Adaptive, distributed control of constrained multi-agent systems

Stefan Bieniawski
253 Durand, Dept. of Aeronautics
Stanford, CA 94305, stefanb@stanford.edu

David H. Wolpert
NASA Ames Research Center, Moffett
Field, CA 94035, dhw@ptolemy.arc.nasa.gov

Abstract

Product Distribution (PD) theory was recently developed as a broad framework for analyzing and optimizing distributed systems. Here we demonstrate its use for adaptive distributed control of Multi-Agent Systems (MAS's), i.e., for distributed stochastic optimization using MAS's. First we review one motivation of PD theory, as the information-theoretic extension of conventional full-rationality game theory to the case of bounded rational agents. In this extension the equilibrium of the game is the optimizer of a Lagrangian of the (probability distribution of) the joint state of the agents. When the game in question is a team game with constraints, that equilibrium optimizes the expected value of the team game utility, subject to those constraints. One common way to £nd that equilibrium is to have each agent run a Reinforcement Learning (RL) algorithm. PD theory reveals this to be a particular type of search algorithm for minimizing the Lagrangian. Typically that algorithm is quite ine£cient. A more principled alternative is to use a variant of Newton's method to minimize the Lagrangian. Here we compare this alternative to RL-based search in three sets of computer experiments. These are the N Queen's problem and bin-packing problem from the optimization literature, and the Bar problem from the distributed RL literature. Our results con£rm that the PD-theory-based approach outperforms the RL-based scheme in all three domains.

1. Introduction

Product Distribution (PD) theory is a recently introduced broad framework for analyzing, controlling, and optimizing distributed systems [16, 17, 18]. Among its potential applications are adaptive, distributed control of a Multi-Agent System (MAS), (constrained) optimization, sampling of high-dimensional probability densities (i.e., improvements to Metropolis sampling), density estimation, numerical integration, reinforcement learning, information-theoretic bounded rational game theory, population biology, and management theory. Some of these are investigated in [2, 1, 13, 11].

Here we investigate PD theory's use for distributed stochastic optimization using a MAS (which for our purposes is the same as adaptive, distributed control of a MAS). Often in stochastic approaches to optimization one uses probability distributions to help search for a point $x \in X$ optimizing a function $G(x)$. In contrast, in the PD approach one searches for a probability distribution $P(x)$ that optimizes an associated Lagrangian, $\mathcal{L}(P)$. Since P is a vector in a Euclidean space, the search can be done via techniques like gradient descent or Newton's method — even if X is a categorical, £nite space.

One motivation of this approach embodied in PD theory starts with the fact that in any game the agents are independent, with each agent i choosing its move x_i at any instant by sampling its probability distribution (mixed strategy) at that instant, $q_i(x_i)$. Accordingly, the distribution of the joint-moves is a product distribution, $P(x) = \prod_i q_i(x_i)$. In this representation of a MAS, all coupling between the agents occurs indirectly; it is the separate distributions of the agents $\{q_i\}$ that are statistically coupled, while the actual moves of the agents are independent.

This representation has been adopted implicitly before, in algorithms that £nd the equilibria by having each agent run its own Reinforcement Learning (RL) algorithm [15, 6, 10, 20, 21, 19]. In these approaches the utility function of each agent is based on the **world utility** $G(x)$ mapping the joint move of the agents, $x \in X$, to the performance of the overall system. However the agents in a MAS are bounded rational. Moreover the equilibrium they reach will typically involve mixed strategies rather than pure strategies, i.e., they don't settle on a single point x optimizing $G(x)$. This suggests formulating an approach that explicitly accounts for the bounded rational, mixed strategy character of the agents.

This is done in PD theory, which uses information theory to recast the optimization problem as one of minimizing a Lagrangian, $\mathcal{L}(P)$, rather than settling to the equilibrium of the game. From the perspective of PD theory, the update rules used by the agents in RL-based systems are just

one particular set of (ineff£cient) ways of £nding that minimizing distribution [16, 17]. More principled alternatives like variants of Newton's should perform better. In addition, such alternatives allow us to leverage well-understood techniques of convex optimization for incorporating constraints over X. In contrast, RL-based schemes typically incorporate constraints in an ad hoc fashion, via penalty functions.

