Conference paper

Koh, Andrew (2008) *Nash Equilibria, collusion in games and the coevolutionary particle swarm algorithm.* In: Filipic, B. and Silc, J., (eds.) Bioinspired Optimization Methods and their Applications. Third International Conference on Bioinspired Optimization Methods and their Applications (BIOMA 2008), 13-14 October 2008, Jozef Stefan Institute, Ljublana, Slovenia. Jozef Stefan Institute , Slovenia, pp. 19-28.

NASH EQUILIBRIA, COLLUSION IN GAMES AND THE COEVOLUTIONARY PARTICLE SWARM ALGORITHM

Andrew Koh
Institute for Transport Studies, University of Leeds, Leeds LS2 9JT, United Kingdom

a.koh@its.leeds.ac.uk

Abstract: In recent work, we presented a deterministic algorithm to investigate collusion between players in a game where the players' payoff functions are subject to a variational inequality describing the equilibrium of a transportation system. In investigating the potential for collusion between players, the diagonalization algorithm returned a local optimum. In this paper, we apply a coevolutionary particle swarm optimization (PSO) algorithm developed in earlier research in an attempt to return the global maximum. A numerical experiment is used to verify the performance of the algorithm in overcoming local optimum.

Key words: Nash Equilibrium, Equilibrium Problems with Equilibrium Constraints, Diagonalisation, Bilevel Variational Inequality

1. INTRODUCTION

This paper discusses the determination of Nash Equilibrium (NE) subject to variational inequality constraints. This is an emerging area of research within transportation network analysis and has particular significance in an environment of deregulated infrastructure provision. When these private sector participants compete in a market in simultaneously and non-cooperatively deciding their strategic variables to offer to consumers, the concept of the Cournot-Nash game can be used to model the equilibrium variables offered by each firm to the consumers. However when the firms (players) are constrained by another variational inequality which describes the equilibrium condition of a given system, we obtain an Equilibrium Problem with Equilibrium Constraints (EPEC) [11]. Note that we use the term "firm" and "player" interchangeably throughout this paper as the subject matter transcends game theory and market structures.

In this paper, we focus on this problem but consider implicit collusion and the potential for these lead to equilibrium strategies that are beneficial to both parties. In doing so, we make use of the concept of local NE introduced in [16] to distinguish that from a global NE.

This paper is organized as follows. In the next section, we introduce the EPEC and define associated concepts of Nash equilibrium taken from [16]. A currently available deterministic optimization algorithm available is also given. In Section 3, we will outline the essential concepts in a transportation network setting which is our application area. We then describe the coevolutionary particle swarm optimization algorithm (CoPSONash) [7] in Section 4 to this problem and by way of a numerical example in Section 5 show that the CoPSONash obtains a global optimum for this problem overcoming the local NE trap. Section 6 summarizes.

2. EPECS AND NASH EQUILIBRIA

An Equilibrium Problem with Equilibrium Constraints (EPEC) [11, 12] seeks to find equilibrium points in a game when the constraints describe a variational inequality that defines an overall system equilibrium. For the purposes of this paper, the system equilibrium is the equilibrium in route choices in a (highway) transportation setting. The study of EPECs has only just recently surfaced as an important research area within mathematics and optimization theory with significant practical applications e.g. in deregulated electricity markets (e.g. [5]). To do so, we formally define the various facets of NE and give an outline of an available deterministic local search algorithm that can be used to seek such equilibrium points.

2.1 Nash Equilibrium

In a single shot normal form game with N players indexed by $i, j \in \{1,2,...,N\}, i \neq j$ each player can play a strategy $u_i \in U_i$ which all players are assumed to announce simultaneously. Let $u = (u_1, u_2, ..., u_N) \in U$ be the combined strategy space of all players in this game and let $\psi_i(u)$ be some payoff or profit function to $i \in \{1,2,...,N\}$ player if the combined strategy is played.

The combined strategy tuple $u^* = (u_1^*, u_2^*, ..., u_N^*) \in U$ is a Nash Equilibrium (NE) for the game if the following holds

$$\psi_i(u_i^*, u_j^*) \geq \psi_i(u_i, u_j^*) \; \forall u_i \in U_i, \forall i, j \in \{1, 2, ...N\}, i \neq j \quad (1)$$

Equation (1) states that a NE is attained when no player has an incentive to deviate from his current strategy. This is the concept as introduced by Nash in [13]. We now consider two refinements from [16].

