Paper:
Koh, ATM (2011) *An Evolutionary Algorithm based on Nash Dominance for Equilibrium Problems with Equilibrium Constraints.* Applied Soft Computing, (in press).

http://dx.doi.org/10.1016/j.asoc.2011.08.056

An Evolutionary Algorithm based on Nash Dominance for Equilibrium Problems with Equilibrium Constraints

Andrew Koh

Institute for Transport Studies, 36-40 University Road, University of Leeds, Leeds, LS2 9JT, United Kingdom

---

Abstract

This paper introduces an evolutionary algorithm for the solution of a class of hierarchical ("leader-follower") games known as Equilibrium Problems with Equilibrium Constraints (EPECs). In one manifestation of such games, players at the upper level who assume the role of leaders, are assumed to act non cooperatively to maximise individual payoffs. At the same time, each leader's payoffs are constrained not only by their competitor's actions but also by the behaviour of the followers at the lower level which manifests in the form of an equilibrium constraint. By a redefinition of the selection criteria used in evolutionary methods, the paper demonstrates that the solution for such games can be found via a simple modification to a standard evolutionary multiobjective algorithm. We give a proposed algorithm (NDEMO) and illustrate it with numerical examples drawn from both the transportation systems management literature and the electricity generation industry underlying the applicability of NDEMO in multidisciplinary contexts.

Keywords: Nash Equilibrium, Equilibrium Problems with Equilibrium Constraints, Transportation Systems Management, Electricity Markets

---

1. Introduction

A major trend in the provision of transportation services and facilities has been deregulation coupled with the private sector playing a larger role. When it occurs in highway [60] or transit [62], entities providing such services face competition from others with similar offerings. It is of interest to regulators

Email address: a.koh@its.leeds.ac.uk (Andrew Koh)

to understand how such organizations make decisions on their service levels in this deregulated environment.

In this environment, the service levels provided are an outcome of a non-cooperative Nash game [39] amongst the players. However in transportation, this game possesses a feature that distinguishes it from the classic Nash game: The players' actions are constrained by a condition defining equilibrium in the transportation system [12]. Users of the transportation network make their route choice decisions by choosing routes that are the lowest cost according to Wardrop's Equilibrium Principle [58] and their route choice is parameterized in the decision variables of these firms. Therefore this is a hierarchical (i.e. leader-follower) game with the firms as leaders at the upper level engaged in a Nash game and travelers as followers at the lower level routing according to an equilibrium condition. Thus the terms "firms", "leaders" and "players" are synonymous in this context.

The game just described is an instance of a broader class of Equilibrium Problems with Equilibrium Constraints (or EPECs) ([35],[36]). EPECs have emerged as an area of research ([2],[10],[57]) in mathematics applicable to transportation systems management and other disciplines ([22],[36]). This paper focuses on the determination of equilibrium values of the strategic variables for each profit maximizing leader when in competition with others.

This paper is an extension of the earlier work by the present author [29] but has been extended in two key areas. Firstly on the theoretical aspect, we strengthen the theoretical justification of the proposed algorithm. On the practical aspect, we demonstrate the applicability of our algorithm to the examples of EPECs that arise not only within transportation systems management but also those arising in the electricity generation industry to demonstrate that our proposed algorithm is indeed applicable in multidisciplinary contexts.

The rest of this paper is organized as follows. In the next section we outline the literature of the leader follower game paradigm that forms the basis of this research. Section 3 subsequently focuses on the notions associated with the non-cooperative Nash game underlying the behaviour of the leaders in the EPEC. Section 4 reviews both the deterministic (i.e. gradient based) and evolutionary approaches for computing NE. Section 5 elucidates the Nash Domination criteria developed in [33] and provides an algorithm. Section 6 presents numerical examples of the solution of EPECs utilizing the concept of Nash Domination. Section 7 concludes the paper with a summary and directions for further research.