Here we compare this alternative to RL-based search algorithms in three sets of computer experiments. These experiments also show how the perspective of PD theory can be used to incorporate constraints into RL-based search algorithms without relying on ad hoc penalty functions.

In the next section we review the game-theory motivation of PD theory. We then present details of our Lagrangian-minimization algorithm. We end with computer experiments comparing this algorithm to some state-of-the-art RL-based algorithms. These experiments involve the N Queen's problem and bin-packing problem from the optimization literature, and the Bar problem from the distributed RL literature. Our results con£rm that the PD-theory-based approach outperforms the RL-based scheme in all three domains.

2. Bounded Rational Game Theory

In this section we motivate PD theory as the information-theoretic formulation of bounded rational game theory.

2.1. Review of noncooperative game theory

In noncooperative game theory one has a set of N **players**. Each player i has its own set of allowed **pure strategies**. A **mixed strategy** is a distribution $q_i(x_i)$ over player i's possible pure strategies. Each player i also has a **private utility** function g_i that maps the pure strategies adopted by all N of the players into the real numbers. So given mixed strategies of all the players, the expected utility of player i is $E(g_i) = \int dx \ \prod_j q_j(x_j) g_i(x)$ [1].

In a **Nash equilibrium** every player adopts the mixed strategy that maximizes its expected utility, given the mixed strategies of the other players. More formally, $\forall i, q_i = \text{argmax}_{q_i'} \int dx \ q_i' \prod_{j \neq i} q_j(x_j) \ g_i(x)$. Perhaps the major objection that has been raised to the Nash equilibrium concept is its assumption of **full rationality** [8, 9, 3]. This is the assumption that every player i can both calculate what the strategies $q_{j \neq i}$ will be and then calculate its associated optimal distribution. In other words, it is the assumption that every player will calculate the entire joint distribution $q(x) = \prod_j q_j(x_j)$. If for no other reasons than computational limitations of real humans, this assumption is essentially untenable.

2.2. Review of the maximum entropy principle

Shannon was the £rst person to realize that based on any of several separate sets of very simple desiderata, there is a unique real-valued quanti£cation of the amount of syntactic information in a distribution $P(y)$. He showed that this amount of information is (the negative of) the Shannon entropy of that distribution, $S(P) = -\int dy \ P(y) ln[\frac{P(y)}{\mu(y)}]$. So for example, the distribution with minimal information is the one that doesn't distinguish at all between the various y, i.e., the uniform distribution. Conversely, the most informative distribution is the one that speci£es a single possible y. Note that for a product distribution, entropy is additive, i.e., $S(\prod_i q_i(y_i)) = \sum_i S(q_i)$.

Say we given some incomplete prior knowledge about a distribution $P(y)$. How should one estimate $P(y)$ based on that prior knowledge? Shannon's result tells us how to do that in the most conservative way: have your estimate of $P(y)$ contain the minimal amount of extra information beyond that already contained in the prior knowledge about $P(y)$. Intuitively, this can be viewed as a version of Occam's razor. This approach is called the maximum entropy (maxent) principle. It has proven useful in domains ranging from signal processing to supervised learning [5, 12].

2.3. Maxent Lagrangians

Much of the work on equilibrium concepts in game theory adopts the perspective of an external observer of a game. We are told something concerning the game, e.g., its utility functions, information sets, etc., and from that wish to predict what joint strategy will be followed by real-world players of the game. Say that in addition to such information, we are told the expected utilities of the players. What is our best estimate of the distribution q that generated those expected utility values? By the maxent principle, it is the distribution with maximal entropy, subject to those expectation values.

To formalize this, for simplicity assume a £nite number of players and of possible strategies for each player. To agree with the convention in other £elds, from now on we implicitly ¤ip the sign of each g_i so that the associated player i wants to minimize that function rather than maximize it. Intuitively, this ¤ipped $g_i(x)$ is the "cost" to player i when the joint-strategy is x, though we will still use the term "utility".

Then for prior knowledge that the expected utilities of the players are given by the set of values $\{\epsilon_i\}$, the maxent estimate of the associated q is given by the minimizer of

1 Throughout this paper, the integral sign is implicitly interpreted as appropriate, e.g., as Lebesgue integrals, point-sums, etc.

the Lagrangian

$$\mathcal{L}(q) \equiv \sum_i \beta_i [E_q(g_i) - \epsilon_i] - S(q) \quad (1)$$

$$= \sum_i \beta_i [\int dx \prod_j q_j(x_j) g_i(x) - \epsilon_i] - S(q) \quad (2)$$

where the subscript on the expectation value indicates that it evaluated under distribution q, and the $\{\beta_i\}$ are "inverse temperatures" implicitly set by the constraints on the expected utilities.

