Budgeted Personalized Incentive Approaches for Smoothing Congestion in Resource Networks

Pradeep Varakantham[1], Na Fu[1], William Yeoh[2], Shih-Fen Cheng[1], and Hoong Chuin Lau[1]

[1] School of Information Systems, Singapore Management University, Singapore
{pradeepv,nafu,sfcheng,hclau}@smu.edu.sg
[2] New Mexico State University, United States
wyeoh@cs.nmsu.edu

Abstract. Congestion occurs when there is competition for resources by selfish agents. In this paper, we are concerned with smoothing out congestion in a network of resources by using personalized well-timed incentives that are subject to budget constraints. To that end, we provide: *(i)* a mathematical formulation that computes equilibrium for the resource sharing congestion game with incentives and budget constraints; *(ii)* an integrated approach that scales to larger problems by exploiting the factored network structure and approximating the attained equilibrium; *(iii)* an iterative best response algorithm for solving the unconstrained version (no budget) of the resource sharing congestion game; and *(iv)* theoretical and empirical results (on an illustrative theme park problem) that demonstrate the usefulness of our approach.

1 Introduction

Competition for resources by autonomous agents typically leads to congestion if the agents access these resources in an uncoordinated fashion [1]. It is hence common for a network to experience congestion even when the average demand for a resource is much less than its capacity. Researchers have generally taken three approaches to address this issue. The first approach is to use the theory of *mechanism design*, where a central authority designs rules of agent interactions [2–4] by taking agent incentives into account . By designing appropriate rules, the central authority can obtain desirable goals such as maximizing social welfare. This assumes that the central authority defines and controls the rules of interaction. However, in this paper, we consider scenarios where the basic settings (rules) of the environment cannot be modified (like preferences of people going to a theme park or theme park configuration or communication protocols in a computer network).

Secondly, researchers have investigated the use of *penalties or incentives* on certain resources to discourage or encourage interactions that will lead to desirable goals. A central authority can alter the demand for certain resources by tweaking the amount of penalty or incentive for those resources. Much of the initial work in this area, especially in transportation applications [5, 6], assumes that every agent using the same resource will get the same penalty or incentive. A good example is the use of toll gates on roads. [7] and [8] provide further examples of settings where using external penalties or incentives affect the utilities involved. More recently, researchers have relaxed this assumption and implemented penalties or incentives that are probabilistic in nature [9]. For example, a public radio listener will be entered in a draw for a free iPad if he/she donates to the radio station.

Finally, Monderer and Tennenholtz have studied the problem of minimizing incentive needed to sufficiently incentivize agents to take desirable strategies (that are inputs to the problem) [10]. While there are similarities, we differ from this work in multiple ways: (1) Our focus is on finding an equilibrium strategy that is closest to the set of desired strategies given a budget; (2) we assume that the total amount of incentives that can be used must be within a given budget; and (3) our desirable strategies are specified at an aggregate level with respect to a set of agents. For instance, "no more than 10 agents can consume resource 3" as opposed to "agent 2 should take strategy 3". These differences preclude the applicability of their approach on problems with budget constraints and large number of agents.

These differences are motivated by a crowd congestion control problem in an actual theme park. Through interviews with park operators, we learnt that they can provide well-timed incentives to specific patrons through mobile devices to change their behavior and thereby ease congestion (long queues at certain attractions). Naturally, the (monetary) incentives must be within a given budget. Lastly, the park operators are interested in specifying aggregated desirable levels of congestion instead of individualized desirable strategies.

More precisely, we are interested in the problem on how best to distribute incentives among different agents at different time points so that certain resource congestion thresholds are satisfied at equilibrium and that the incentives distributed are within a given budget. We make the following contributions:

(1) We introduce a non-linear mathematical programming formulation and show how it can be linearized into a mixed-integer linear program (MILP) to compute the equilibrium for a networked congestion game with incentives and budget constraints.
(2) We exploit the factored network structure to drastically reduce the complexity of enumerating the space of agent strategies and provide an enhancement to compute approximation equilibria to scale up the MILP.
(3) We provide a scalable iterative best response algorithm to solve a version of the game without budgets while minimizing the overall incentive required.
(4) Lastly, we provide theoretical and empirical results showing that congestion is reduced at equilibrium on an illustrative theme park problem.