2.2 Local Nash Equilibrium and NE Trap

LOCAL NASH EQUILIBRIUM [16, DEFINITION 2, P306]:

The combined strategy tuple u^* (as above) is a local NE if: $\exists \omega > 0$ such that $\forall i, \forall u_i \in B_i^{\omega}(u_i^*)$, $B_i^{\omega}(\hat{u}_i) = \{u_i \in U_i \,\|\, \|u_i - \hat{u}_i\| < \omega\}$ the following holds:

$$\psi_i(u_i^*, u_j^*) \geq \psi_i(u_i, u_j^*) \; \forall u_i \in U_i, \forall i, j \in \{1, 2, ...N\}, i \neq j \quad (2)$$

Each NE that satisfies the above definition given in (1) clearly also satisfies the definition of local NE given by (2). But the converse is not true generally. In essence this means that a strategy is only a Nash equilibrium within some ball radius in strategy space; but it may not be necessarily so globally. Hence we define the notion of a local NE trap.

LOCAL NE TRAP [16, DEFINITION 3, P306]:

The combined strategy u^* is a local NE trap if: It is a local NE as defined above in (2) *and* in addition: $\exists i$ such that $\exists u_i^{**} \in U_i$ the following holds:

$$\psi_i(u_i^{**}, u_j^*) \geq \psi_i(u_i^*, u_j^*) \; \forall u_i \in U_i, \forall i, j \in \{1, 2, ...N\}, i \neq j \quad (3)$$

2.3 A Deterministic Algorithm for EPECs

While novel deterministic algorithms have been recently have been recently proposed for EPECs [12], their use has not been widely adopted. Instead, we describe a simple and well known deterministic (gradient based) solution method for this problem known as the diagonalization algorithm.

This algorithm decomposes the problem into a series of interrelated optimization problems and subsequently solving each individually. This is a fixed point iteration (Gauss-Jacobi/Gauss-Siedel) algorithm. Harker [15] popularized this algorithm for solving the classical Cournot Nash game from economics and EPECs arising in the deregulated electricity markets were solved using this way in [5]. The algorithm is presented as follows:

Step 1: Set iteration counter $k = 0$. Select a convergence tolerance parameter, ε(ε>0). Choose a strategy for each player. Let the initial strategy set be denoted $u^k = (u_1^k, u_2^k, ..., u_N^k)$ Set $k = k+1$ and go to Step 2,

Step 2: For the ith player $i \in \{1,2,...,N\}$, solve the following optimization problem: $u_i^{k+1} = \max_{u_i \in U_i} \psi_i(u_i, u_j^k)\ \forall i, j \in \{1,2,...N\}, i \neq j$

Step 3: If $\sum_{i=1}^{N} \left| u_i^{k+1} - u_i^k \right| \leq \varepsilon$ terminate, else set $k = k+1$ and return to Step 2.

The problem with the above algorithm is that it could terminate at the local NE and fall prey to the local NE trap. This occurrence is crucially dependent on the starting point in Step 1 of the algorithm i.e. in the choice of the initial strategy of each player. [16] in fact shows that iterative search algorithms such as the diagonalization algorithm presented above cannot differentiate the real NE from a local NE trap.

3. PROBLEM DEFINITION

We now describe in more detail the optimization problem at Step 2 of the above algorithm. This is to find an optimal equilibrium toll (level of road user charge per vehicle) for each firm who separately controls[1] a predefined link on the traffic network under consideration. We can consider this problem to be a Cournot Nash game between these individual players. Therefore the equilibrium decision variables can be determined using the concepts of NE as defined above.

[1] "Control" is used as a short hand to imply that the firm has been awarded some franchise for collection of the tolls.

3.1 Notation

Define:
A : the set of directed links in a traffic network,
B : the set of links which have their tolls optimised, $B \subset A$
K : the set of origin destination (O-D) pairs in the network
$\mathbf{v}$: the vector of link flows $\mathbf{v} = [v_a], a \in A$
$\boldsymbol{\tau}$: the vector of link tolls $\boldsymbol{\tau} = [\tau_a], a \in B$
$\mathbf{c(v)}$: the vector of monotonically non decreasing travel costs as a function of link flows $\mathbf{c} = [c_a(v_a)], a \in A$
$\mathbf{d}$: the continuous and monotonically decreasing demand function for each O-D pair as a function of the generalized travel cost between OD pair k alone, $\mathbf{d} = [d_k], k \in K$ and
$\mathbf{D}^{-1}$: the inverse demand function
$\boldsymbol{\Omega}$: feasible region of flow vectors, (defined by a linear equation system of flow conservation constraints).