Solving, we £nd that the mixed strategies minimizing the Lagrangian are related to each other via

$$q_i(x_i) \propto e^{-E_{q_{(i)}}(G|x_i)} \quad (3)$$

where the overall proportionality constant for each i is set by normalization, and $G \equiv \sum_i \beta_i g_i$ [2]. In Eq. 3 the probability of player i choosing pure strategy x_i depends on the effect of that choice on the utilities of the other players. This re¤ects the fact that our prior knowledge concerns all the players equally.

If we wish to focus only on the behavior of player i, it is appropriate to modify our prior knowledge. To see how to do this, £rst consider the case of maximal prior knowledge, in which we know the actual joint-strategy of the players, and therefore all of their expected costs. For this case, trivially, the maxent principle says we should "estimate" q as that joint-strategy (it being the q with maximal entropy that is consistent with our prior knowledge). The same conclusion holds if our prior knowledge also includes the expected cost of player i.

Modify this maximal set of prior knowledge by removing from it speci£cation of player i's strategy. So our prior knowledge is the mixed strategies of all players other than i, together with player i's expected cost. We can incorporate prior knowledge of the other players' mixed strategies directly, without introducing Lagrange parameters. The resultant **maxent Lagrangian** is

$$\mathcal{L}_i(q_i) \equiv \beta_i[\epsilon_i - E(g_i)] - S_i(q_i)$$

$$= \beta_i[\epsilon_i - \int dx \prod_j q_j(x_j) g_i(x)] - S_i(q_i)$$

The £rst term in $\mathcal{L}_i$ is minimized by a perfectly rational player. The second term is minimized by a perfectly *irrational* player, i.e., by a perfectly uniform mixed strategy q_i. So β_i in the maxent Lagrangian explicitly speci£es the balance between the rational and irrational behavior of the player. In particular, for $\beta \to \infty$, by minimizing the Lagrangians we recover the Nash equilibria of the game. More formally, in that limit the set of q that simultaneously minimize the Lagrangians is the same as the set of delta functions about the Nash equilibria of the game. The same is true for Eq. 3.

Eq. 4 is solved by a set of coupled **Boltzmann distributions**:

$$q_i(x_i) \propto e^{-\beta_i E_{q_{(i)}}(g_i|x_i)}. \quad (4)$$

Following Nash, we can use Brouwer's £xed point theorem to establish that for any non-negative values $\{\beta\}$, there must exist at least one product distribution given by the product of these Boltzmann distributions (one term in the product for each i).

Eq. 3 is just a special case of Eq. 4, where all player's share the same private utility, G. (Such games are known as **team games**.) This relationship re¤ects the fact that for this case, the difference between the maxent Lagrangian and the one in Eq. 2 is independent of q_i. Due to this relationship, our guarantee of the existence of a solution to the set of maxent Lagrangians implies the existence of a solution of the form Eq. 3. Typically players will be closer to minimizing their expected cost than maximizing it. For prior knowledge consistent with such a case, the β_i are all non-negative.

For each player i de£ne

$$f_i(x, q_i(x_i)) \equiv \beta_i g_i(x) + \ln[q_i(x_i)]. \quad (5)$$

Then we can maxent Lagrangian for player i is

$$\mathcal{L}_i(q) = \int dx\, q(x) f_i(x, q_i(x_i)). \quad (6)$$

Now in a bounded rational game every player sets its strategy to minimize its Lagrangian, given the strategies of the other players. In light of Eq. 6, this means that we interpret each player in a bounded rational game as being perfectly rational for a utility that incorporates its computational cost. To do so we simply need to expand the domain of "cost functions" to include probability values as well as joint moves.

Often our prior knowledge will not consist of exact speci£cation of the expected costs of the players, even if that knowledge arises from watching the players make their moves. Such alternative kinds of prior knowledge are addressed in [17, 18]. Those references also demonstrate the extension of the formulation to allow multiple utility functions of the players, and even variable numbers of players. Also discussed there are **semi-coordinate** transformations, under which, intuitively, the moves of the agents are modi£ed to set in binding contracts.