2 Model: NRSG

We provide the *Network Resource-Sharing Game* (NRSG), which builds on the Resource Sharing (RS) model [11] and network cost-sharing games [12]. A NRSG is similar to a network cost-sharing game except for positive rewards in NRSGs compared to positive penalties in cost-sharing games. An NRSG is the tuple:

$$\left\langle N, \mathcal{V}, \mathcal{E}, \{U_v^i\}_{i\in N, v\in\mathcal{V}}, \{s^i\}_{i\in N}, H\right\rangle$$

N $= \{1, 2, \ldots, n\}$ represents the set of agents.

$\mathcal{V}$ represents the resources and also the vertices in a graph that are connected by the edges in $\mathcal{E}$. This graph constrains certain orders of consuming resources or connections between resources.

U_v^i represents the utility obtained by agent i when it consumes one unit of resource v. For a joint strategy $\mathbf{a} = \left\langle a^1, a^2, \cdots, a^i, \cdots a^n\right\rangle$, where a^i is the action of agent i, the utility obtained by agent i is given by

$$u^i(a^1, \cdots, a^i, \cdots, a^n) = \frac{U_{a^i}^i}{\sigma_{\mathbf{a}}(a^i)} \tag{1}$$

where $\sigma_{(a)}(a^i) = \sum_{k\leq n} I(a^k = a^i)$, with $I(a^k = a^i) = 1$ if $a^k = a^i$ and 0 otherwise. While we focus on this definition of utility, our approaches can be trivially modified to work with any non-increasing function over number of agents consuming a resource.

s^i represents the starting vertex for agent i.

H represents the time horizon of the problem.

The goal in an NRSG is to find Nash equilibrium strategies for all individual agents, that is, no agent has an incentive to deviate from its strategy. It should be noted that this repeated game cannot be represented by a single-shot decision-making problem [12] because a resource selection path (of length H) cannot be considered as an independent resource. Also, it should be noted that this is not a single stage game repeated multiple times due to the following reason: (a) Utility can change over time (e.g., preferences for rollercoasters before and after lunch are different). (b) There exists a network structure on how resources can be utilised. (c) There can be domain-specific constraints (e.g., each resource/attraction can only be visited once or should visit at least 3 of my 5 preferred attractions). Note that these constraints are all linear.

A *pure strategy* for an agent i is the sequence of resources selected at each time step, and the set of all pure strategies is given by $\Pi^i = \{\pi^i \mid \pi^i = \left\langle a_1^i, a_2^i, \cdots, a_H^i\right\rangle, \forall t : a_t^i \in \mathcal{V}\}$. We do not have edges as part of the strategy, because, given a source and destination vertex, the edge is uniquely determined. A *mixed strategy* can be defined as a probability distribution over all possible pure strategies $\Delta(\Pi^i)$.To provide better understanding of the concepts, we will use the following toy example throughout the paper.

Example 1 *We consider a theme park with four attractions (resources)* $\mathcal{A} = \{A1, A2, A3, A4\}$ *that is being visited by eight patrons (agents)* $\mathcal{P} = \{P1, \cdots, P8\}$. *For ease of explanation, we assume that the service rate of each attraction* d_i *is 1 for all attractions. Let the utility for all patrons in getting serviced at an attraction is the same, which is as follows:* $\mathcal{U} = \{2, 3, 5, 7\}$. *The horizon* H *for decision making is 1 and the ideal minimum queue length* γ_i^* *desired by the theme park operator is* 2 *for all attractions* i.

3 Incentivized Budget Constrained Equilibrium

In this section, we represent the problem of finding a Nash equilibrium in an NRSG with incentives and budget constraints as an optimization problem. Traditionally, iterative best response mechanisms such as fictitious play [13] have been used to compute equilibrium solutions in congestion game models. The presence of budget constraints and desired congestion levels preclude the application of such methods.

Our approach provides personalized incentives constrained by a budget so as to achieve certain properties of resource congestion like ensuring that all queue lengths at attractions are no less than a minimum queue length or no greater than a maximum queue length. Examples of personalized incentives are freebies at an attraction if it is visited at a certain time. The key assumption in our approach is that such incentives increase the utility for individual agents.