3.2 Optimization Problem for Individual Players

If we assume that each player controls only a single link in the network then, the optimization problem for each player, with the objective being maximizing the revenue[2] is as follows:

$$\max_{\tau_i} \psi_i(\boldsymbol{\tau}) = v_i(\boldsymbol{\tau})\tau_i, \forall i \in N \tag{4}$$

Where v_i is obtained by solving the variational inequality (see [2],[14])

$$\mathbf{c}\left(\mathbf{v}^*, \boldsymbol{\tau}\right)^T \cdot \left(\mathbf{v} - \mathbf{v}^*\right) - \mathbf{D}^{-1}\left(\mathbf{d}^*, \boldsymbol{\tau}\right)^T \cdot \left(\mathbf{d} - \mathbf{d}^*\right) \geq 0 \ \text{ for } \forall\left(\mathbf{v}, \mathbf{d}\right) \in \boldsymbol{\Omega} \tag{5}$$

Note that the vector of link flows can only be obtained by solving the variational inequality given by (5). This variational inequality represents Wardrop's user equilibrium condition where user equilibrium in route choice is attained when no user can decrease his travel costs by unilaterally changing routes [15]. If we further assume that the travel cost of any link in the network is dependent only on flow on the link itself, the above variational inequality in (5), for given $\boldsymbol{\tau}$, can be solved by means of a

[2] Costs of toll collection could easily be accounted for in the model but ignored here for simplicity.

convex optimization problem [1] (this is the known as the Traffic Assignment Problem). Details can be found in [8].

In practice, if this problem is to be solved by the diagonalization algorithm outlined in Section 2.3, then the optimization problem in Step 2 at each iteration involves using a gradient based optimization method (here we use the Cutting Constraint Algorithm (CCA) [10]) to solve the problem for each player using the objective given in (4). Note that the variational inequality constraint (5) implicitly handled within the CCA[3]. In doing so, the decision variables for other players in the game are held fixed.

3.3 Considering Collusion

In [8], for a model with 2 players, we introduced a collusion parameter α ($0 \le \alpha \le 1$) to model the possibility of players colluding. With α we can consider a more general form of the expression for the payoff function given in (4). In this case, at each iteration, the optimization problem to be solved at Step 2 by the diagonalization algorithm becomes that as given by (6)

$$\underset{\tau_i}{\text{Max}}\ \psi_i(\boldsymbol{\tau}) = v_i(\boldsymbol{\tau})\tau_i + \alpha(v_j(\boldsymbol{\tau})\tau_j), \forall i, j \in N, i \neq j \tag{6}$$

Subject to (5)

Equation (6) reduces to (4) when α = 0; similarly when α = 1, the objective of each player is to maximize the total toll revenue of both players. Note that he can only change tolls on the link under his control and still continues to take the other player's toll as an exogenous input in his optimization process. Thus whilst the ith player is in the process of optimizing his revenue, he takes into account a proportion represented by α of the jth player's toll revenue. In doing so, via the diagonalization algorithm, he is "signaling" to his competitor that he wishes to "collude" to maximize the total revenue, not just his own. Thus α represents some intuitive level of collusion between players. We also assume throughout that players reciprocate the actions of the competitors and would do likewise.

The interesting question therefore is whether it is possible to "perturb" the diagonalization algorithm at each iterate with the intent of simulating this implicit signaling to each other an alternative objective and in so doing collude to raise overall revenues.

[3] Details of the implementation of CCA for each individual player's optimisation problem can be found in [9].

4. COEVOLUTIONARY PSO ALGORITHM

The PSO algorithm [6] forms the basis of the coevolutionary PSO algorithm (CoPSONash) we developed in [7] as an alternative to the diagonalization algorithm. For a game with N players, each sub-population represents particles comprising the strategic decision variables (tolls, $\boldsymbol{\tau}$) for each of these players. Each of these players' strategies are encoded in a swarm with H particles. The steps of CoPSONash are as follows:

