, how do we start with a set of high-level organizational objectives and decompose these into the objectives of the constituent business units, while maintaining consistency with a global objective?). Finally, we need to understand how to measure (or monetize) the trade-offs between the local objectives of an optimizer and the global (carbon mitigation) objective. In other words, we need to devise mechanisms to incentivize an agent to adopt behaviour that is potentially sub-optimal relative to its own objectives, in the interests of the global objective.

There are several other interesting challenges. We need to be able devise means for agents to *discover footprints*, i.e., answer the question: which agents does a given agents share constrainst with? Sometimes the answer to this question is relatively static, but in other settings (such as traffic planning) the answer can be highly dynamic, and some modicum of predictive reasoning is required. The maintenance of optimization architectures in highly dynamic settings is another major challenge. The design of social mechanisms (such as carbon credits) to incentivize agents to adopt locally sub-optimal behaviour poses challenges. Existing agent-based optimization protocols need to extended, as do agent communication standards to enable the kinds of messaging/negotiation necessary in this context. Finally, we need to be able to *disocver constraints and objectives* by mining the "big data"' repositories that our current technologies have made possible.

3 References

[1] Billiau G. and Ghose A. (2011). Agent-based optimization in healthcare. Decision Systems Lab, Univ. of Wollongong Technical Report 2011-TR-02.
[2] Dasgupta, A. and Ghose, A. K. (2010). Implementing reactive BDI agents with user-given constraints and objectives. Int'l Journal of Agent-Oriented Software Engineering (IJAOSE) 4(2): 141-154, 2010.
[3] Srivastav, B. Billiau, G. Lim, M., Lee T. and Ghose, A. (2011). Optimal traffic planning. Decision Systems Lab, Univ. of Wollongong Technical Report 2011-TR-01.