We use the following notation to describe the optimization problem, where lower case letters such as x represent variables, bold letters such as $\mathbf{x}$ represent vectors, bold and upper case letters such as $\mathbf{X}$ represent sets of vectors:

U^i_j is the utility at resource j for agent i. U_j is the utility at resource j (if it is the same for all agents i).

$x^i_{j,t}$ is a binary variable indicating whether agent i has selected $(= 1)$ resource j at time t.

$\mathbf{x}^i$ is the strategy of agent i:

$$\begin{pmatrix} x^i_{1,1} & x^i_{1,2} & \cdots & x^i_{1,H} \\ x^i_{2,1} & x^i_{2,2} & \cdots & x^i_{2,H} \\ \cdots & \cdots & \cdots & \cdots \\ x^i_{|\mathcal{V}|,1} & x^i_{|\mathcal{V}|,2} & \cdots & x^i_{|\mathcal{V}|,H} \end{pmatrix}$$

where $|\mathcal{V}|$ and H are number of resources and horizon, respectively.

$\mathbf{x}$ is the strategy profile of all players over all resources and the entire horizon: $\mathbf{x} = (\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n)$

$\mathbf{X}^i$ is the set of all possible strategies for agent i: $\mathbf{X}^i = \{\mathbf{x}^i \mid \sum_j x^i_{j,t} \leq 1, x^i_{j,t} \in \{0,1\}, \forall t \leq H\}$

$\boldsymbol{\Delta}$ is the matrix of incentives of all agents, $\boldsymbol{\Delta} = (\boldsymbol{\Delta}^1, \ldots.\boldsymbol{\Delta}^n)$,

$$\boldsymbol{\Delta}^i = \begin{pmatrix} \delta^i_{1,1} & \delta^i_{1,2} & \cdots & \delta^i_{1,H} \\ \delta^i_{2,1} & \delta^i_{2,2} & \cdots & \delta^i_{2,H} \\ \cdots & \cdots & \cdots & \cdots \\ \delta^i_{|\mathcal{V}|,1} & \delta^i_{|\mathcal{V}|,2} & \cdots & \delta^i_{|\mathcal{V}|,H} \end{pmatrix}$$

where $\delta^i_{j,t}$ is a decision variable representing the incentive agent i obtained at resource j time t.

B is a constant representing the total amount of budget available for incentives.

m is index of a policy of an agent in the set $\mathbf{X}^i$.

$$\min_{\mathbf{X},\boldsymbol{\Delta}} \quad \Gamma \qquad \textbf{such that}$$

$$u^i_{j,t} = \frac{x^i_{j,t} \cdot U_j}{\max\{\sum_k x^k_{j,t}, 1\}} + x^i_{j,t} \cdot \delta^i_{j,t} \qquad \forall i, j, t \qquad (2)$$

$$u^{m,i}_{j,t} = \frac{x^{m,i}_{j,t} \cdot U_j}{\max\{\sum_{k \neq i} x^k_{j,t} + x^{m,i}_{j,t}, 1\}} + x^{m,i}_{j,t} \cdot \delta^i_{j,t} \qquad \forall m, i, j, t \qquad (3)$$

$$\sum_{j,t} u^i_{j,t} \geq \sum_{j,t} u^{m,i}_{j,t}, \qquad \forall m, i \qquad (4)$$

$$\sum_{i,j,t} f_j(\delta^i_{j,t}) \leq B \qquad (5)$$

$$\Gamma \geq \gamma^*_j - \sum_i x^i_{j,t} \qquad \forall j, t \qquad (6)$$

$$\sum_j x^i_{j,t} \leq 1 \qquad \forall i, t \qquad (7)$$

$$x^i_{j,t} \leq x^i_{k,t-1} \qquad \forall (k, j) \in \mathcal{E} \qquad (8)$$

$$x^i_{j,t} \in \{0, 1\} \qquad \forall i, j, t \qquad (9)$$

Fig. 1. Non-Linear Optimization Problem

$x^{m,i}_{j,t}$ is a value representing if agent i chooses resource j and at time t under agent i's m^{th} policy.