3. Optimizing the Lagrangian

In this paper we consider two algorithms for optimizing the Lagrangian. The £rst is Brouwer updating, which under

2 The subscript $q_{(i)}$ on the expectation value indicates that it is evaluated according the distribution $\prod_{j \neq i} q_j$.

different names is perhaps the most common scheme employed in RL-based algorithms for £nding game equilibria. The second is a variant of Newton's method for directly descending the Lagrangian.

3.1. Brouwer updating

One crude way to try to £nd the q given by Eq. 4 would be an iterative process akin to the best-response scheme of game theory [8]. Given any current distribution q, in this scheme all agents i simultaneously replace their current distributions. In this replacement each agent i replaces q_i with the distribution given in Eq. 4 *based on the current* $q_{(i)}$. This scheme is the basis of the use of Brouwer's £xed point theorem to prove that a solution to Eq. 4 exists.

This scheme requires estimating a conditional expected utility for each agent at each iteration. These can be estimated via Monte-Carlo sampling across a block of time in which q is £xed. During that block the agents all repeatedly and jointly IID sample their probability distributions to generate joint moves, and the associated utility values recorded. This is exactly what is done in RL-based schemes in which each agent maintains a data-based estimate of its utility for each of its possible moves, and then chooses its actual move stochastically, by sampling a Boltzmann distribution of those estimates.

Since accurate estimates usually requires extensive sampling, we replace the G occurring in each agent i's update rule with a private utility g_i chosen to ensure that the Monte Carlo estimation of $E(g_i \mid x_i)$ both low bias (with respect to estimating $E(G \mid x_i)$ and low variance [7]. Intuitively, this bias re¤ects the alignment between the private and world utilities. At zero bias, reducing private utility necessarily reduces world utility. Variance instead re¤ects how much the utility depends of on the agent's own move rather than that the other agents. With low variance, the agents can perform the individual optimizations accurately with minimal Monte-Carlo sampling.

In this paper we concentrated on two types of private utility in addition to the team game (TG) utility. The £rst is the **Aristocrat Utility** (AU) utility. It is a correction to one of the same name previously investigated in the RL literature (see [20, 19, 21] and references therein). It is the utility, out of all those guaranteed to have zero bias, that has minimal variance:

$$g_{AU_i}(x_i, x_{(i)}) = G(x_i, x_{(i)}) - \sum_{x'_i} \frac{N_{x'_i}^{-1}}{\sum_{x''_i} N_{x''_i}^{-1}} G(x'_i, x_{(i)}) \quad (7)$$

where N_{x_i} is the number of times that agent i makes move x_i in the most recent set of Monte Carlo samples. Due to the subtracted term, AU should have lower variance than TG.

In addition we consider the **Wonderful Life Utility** (WLU), which is an approximation to AU that also has zero bias:

$$g_{WLU_i}(x_i, x_{(i)}) = G(x_i, x_{(i)}) - G(CL_i, x_{(i)}) \quad (8)$$

where the clamping value CL_i £xes agent i's move to the one to which it assigns lowest probability action [16, 18]. (Again, this is a correction to a utility of the same name previously investigated in [20, 19, 21] and references therein.)

However the utilities are estimated, one obvious problem with Brouwer updating is that there is no *a priori* reason to believe that it will converge. Implicitly acknowledging this, in practice the Monte Carlo samples are "aged", to weight older sample points less heavily than more recent points. See [20, 19, 21] for details. This modi£cation still provides no formal guarantees however. Such guarantees do obtainthough if rather than conventional "parallel" Brouwer updating, one uses "serial Brouwer updating", in which only one agent at time updates its distribution. Other alternatives are mixed serial-parallel Brouwer updating. See [16, 18] for a discussion of such technqiues. Regardless of what type of Brouwer updating one uses however, its intrinsic nature is make no use whatsoever of the many powerful techniques known for descending across functions like $\mathcal{L}(q)$ to £nd its minimum. This is not the case with the variant of Newton's method discussed below.