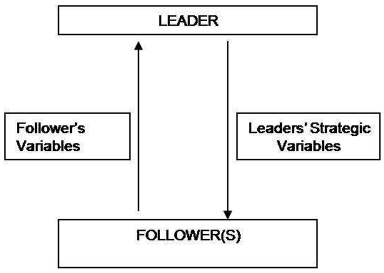

Figure 1: Stackleberg Game - Single Leader (MPEC)

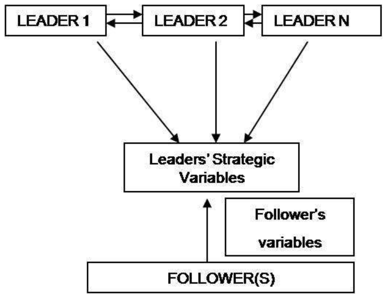

Figure 2: Multiple Leader Follower Game (EPEC)

2. Leader Follower Games

Figure 1 gives a pictorial representation of what has come to be known as the Stackelberg game [56]. It is a model of the market structure whereby a single leader is able to gain increased profits by anticipating the reactions of the rest of the market participants (known as the "followers"). In the field of mathematics, the Stackelberg game is referred to as a Mathematical Program with Equilibrium Constraints (MPEC) and has been investigated in detail by a number of researchers (see [31],[41]). The characteristic unifying feature of MPECs is that in addition to general constraints, there exists a constraint specifying equilibrium in some parametric system. The key point to note is that the followers are assumed to take the single leader's decision variables as exogenous when optimizing their individual objectives [32].

This equilibrium constraint is also present in the case of a Multiple Leader Follower Game shown in Figure 2. Though both models possess in common the hierarchical feature, the key difference between Figure 1 and Figure 2 is that multiple leaders are present in the latter and these leaders are assumed to play a game amongst themselves. We thus seek the equilibrium points of the game played by these upper level leaders. Hence as an extension of the MPEC, Figure 2 illustrates the more general class of Equilibrium Problem with Equilibrium Constraints (EPECs).

In this multi-leader generalization of the Stackelberg game articulated e.g. in [35], researchers have conjectured that there could be two possible potential behaviours of the leaders at the upper level [42]. At one extreme, leaders could act cooperatively and this results in a multiobjective problem subject to an equilibrium constraint at the lower level (MOEPEC) [61]. At the other extreme, the leaders can act non-cooperatively and play a Nash game amongst themselves resulting in a Non Cooperative EPEC (NCEPEC). In one of the numerical examples, we will revisit the distinction between the MOEPEC and NCEPEC. For the main part of this paper though we concentrate exclusively on the situation in which leaders act non-cooperatively with the objective of maximizing personal gain.

Casting our present work within the broader research context, the existence of the binding equilibrium condition distinguishes the games we describe herein from standard Nash Games. In particular [12] have pointed out that the NCEPEC is a special case of a Generalised Nash Equilibrium Problem as described in (e.g. [19],[24],[52]).

3. Nash Equilibrium

Much of the game theory literature deals with games that are either zero sum where victory or gains for one player is exactly balanced by the defeat or losses for the other (as in games such as checkers [3]) or where the actions of players are constrained to be in a discrete set (such as the binary options of confess/do not confess in games like the Prisoner's Dilemma [49]). However the solution algorithms proposed for these are generally not applicable to NCEPECs. In such games, the payoffs to the players are continuous and the strategic decision variables are subsets of the real line (as described in Chapter 6 of [59]).

Consider the leaders' problem in the NCEPEC. This is a single shot normal form game with a set N of players indexed by $i \in \{1, 2, ..., n\}$ and each player can play a strategy $s_i \in S_i$ which all players are assumed to announce simultaneously. $S = \prod_{i=1}^{n} S_i$ is the collective action space for all players. It is convenient to denote s_{-i} as the combined strategies of all players in the game excluding that of player i i.e. $s_{-i} \equiv (s_1, ..., s_{(i-1)}, s_{(i+1)}, ..., s_n)$. So with a slight abuse of notation, we have that $s \equiv (s_i, s_{-i})$. We emphasize that the notation (s_i, s_{-i}) *does not* imply that the components of s are reordered such that s_i becomes the first block. We refer to s as a strategy profile of all players in the game. Let $U_i(s)$ be the payoff to player $i, i \in N$ if s is played.