γ^*_j is a constant representing the preferred number of agents selecting resource j at any time step.

Figure 1 shows the optimization problem formulated as a non-linear mixed-integer program. For ease of explanation, we assume that all agents consuming a resource get the same utility U_j. However, the optimization problem and the proceeding linearization can be trivially adapted to have a different utility for each agent U^i_j. The key aspects of the optimization problem are:

- **No Incentive to Deviate**: Constraint 4 ensures that when all agents follow their equilibrium strategies, the overall utility (including the allocated incentive) $u^i_{j,t}$ of agent i obtained by following its equilibrium strategy is no less than the utility $u^{m,i}_{j,t}$ obtained by any other strategy m for all resources j and time steps t.
- **Budgeted Incentives**: Constraint 5 ensures that the total amount of all incentives is bounded by the budget B. One key assumption here is that the function f_j is a linear function and $\delta^{max} = \sum_j \delta^{max}_j$ is a constant computed from the following expression: $\sum_j f_j(\delta^{max}_j) = B$.
- **Desired Resource Congestion Properties**: These properties are inputs to the problem and can be constraints on the minimum or maximum number of agents consuming a resources. Constraint 6 represents the constraint for

$$u^i_{j,t} = w^i_{j,t} + \delta^i_{j,t} \quad \forall i,j,t \quad (10)$$

$$0 \le w^i_{j,t} \le x^i_{j,t} \cdot U_j \quad \forall i,j,t \quad (11)$$

$$0 \le \delta^i_{j,t} \le x^i_{j,t} \cdot \delta^{max} \quad \forall i,j,t \quad (12)$$

$$w^i_{j,t} - w^k_{j,t} \le (2 - x^i_{j,t} - x^k_{j,t}) \cdot U_j \quad \forall i,j,t,k \quad (13)$$

$$w^k_{j,t} - w^i_{j,t} \le (2 - x^i_{j,t} - x^k_{j,t}) \cdot U_j \quad \forall i,j,t,k \quad (14)$$

$$\sum_k w^k_{j,t} = U_j \cdot \alpha_{j,t} \quad \forall j,t \quad (15)$$

$$\frac{\sum_k x^k_{j,t}}{N} \le \alpha_{j,t} \le \sum_k x^k_{j,t}, \quad \forall j,t \quad (16)$$

$$\alpha_{j,t} \in \{0,1\} \quad \forall j,t \quad (17)$$

Fig. 2. Linearization Constraints for Constraint 2

the minimum number of agents γ^*_j at any resource j, where Γ represents the maximum deviation from the desired consumption.

- **Deviation Minimization**: The maximum deviation from the desired congestion properties Γ is minimized in the objective.
- **Network Structure**: Consraint 8 enforces the network structure.

While this optimization problem can model incentives accurately, there are two key issues: (1) Non-linear constraints in constraints 2 and 3 prevent scalability to larger problems, and (2) enforcing the equilibrium for each agent requires enumerating over all possible pure strategies possible for each agent, which can be exponential in the horizon and the number of resources. To address these issues, we propose three methods that increase the scalability considerably.

3.1 Linearizing the Non-Linear Constraints

As indicated earlier, the utility function can be any non-increasing piecewise constant or piecewise linear function over number of agents for us to employ similar linearization tricks on the utility function that will be explained in this section. Figure 2 shows the equivalent linear constraints to the non-linear constraints in constraint 2. The same techniques can be applied to linearize constraint 3. Using these linearized constraints, the optimization problem in Figure 1 can be represented as a mixed-integer linear program (MILP). Furthermore, for each agent, we introduce new variables $w^i_{j,t}$ and $\delta^i_{j,t}$ to represent the unincentivized utility and incentive, respectively. Thus they sum up to the overall utility $u^i_{j,t}$ (constraint 10). The intuitions for the linearization constraints are as follows:

- Constraints 11, 12: If an agent i is not consuming resource j at time t ($x^i_{j,t} = 0$), then the unincentivized utility $w^i_{j,t}$ and incentive $\delta^i_{j,t}$ are zero.
- Constraints 13, 14: If an agent i is consuming resource j at time step t ($x^i_{j,t} = 1$), then its unincentivized utility $w^i_{j,t}$ is equal to the unincentivized utility

$w^k_{j,t}$ of any other agent k that consumes the same resource at the same time ($x^k_{j,t} = 1$).
- Constraints 15-17 account for the "max" in the denominator of constraint 2.