3.2. Constrained Newton's descent

Typically in the RL-based work employing Brouwer updating, constraints are introduced by ad hoc use of penalty functions. However an alternative is provided by the straightforward extension of the PD framework to constrained optimization. Given that the agents in a MAS are bounded rational, if we have them play a constrained team game with world utility G, their equilibrium will be the optimizer of G subject to those (potentially inexact) constraints [16, 18]. Formally, let $\{c_j(x)\}$ be the constraint functions, i.e., we seek a joint-move x such that all of the $\{c_j(x)\}$ are nowhere negative. Then the bounded rational equilibrium will minimize the Lagrangian of Eq. 2 where the world world utility is augmented with Lagrange multipliers, λ_j, for each of the

$$G(x) \rightarrow G(x) + \textstyle\sum_j \lambda_j c_j(x). \quad (9)$$

Consider a £xed set of values for the Lagrange parameters. We can minimize the associated Lagrangian using gradient descent, since the gradient can be evaluated in closed form. We can also evaluate the Hessian in closed form. This allows us to use **constrained Newton's method**. This is a variant of Newton's method in which we £rst modify the Lagrangian, and then enforce both independence of the agents, and that the search stays on the simplex of valid probabilities [18, 2]:

$$q_i(x_i) \rightarrow q_i(x_i) -$$

$$q_i(x_i)[(E[G|x_i] - E[G]) + S(q_i) + lnq_i(x_i)] \quad (10)$$

where α plays the role of a step size.
The Lagrange multipliers are then updated in the usual way, by taking the partial derivatives of the augmented Lagrangian:

$$\lambda_j \rightarrow \lambda_j + \delta E[c_j(x)] \quad (11)$$

where δ is the step size.

Just as in Brouwer updating, to evaluate the update of Eq. 10 we need to estimate conditional expected utilities of each agent. Here we use the exact same Monte Carlo-based algorithms and private utilities used in Brouwer updating.

4. Experiments

4.1. Queens Problem

The N-queens problem is not hard to solve, especially with centralized algorithms [14]. However it is a good illustration and testbed of the PD-theory approach. The goal in this problem is to locate N queens on a N-by-N chessboard such that there are no con¤icts between any of the queens, i.e., no shared rows, columns or diagonals. For the results presented here, N = 8 and each agent's move is the position of a queen on an associated row of the chessboard. Denoting agent i's making move j as $x_i(j)$, the constraints are

$$\begin{aligned} x_i(j) &\neq x_k(j) \\ x_i(j) &\neq x_{i+k}(j+k) \neq x_{i-k}(j-k) \\ x_i(j) &\neq x_{i+k}(j-k) \neq x_{i-k}(j+k) \end{aligned}$$

For 8 queens this results in 84 constraints.

For this study the step size α was set to 1.0, while the data aging rate γ was set to 0.5. The optimizations were performed at a range of £xed "temperatures" $T \equiv \beta^{-1}$. 10 Monte-Carlo samples were used for each probability and Lagrange multiplier update. We concentrated on the number of iterations to convergence, i.e., the number of probability updates times the number of Monte-Carlo samples per update, for 50 random trials of the problem. The optimization was terminated when a single Monte-Carlo sample within an iteration was found which satis£ed all of the constraints.

In other work we have used this problem to validate the predictions of PD theory about the relative merits of our three utilities [13]. Here we concentrate on comparing constrained Newton descent with an improved version of Brouwer, in which the constrained are implemented with Lagrange multipliers updated according to Eq. 11 rather than with penalty functions. So in these experiments, only the method for updating the probability distributions differed from that of constrained Newton updating. Tthe step size α was set to 1.0, while the data aging rate γ was set to 0.5. The optimizations were performed at a range of £xed temperatures and 10 Monte-Carlo samples were used for each probability and Lagrange multiplier update.

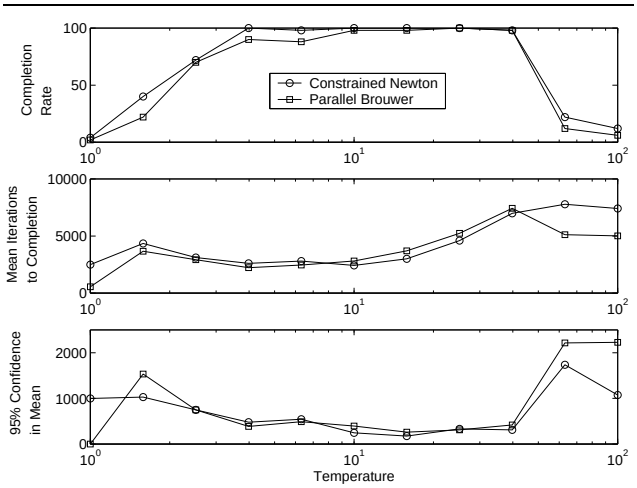

Figure 1. Comparison of RL-based and PD theory-based equilibration methods for the Queens problem. The top £gure presents the fraction of trials which successfully solved the problem, the second £gure present the mean number of iterations to that solution when it was arrived at, and the bottom £gure is the associated 95% con£dence value.