Step 1: Generate N subpopulations (1 for each player) of particles ($\boldsymbol{\tau}$) and velocities randomly
Step 2: Randomly select one particle from each player as its Nash strategy
Step 3: Evaluate each subpopulation by solving a Traffic Assignment Problem (5) for fixed $\boldsymbol{\tau}$ (and compute (6)) given the Nash strategy (from Step 2). Identify the global best (gb) particle from each subpopulation and set this as the new Nash strategy for each player.
Repeat
Step 4: Synchronization: Announce Nash strategy to all players.
  For each subpopulation i = 1 to N do
  Step 5: Re-evaluate i^{th} subpopulation given the announced Nash strategy of all other players and obtain personal and global bests.
    For each particle j = 1 to H do
    Step 6: Fly each particle through problem space using PSO velocity update equation (see e.g. [3] for details). Update j^{th} particle position using PSO position update equation (see e.g. [3] for details).
    Step 7: Solve (5) with new $\boldsymbol{\tau}$ (and compute (6)) to obtain objective.
    Step 8: Update personal bests (pb) if fitter than previous pb. Update gb if fitter than fittest discovered by i^{th} subpopulation so far.
    Next j
  Step 9: Identify gb particle and set this as the Nash strategy for i^{th} subpopulation.
  Next i
Until Termination Criteria is met (e.g. after a given maximum number of iterations)

During initialization, particle positions and associated velocities are randomly generated. One strategy from each subpopulation is randomly selected as the initial Nash strategy for that player. Each subpopulation is evaluated separately, by solving the traffic assignment problem (5), to determine the objective for each player (6), given the Nash strategy of the other players. Hence, the personal bests and global best particle for each player can be identified. With all subpopulations evaluated, each player's global best particle is announced to the whole group during the key synchronization phase of the algorithm. This synchronization intrinsically embodies coevolution as the fitness of a particular strategy is dependent on that of others in the game. This process continues for a maximum number of user defined iterations. The aim of the algorithm is to evolve a swarm of strategy vectors for each player robust to the strategies of others which would then satisfy the Nash equilibrium condition as defined by (1). For more details of the algorithm, the reader is referred to [7].

Our numerical example in the next section shows that by harnessing the global search capabilities of PSO, the pitfall of falling into the local NE trap can be obviated. In addition, any variant of PSO (see e.g. [3] for a full review) can be employed in the search process in Steps 6 to 9 of the above.

5. NUMERICAL EXAMPLE

The numerical example used here is a network shown in *Figure 1* and taken from [9]. The link parameters and the elastic demand functions can be found therein. This network has 18 one way links with 6 O-D pairs (1 to 5, 1 to 7, 5 to 1, 5 to 7, 7 to 1 and 7 to 5). Links 7 and 10, shown as dashed lines in *Figure 1*, are the only links in the network subject to tolls. The maximum allowable toll for each link was set to be 1000 seconds.

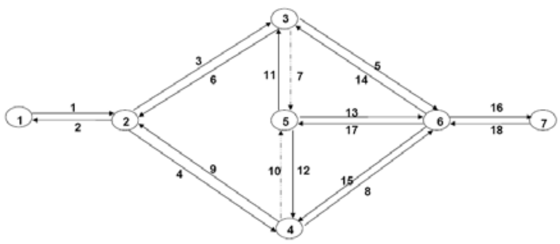

Figure 1. Traffic Network for Numerical Example

Our numerical example focuses only on the case when α, the collusion parameter, for each player, equals 1. In this case, the solution of the EPEC should be the similar to assuming that 1 player has control over both links 7 and 10 in the network. Therefore this represents the maximum total possible revenue arising from tolls on these two links and serves as a benchmark in terms of the total revenues received. The results of the diagonalization algorithm (with CCA) are contrasted with that obtained by CPSONash and the benchmark and are shown in *Table 1*. This table also shows the best solution from 20 runs of the CPSONash algorithm (with 200 iterations per run).

Table 1. Comparing Diagonalisation with CPSONash ($\alpha = 1$)

	Solution when 1 player controls both links		Diagonalisation Algorithm with CCA		CPSONash	
	Toll (secs)	Revenue (secs)	Toll (secs)	Revenue (secs)	Toll (secs)	Revenue (secs)
Link 7	713.19	280,255	189.76	116,186	713.17	282,291
Link 10	709.53	266,465	186.58	111,216	709.55	264,427
Total		546,720		227,402		546,719