Example 2 *At equilibrium, the number of agents at attractions A1,A2,A3 and A4 is 1, 1, 2 and 4, respectively for Example 1. That is to say, attraction A4 is more crowded than any other attractions. We can use the optimization problem above to help reduce the congestion at A4. Suppose the theme park operator provided the minimum queue length* γ_a*, which is* 2 *for all attractions* a*, and the budget* B*, which is 5. Then, the resulting equilibrium (along with the incentives in terms of utility that is same for all agents selecting the same attraction) is*

$$A1 = \{P2, P3\}, \delta^{P2}_{A1,1} = 1.33\text{;} A3 = \{P6, P7\}, \delta^{P6}_{A3,1} = 0$$
$$A2 = \{P4, P5\}, \delta^{P4}_{A2,1} = 0.83\text{;} A4 = \{P1, P8\}, \delta^{P1}_{A4,1} = 0$$

The number of agents at each attraction now is 2, which satisfies the minimum queue length, and so is the criterion for equilibrium.

3.2 Exploiting Factored Structure

We exploit the factored structure of the NRSG graph to solve the MILP faster. This efficiency comes about due to the reduction in the number of elements in the set $\mathbf{X}^i$ for every agent i and thus the number of equilibrium constraints (Constraint 4). The basic definition for X^i is given by:

$$\mathbf{X}^i = \{\mathbf{x}^i \,|\, \sum_j x^i_{j,t} \leq 1, x^i_{j,t} \in \{0,1\}, \forall t \leq H\}$$

We can update the expression to exploit the graph structure:

$$\mathbf{X}^i = \{\mathbf{x}^i \,|\, \sum_j x^i_{j,t} \leq 1, x^i_{j,t} \leq \sum_{k|(k,j)\in\mathcal{E}} x^i_{k,t}, x^i_{j,t} \in \{0,1\}, \forall t \leq H\}$$

Furthermore, if the graph is fully connected, that is, agents can consume any resource at any time step (a reasonable assumption for theme parks, where patrons can go to any attraction at any time step), then the equilibrium constraints on constraint 4 can be replaced with $\sum_j u^i_{j,t} \geq \sum_j u^{m,i}_{j,t}$ for all m, i and t. The key difference is that the new constraints sums over resources j only as opposed to over resources j and time steps t. This difference yields a reduction in number of equilibrium constraints from $|\mathcal{V}|^H$ to $|\mathcal{V}| \cdot H$.

3.3 Finding ϵ-Nash Equilibrium Solutions

The MILP representation provides the flexibility to compute an approximate Nash Equilibrium. If we modify the equilibrium constraints on Line 4 to $\epsilon + \sum_{j,t} u^i_{j,t} \geq \sum_{j,t} u^{m,i}_{j,t}$ for all m and i. Then, the resulting Nash equilibrium is an ϵ-Nash equilibrium, where each agent has an incentive of at most ϵ to deviate from the equilibrium strategy.

Example 3 *An ϵ-equilibrium strategy with $\epsilon = 0.1$ to the problem in Example 2 is given by*

$$A1 = \{P2\}, u^{P2}_{A1,1} = 2; A3 = \{P6, P7, P8\}, u^{P6}_{A3,1} = 1.66$$
$$A2 = \{P3\}, u^{P3}_{A2,1} = 3; A4 = \{P1, P4, P5\}, u^{P1}_{A4,1} = 2.33$$

This solution is not a true equilibrium because patron $P8$ can switch to attraction A4 to gain an additional 0.09 units of utility but it is an ϵ-Nash Equilibrium because the gain by changing strategy for each agent is less than $\epsilon = 0.1$.