Changing the method for updating the probability distributions from the constrained Newton approach to parallel Brouwer degraded the completion rate, as indicated by Figure 1. However since we modi£ed Brouwer updating to incorporate the constraints using Lagrange parameters rather than penalty functions, the improvement was not as pronounced as it might be. In deed, the same £gure shows that over some temperatures, constrained Newton does not outperform parallel Brouwer updating.

4.2. Bar Problem

A modi£ed version of Arthur's El Farol Bar Problem has been used before to investigate the RL-based approach [19]. Here we use that same problem to compare constrained Newton updating and parallel Brouwer updating on an unconstrained problem optimization.

In this scenario there are N agents, each selecting one of seven nights to attend a bar, i.e., each agent has 7 categori-

cal moves. The world utility function is given by

$$G(\zeta) \equiv \sum_{k=1}^{7} \phi(x_k(\zeta)), \tag{12}$$

where $x_k(\zeta)$ is the total attendance on night k; and $\phi(y) \equiv y \exp(-y/c)$ with c a real-valued parameter. This choice of $\phi(.)$ means that either too few or too many agents attending a single night results in low world utility G. The parameter c is set to control the size of this congestion effect. For the results presented here, N was set to 168 with c set to 6. Note that here the goal is to maximize G, which results in appropriate sign ¤ips in the Lagrangian search.

Previous work with this problem had illustrated the advantages of using modi£ed private utilities rather than the team game with parallel Brouwer updating [20, 19]. The goal of the current work is to test the advantages of constrained Newton over parallel Brouwer updating. Results were generated over a range of £xed temperatures to highlight the difference. Figure 2 shows the comparison between constrained Newton and parallel Brouwer over the temperature range. Shown is the performance (value of G) after 1000 iterations averaged over 20 cases. The 95% con£dence intervals in all cases is less than 0.5. Constrained Newton is seen to outperform parallel Brouwer at all temperatures.

More detail is provided in Figure 3. This £gure compares the two updating schemes as the number of iterations increases, for a £xed temperature T of 0.01. The constrained Newton scheme initially improves much faster, with parallel Brouwer catching up several hundred iterations later. However constrained Newton then continues to improve. In contrast, parallel Brouwer updating oscillates, without signi£cant improvement. Also note the tighter con£dence bars on the constrained Newton results, re¤ecting higher robustness. The step size α for the results shown was £xed at 0.01 while the data aging rate γ was held at 0.1. Wonderful Life Utility (WLU) was used with a Monte-Carlo block size of 1. Other values for these parameters and other utilities were also considered and resulted in similar trends.

4.3. Bin Packing Problem

We also compared the two updating methods on an intermediate problem, in which rather than just try to solve a set of constraints (as in the Queen's problem) or try to optimize an unconstrained problem (as in the Bar problem), we try to optimize a discrete constrained problem [4]. We chose the bin packing problem for this conmparison. This problem consists of assigning N items of differing sizes into the smallest number of bins each with capacity c. For the current study instances were chosen which have a designed minimum number of bins /cite-Falk94 and were obtained from the OR-Library at

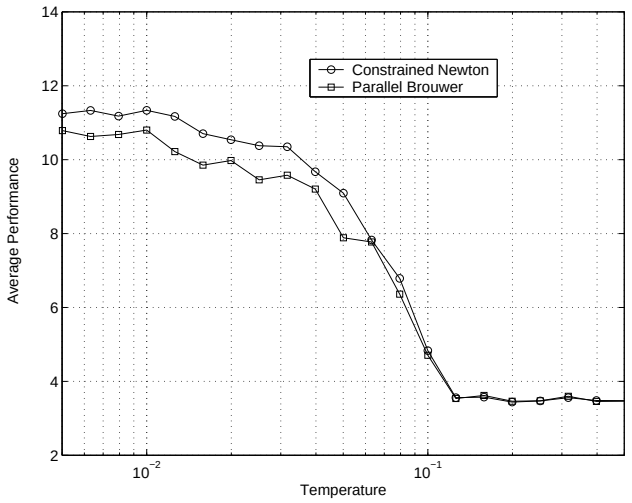

Figure 2. Comparison of RL-based and PD theory-based equilibration methods for the Bar problem. Performance is measured at the end of the run. Higher is better.