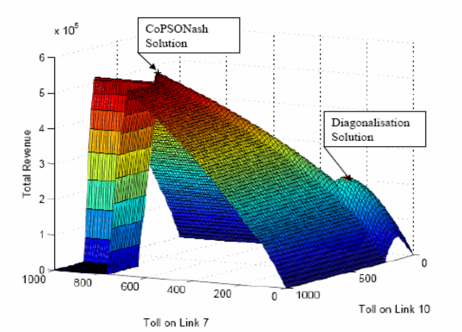

Figure 2. Total Revenue Surface as Tolls on Link 7 and Link 10 vary

Figure 2 plots the revenue surface (i.e. the revenue obtained by simultaneously varying tolls on Links 7 and 10) and illustrates that the solution obtained by diagonalization is in fact a local optimum of this function. These results, as reported in *Table 1*, are highlighted in *Figure 2*. From *Figure 2*, it is evident that the diagonalization algorithm fell into a local NE trap defined by (3) while the CoPSONash converged to the global optimum of this problem.

6. CONCLUSIONS AND FURTHER RESEARCH

In this paper, we applied a coevolutionary particle swarm algorithm to overcome a local NE trap defined by [16]. Our particular application showed that it is possible for players to collude by taking into account a modified objective function. Using a coevolutionary PSO algorithm, we demonstrated that it was possible to bypass the NE trap, attain the global optimum and thereby increase toll revenues. A limitation of this work is the problem considered here is a game with only two players and a single strategy variable (tolls). Nevertheless there appears to be potential in applying the proposed algorithm to more difficult EPECs with increased dimension in both strategies and players. Further work on this topic is ongoing.

Acknowledgements: This work is supported by UK EPSRC.

REFERENCES

[1] Beckmann M, McGuire C B, Winsten C B,(1956) *Studies in the Economics of Transportation*, Yale University Press, New Haven, Connecticut.
[2] Dafermos S C,(1980) Traffic Equilibrium and Variational Inequalities, *Transportation Science*,14(1):42-54.
[3] Engelbrecht A P,(2002) *Computational Intelligence: An Introduction*, John Wiley, New York.
[4] Harker P T,(1984)A variational inequality approach for the determination of Oligopolistic Market Equilibrium, *Mathematical Programming*, 30(1):105-111.
[5] Hobbs B F, Metzler C B, Pang J S,(2000) Strategic gaming analysis for electric power networks: An MPEC approach, *IEEE Transactions on Power Systems*, 15(2):638-645.
[6] Kennedy J, Eberhart R C,(1995) Particle swarm optimization, *Proceedings of IEEE International Conference on Neural Networks*, IEEE Press, Piscataway, N J,:1942-1948.
[7] Koh A, (submitted) Coevolutionary Particle Swarm Algorithm for Cournot-Nash Games, Manuscript submitted to Middendorf M, Engelbrecht A P (eds) *Applied Swarm Intelligence*, Springer.
[8] Koh A, Shepherd S P, (2008),Tolling, Capacity Selection and Equilibrium Problems with Equilibrium Constraints, *Paper for the 3rd Kuhmo-Nectar Conference*, Free University of Amsterdam, July 3 – 4, 2008.
[9] Koh A, Shepherd S P, Sumalee A S (2007) Second Best Toll and Capacity Optimisation in Networks, *Proceedings of the Universities Transport Studies Group*, Harrogate, January.
[10]Lawphongpanich, S Hearn, D W (2004) An MPEC Approach to Second-Best Toll Pricing, *Mathematical Programming Part B*, 101(1):33-55.
[11]Mordukhovich B S,(2005) Optimization and equilibrium problems with equilibrium constraints, *Omega-International Journal of Management Science*,33(5):379-384.
[12]Mordukhovich, B S,(2006) *Variational Analysis and Generalized Differentiation. II: Applications*, Grundlehren Series in Fundamental Principles of Mathematical Sciences, Vol 331, Springer-Verlag, Berlin.
[13]Nash J,(1950) Equilibrium points in N-person games, *Proceedings of the National Academy of Science*, 36(1):48-49.
[14]Smith M J,(1979) The existence, uniqueness and stability of traffic equilibria, *Transportation Research Part B*, 13(4): 295-304.
[15]Wardrop J G,(1952) Some theoretical aspects of road traffic research, *Proceedings of the Institute of Civil Engineers(Transport)* Part II, 1(2), 325-378.
[16]You S, Baldick R,(2004) Hybrid coevolutionary programming for Nash equilibrium search in games with local optima, IEEE *Transactions on Evolutionary Computation*,8(4):305-315.