4 Incentivized Unconstrained Equilibrium

In this section, we provide a technique for solving the problem where: (1) there is no constraint on the budget; (2) there are hard constraints on the desired consumption of resources; and (3) the goal is to minimize the total amount of incentive required to achieve the equilibrium. This problem is similar to the problem solved by the k-implementation approach [10]. However, the main d-ifferences are that we assume that agents can be individually incentivized and desired strategies are specified at an aggregate level in our work (e.g., "no more than 200 agents can consume resource 3") as opposed to specific strategies in [10] (e.g., "agent 2 should take strategy 5").

The optimization problem mentioned in Figure 1 with the linearized constraints can be easily modified to solve the unconstrained problem. Here, we provide another approach that is more scalable and based on the more typical iterative best response mechanism that also allows for mixed strategies in the equilibrium. Figure 3 shows the best response linear program for each agent i. We use the following additional variables:

$p^i_{j,t}$ is the probability of agent i choosing resource j at time t.

$\mathbf{p}^i$ is the mixed strategy of agent i similar to how $\mathbf{x}^i$ is the pure strategy of agent i in the previous MILP.

u^{*i} is the utility of best response strategy of agent i given strategies of other agents.

u^i is the utility of a strategy of agent i given the strategies of other agents.

In this approach, at each iteration and for each agent i, we fix the policies of all other agents and compute the best response strategy $\mathbf{p}^i$ that satisfies the constraint on desired resource consumption (constraint 21) and the required incentive δ^i to incentivize agent i to take that strategy. The incentives are computed in constraint 18 as the difference in utility for the best strategy u^{*i} (without the constraint on desired resource consumption) and the current utility (with the constraint on desired resource consumption) u^i. We continue this process until convergence. We do not yet have a proof for guaranteed convergence. However, if the iterative best response process converges, then we obtain an equilibrium strategy.

$$\min_{\mathbf{P^i}} \quad \delta^i \qquad \textbf{such that}$$

$$\delta^i = u^{*i} - u^i \tag{18}$$

$$u^{*i} = \max_{\mathbf{x}^i} \sum_{j,t} \frac{x^i_{j,t} \cdot U_j}{\sum_{k \neq i} p^k_{j,t} + x^i_{j,t}} \tag{19}$$

$$u^i = \sum_{j,t} \frac{p^i_{j,t} \cdot U_j}{\sum_{k \neq i} p^k_{j,t} + 1} \tag{20}$$

$$\sum_k p^k_{j,t} \leq \gamma^*_j \qquad \forall j,t \tag{21}$$

$$\sum_j p^i_{j,t} \leq 1 \qquad \forall t \tag{22}$$

$$0 \leq p^i_{j,t} \leq 1 \qquad \forall j,t \tag{23}$$

Fig. 3. Best Response Linear Program for Agent i

5 Theoretical Results

We now show that the welfare of any equilibrium solution is at least one half of the optimal social welfare in an NRSG in two steps: (1) We show that the social utility function in NRSGs is sub-modular, and (2) we show that the NRSG game is a utility system [14] and, hence, the Price of Anarchy, PoA (Ratio of social welfare for the worst equilibrium solution to the optimal social welfare) is at least $\frac{1}{2}$. Note that an NRSG with incentives is an NRSG and hence the bounds hold even when we are providing incentives.

Proposition 1. *Social utility for a joint strategy $\boldsymbol{x}$ in NRSG is sub-modular.*

Proof Sketch: Let $\mathcal{Q}$ be the set of all agents and $\mathcal{F} : 2^{\mathcal{Q}} \rightarrow \mathbb{R}$ be the social utility function. The social utility for a joint policy $\mathbf{x}$ given a set $\mathcal{Q}$ of agents is:

$$\mathcal{F}(\mathcal{Q}) = \sum_{i \in \mathcal{Q}} \frac{\sum_{j,t} x^t_{i,j} \cdot U_j}{\sum_{k \in \mathcal{Q},j,t} x^t_{k,j}}$$

For $A \subseteq B$, we show that $\mathcal{F}(A \cup \{p\}) - \mathcal{F}(A) \geq \mathcal{F}(B \cup \{p\}) - \mathcal{F}(B)$. ■

Definition 1 *A utility system represents a game where:*

- *Social and private utilities are in the same standard unit;*
- *Social utility function is sub-modular; and*
- *Private utility of an agent $\geq$ change in social utility if the agent declined to participate in the game.*