http://mscmga.ms.ic.ac.uk/info.html /citeBeas90. The instances consisted of 60 items to be packed in groups of three into 20 bins each of capacity 100. Since in general the minimum number of bins is not known, the move space of the agents was set to the number of items. So the world utility is

$$G = \begin{cases} \sum_{i=1}^{N} |x_i| & if\ x_i \leq c \\ \sum_{i=1}^{N} |x_i - c| & if\ x_i > c \end{cases} \tag{13}$$

where x_i is the total size of the items in bin i.

To mimic the use of penalty functions common with RL-based schemes, we also ran experiments in which the algorithms did not use this world utility directly for our updating algorithms (although we used it to measure performance). Instead we added an additional penalty term to G to smoothly enforce the constraint on the number of bins:

$$G_{added} = 1000(N_{filled} - N_{optimum})^2 \tag{14}$$

This provides a more meaningful comparison between constrained Newton and typical multi-agent techniques using parallel Brouwer updating.

Two different kinds of constrained Newton were explored in our experiments. The £rst used the modi£ed G in conjuntion with constrained Newton approach for updating the probabilities. The second did not add the penalty function but instead used Lagrange parameters to enforce the hard constraints on bin levels that $x_i \leq c$ for all i.

For each problem variant 20 cases were used in determining the averages and error bars. Figure 4 compares all

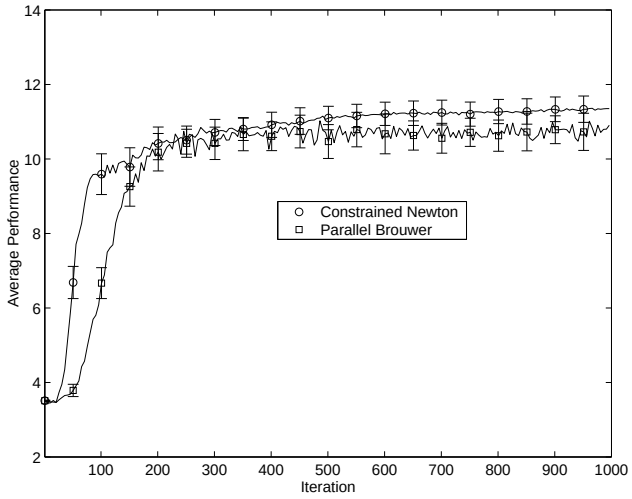

Figure 3. Comparison of RL-based and PD theory-based equilibration methods for the Bar problem. Performance is measured against time, with temperature T £xed to 0.01 for both schemes.

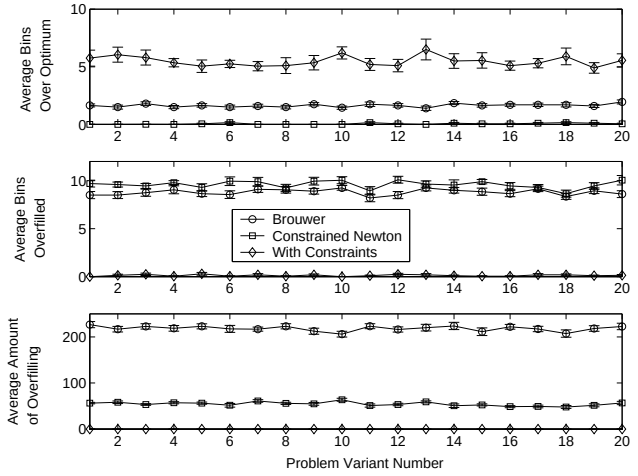

Figure 4. Comparison of RL-based and PD theory-based equilibration methods for the bin packing problem.

three schemes. The £rst plot shows the average number of bins over the optimum found by each approach. Constrained Newton with a penalty function performed best here. The second plot shows the number of bins over capacity while the third plot shows the amount of total overcapacity. Note that the unique optimum is no bins over the optimum and no bins over capacity. The results indicate that the parallel Brouwer and the constrained Newton result in similar numbers of over capacity bins, but the constrained Newton has much lower total over capacity. The constrained Newton with constraints shows even better performance, resulting in very few over capacity bins and on average only 5 bins over the optimum.