Proposition 2. *PoA for NRSGs is at least $\frac{1}{2}$.*

Proof Sketch. We show that NRSG is a utility system and hence from Vetta *et al.* [14], PoA for utility systems is at least $\frac{1}{2}$. ■

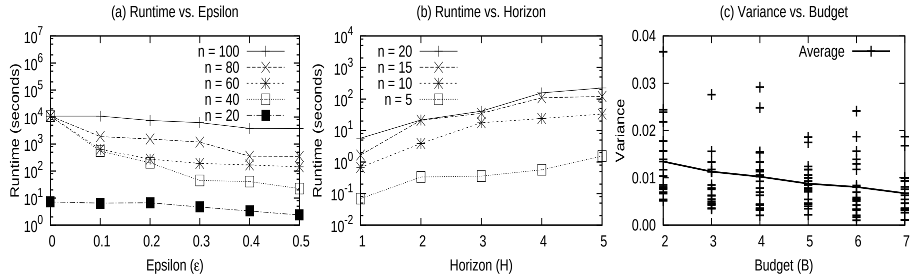

Fig. 4. Results for Incentivized Budget Constrained Problems

6 Experimental Results

We now describe our experimental results for the problem with incentives on the theme park problem described in Example 1. We performed two sets of experiments: one on problems with incentive budgets and one on problems without. For problems with incentive budgets, we only have the linearized optimization problem of Figure 1 (referred to as BC-MILP). For problems without incentive budgets and desired congestion levels, we have the iterative best response (referred to as IBR) and we compare it to a modified BC-MILP-Mod .

There are a number of different parameters that we experimented with, namely the number of agents n, the horizon H, the number of resources $|\mathcal{V}|$, budget B, desired maximum or minimum consumption of any attraction γ^* and finally the approximation parameter ϵ. If not explicitly stated, the default values for some of the parameters are as follows: $H = 1$, $|\mathcal{V}| = 4$, $B = 2$, $\gamma^* = \frac{n}{|\mathcal{V}|} \pm p \cdot n$ (depending on whether we have constraints on maximum or minimum consumption) with a default value of 10% for p and $\epsilon = 0$. Due to space constraints, we only show representative results. We conducted our experiments on a machine with a 2.40GHz CPU and 6GB of RAM.

6.1 Incentivized Budget Constrained Problems

In this set of problems, we demonstrate some of the key results with the BC-MILP. Figure 4(a) shows the runtimes, where we vary n and ϵ. We only show the runtimes for one combination of budget and γ^* parameter as the trends here are similar for other parameters. We make two observations:

(1) As the number of agents increases, the runtime increases as expected. With the increase in the number of agents, the number of variables and constraints in the MILP increases and hence the increase in runtime. However, we are able to solve problems with up to 100 agents with the BC-MILP approach. By exploiting homogeneity in agents (future work), we hope to increase this significantly.

No. of agents (γ^*)	BC-MILP-Mod runtime (sec)	BC-MILP-Mod incentive	BC-MILP-Mod social welfare	IBR runtime (sec)	IBR incentive	IBR social welfare
10 (3)	8.42	2.25	19.25	0.41	6.80	19.14
12 (3)	44.01	5.75	22.75	0.42	9.79	22.54
14 (4)	123.64	4.00	21.00	0.70	7.55	20.90
16 (4)	1105.45	6.80	23.80	0.48	10.07	23.67
18 (5)	4001.52	5.16	22.16	0.67	8.10	22.10
20 (5)	8312.05	7.50	24.50	0.54	10.26	24.42

Table 1. BC-MILP-Mod vs. IBR

(2) As ϵ increases, even by a small value, the runtime decreases significantly.[3] With the increase in ϵ, the problem becomes simpler as the MILP can return solutions with larger deviations from the Nash equilibrium. Thus, these results show the tradeoff between computation time and solution quality in terms of distance from the optimal solution.