5. Conclusion

Product Distribution (PD) theory is a broad framework recently developed for analyzing and optimizing distributed systems. It can be derived as the information-theoretic extension of conventional full-rationality game theory to the case of bounded rational agents. Here we demonstrate its use for adaptive distributed control of MAS's i.e., for distributed stochastic optimization using MAS's.

In PD theory the bounded rational equilibrium of the game is the optimizer of a Lagrangian of the (probability distribution of) the joint state of the agents. When the game in question is a team game with constraints, that equilibrium optimizes the expected value of the team game utility, subject to those constraints. One common way to £nd that equilibrium is to have each agent run a Reinforcement Learning (RL) algorithm. PD theory reveals this to be a particular type of search algorithm for minimizing the Lagrangian. Typically that algorithm is quite ineff£cient. A more principled alternative is to use a variant of Newton's method to minimize the Lagrangian. Here we compare this alternative to RL-based search in three sets of computer experiments. These are the N Queen's problem and bin-packing problem from the optimization literature, and the Bar problem from the distributed RL literature. Our results con£rm that the PD-theory-based approach outperforms the RL-based scheme in all three domains.

References

[1] S. Airiau and D. H. Wolpert. Product distribution theory and semi-coordinate transformations. 2004. Submitted to AAMAS 04.

[2] N. Antoine, S. Bieniawski, I. Kroo, and D. H. Wolpert. Fleet assignment using collective intelligence. In *Proceedings of 42nd Aerospace Sciences Meeting*, 2004. AIAA-2004-0622.

[3] R. Axelrod. *The Evolution of Cooperation*. Basic Books, NY, 1984.

[4] D.P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods*. Athena Scienti£c, Belmont, MA, 1996.

[5] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley-Interscience, New York, 1991.

[6] R. H. Crites and A. G. Barto. Improving elevator performance using reinforcement learning. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neu-*

ral Information Processing Systems - 8, pages 1017–1023. MIT Press, 1996.

[7] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classi£cation (2nd ed.)*. Wiley and Sons, 2000.

[8] D. Fudenberg and D. K. Levine. *The Theory of Learning in Games*. MIT Press, Cambridge, MA, 1998.

[9] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, Cambridge, MA, 1991.

[10] J. Lawson and D. Wolpert. The design of collectives of agents to control non-markovian systems. In *of American Association of Arti£cial Intelligence Conference 2002*, 2002.

[11] C. Fan Lee and D. H. Wolpert. Product distribution theory and semi-coordinate transformations. 2004. Submitted to AAMAS 04.

[12] D. Mackay. *Information theory, inference, and learning algorithms*. Cambridge University Press, 2003.

[13] W. Macready, S. Bieniawski, and D.H. Wolpert. Adaptive multi-agent systems for constrained optimization. 2004. Submitted to AAAI 04.

[14] Rok Sosiđ and Jun Gu. A polynomial time algorithm for the N-queens problem. *SIGART Newsletter (Special Interest Group on Arti£cial Intelligence)*, 1, 1990.

[15] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

[16] D. H. Wolpert. Factoring a canonical ensemble. 2003. cond-mat/0307630.

[17] D. H. Wolpert. Bounded rational games, information theory, and statistical physics. In D. Braha and Y. Bar-Yam, editors, *Complex Engineering Systems*, 2004.

[18] D. H. Wolpert. Generalizing mean £eld theory for distributed optimization and control. 2004. Submitted.

[19] D. H. Wolpert and K. Tumer. Optimal payoff functions for members of collectives. *Advances in Complex Systems*, 4(2/3):265–279, 2001.

[20] D. H. Wolpert and K. Tumer. Collective intelligence, data routing and braess' paradox. *Journal of Arti£cial Intelligence Research*, 2002.

[21] D. H. Wolpert, K. Wheeler, and K. Tumer. General principles of learning-based multi-agent systems. In *Proceedings of the Third International Conference of Autonomous Agents*, pages 77–83, 1999.