Figure 4(b) shows the runtimes, where we vary n and H with $\epsilon = 0.3$. As expected, the runtime increases with increasing horizon. We are able to solve problems with 20 agents and horizon 5 in less than 4 minutes.
Figure 4(c) shows the variance in resource consumption of each agent (as a percentage of n), where we vary n from 10 to 30, and B from 2 to 7. The nice observation from this result is that the average variance decreases as the budget increases. In other words, *we have a better load balance, even at equilibrium strategies, when the budget B increases.*

6.2 Incentivized Unconstrained Problems

We first show the performance comparison of the IBR algorithm with the BC-MILP algorithm modified to suit the unconstrained budget and incentive minimization setting (referred as BC-MILP-Mod). The BC-MILP-Mod thus finds a solution with the least required incentive. Table 1 shows the results. IBR converged and that implies that an equilibrium solution is found in both cases. The results show that IBR is at least one order of magnitude faster than BC-MILP-Mod but finds solutions that requires much higher incentive and slightly lower social welfare, thus highlighting the tradeoff between the two approaches.

To demonstrate the scalability of the IBR algorithm, we increased the number of agents up to 500 and we were still able to solve the problem within 20 seconds. We also computed runtimes for the IBR approach while varying resources, however, there was no significant change in runtime when the number of resources was less than or equal to 10. Finally, we computed the the overall incentive required as a mapping of the ideal resource consumption (γ^*) parameter p. We varied p from 5%-15% and computed the overall incentive required with n varying between 100 to 300. As expected, the incentive decreased as the constraint on resource consumption was relaxed.

[3] Note that ϵ is an absolute error on utility and not a percentage error.

7 Conclusion

Congestion is common in resource networks that exist in domains as varied as transportation, computer networks and theme parks. In this paper, we aim to smooth out the congestion by using well-timed incentives that are constrained by a budget and are personalized to resource consumers. To that end, we provided an efficient mixed-integer linear formulation that can exploit network structure and is amenable to bounded approximation schemes. We also provide a scalable alternative to solve the incentives problem when there is no constraint on the budget and the goal is to find an equilibrium strategy with the least incentive. Our experimental results demonstrate the scalability of our approaches and on an illustrative problem, we also show that there is less congestion when the budget increases and the incentive required increases as the constraints on resource congestion become tighter.

Acknowledgement

This research/project is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

References

1. Axelrod, R.: The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration. Princeton University Press (1997)
2. Shoham, Y., Tennenholtz, M.: Social laws for artificial agent societies: Off-line design. Artificial Intelligence **73** (1995)
3. Conitzer, V., Sandholm, T.: Complexity of mechanism design. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI), Morgan Kaufmann Publishers Inc. (2002) 103–110
4. Nisan, N., Ronen, A.: Algorithmic mechanism design. In: Proceedings of the ACM Symposium on Theory of Computing (STOC). (1999) 129–140
5. Bergendorff, P., Hearn, D., Ramana, M.: Congestion toll pricing of traffic networks. Lecture Notes in Economics and Mathematical Systems (1997) 51–71
6. Hearn, D., Ramana, M.: Solving congestion toll pricing models. In Marcotte, P., Nguyen, S., eds.: Equilibrium and Advanced Transportation Modeling. Kluwer Academic Publishers (1997) 109–124
7. Dybvig, P., Spatt, C.: Adoption externalities as public goods. Journal of Public Economics **20** (1983) 231–247
8. Segal, I.: Contracting with externalities. The Quarterly Journal of Economics **2** (1999) 337–388
9. Leyton-Brown, K., Porter, R., Prabhakar, B., Shoham, Y., Venkataraman, S.: Incentive mechanisms for smoothing out a focused demand for network resources. Computer Communications **26**(3) (2003) 237–250
10. Monderer, D., Tennenholtz, M.: K-implementation. Journal of Artificial Intelligence Research **21** (2004) 37–62

11. Law, L.M., Huang, J., Liu, M.: Price of anarchy for congestion games in cognitive radio networks. IEEE Transactions on Wireless Communications **PP**(99) (2012) 1–10
12. Chen, H.L., Roughgarden, T., Valiant, G.: Designing network protocols for good equilibria. SIAM Journal on Computing **39**(5) (2010) 1799–1832
13. Monderer, D., Shapley, L.S.: Fictitious play property for games with identical interests. Journal of Economic Theory **68**(1) (1996) 258—265
14. Vetta, A.: Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions. In: Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS). (2002) 416–425