Definition 1. *[39] A combined strategy profile $s^* = (s_1^*, s_2^*, ..., s_n^*) \in S$ is a Nash Equilibrium (NE) for the game if :*

$$U_i(s_i^*, s_{-i}^*) \geq U_i(s_i, s_{-i}^*) \ \forall s_i \in S_i \ , \forall i \in N \tag{1}$$

Definition 1 emphasises the fact that at a NE no player can benefit (increase individual payoffs) by unilaterally deviating from its current strategy. Hence each player is doing the best taking into account what the competitors are doing [16]. The NE problem is the determination of strategies that satisfy Equation 1.

4. Computation of Nash Equilibrium

4.1. Deterministic Approaches

In a game, the optimal strategy for a player is governed by the best response function. If $U_i(s)$ is continuously differentiable, then the best response function for player i is given by $dU_i(s_i, s_{-i})/ds_i = 0$ ([16], [59]). The NE is the intersections of these best response functions for all players which amounts to finding solutions to n simultaneous equations i.e. solving $dU_i(s_i, s_{-i})/ds_i = 0, \forall i \in \{1, 2, ..., n\}$ ([7],[59]).

While useful for providing insights into the behaviour of players, the analytical method is not feasible for realistic problems and even less so for NCEPECs due to the binding equilibrium condition. Thus the practical approach for finding NE is by using variants of fixed point iteration (e.g. non-linear Gauss-Siedel) ([25],[57]) or by formulating it as a Complementarity Problem [26]. Applications of these methods are found in (e.g. [18], [30]). Convergence of these algorithms rely on the payoff functions being continuously differentiable and possessing diagonally dominant Jacobians ([16], Theorem 4.1, pp. 280). However, if the payoff functions of the players are not concave, there may exist NE that satisfy Equation 1 locally but not globally. This is known as a "local NE trap" ([54], Definition 3, pp.306). There is thus a parallel with the literature on multi-modal function optimization where the potential for multiple optima cannot be ignored. Thus apart from their differentiability requirements, another drawback of deterministic approaches is that they can fall prey to the local NE trap, an occurrence crucially dependent on the starting point used in these algorithms. For details of these and other deterministic methods, see ([13],[14],[38]).

4.2. Evolutionary Methods

Due to the proven ability of evolutionary algorithms to deal with non-smooth and non-differentiable functions as evidenced by their reported success in escaping local optima and potentially local NE traps, evolutionary counterparts of deterministic fixed point iteration methods were proposed in ([48],[50] and [53]).

In particular, the motivation of the work reported in [53] was to employ the NE paradigm as an alternative to multiobjective optimization. In this work the authors provided an example which suggested that the NE point is on the Pareto Frontier which was generated by a standard evolutionary multiobjective optimization (EMO) algorithm. It was stated in [53] that the EMO required much more computing resources to generate the Pareto Frontier and the Nash Genetic Algorithm that these authors proposed would be robust for finding at least one solution and is hence useful as an alternative. However there is a need to exercise caution. Though there exists games where the NE is also Pareto Optimal, this is generally not the case. Since the NE fundamentally assumes non-cooperative behaviour between players with each maximizing personal rather than collective interests, it is clearly possible that one player can be made better off without making another worse off and thus in this case the NE is not Pareto Optimal. This fact has been demonstrated in [21] and will also be shown in a numerical example to be presented later in Section 6.

A parallel research strand has been the exploitation of co-evolution since it was first demonstrated [44] for tackling multi-dimensional function optimization. Several sub populations (one representing each problem dimension) are evolved simultaneously to avoid premature convergence and to widen the search of the problem space. Ideas from co-evolution have been exported into algorithms designed for the detection of NE; here each sub population encodes the strategies of individual players ([6],[43],[47]). However doubts have been cast on the performance of co-evolutionary methods. In [54], the co-evolutionary algorithm had to be hybridized with local search techniques to enable successful detection of NE. [27] developed a co-evolutionary particle swarm optimization method which attempted to detect the NE by learning the best response functions of the players. Instead of using the co-evolutionary paradigm of previous works, a novel idea exploiting the concept of Nash Dominance was proposed [33] to find NE as discussed in Section 5.

5. Nash Domination

5.1. Theoretical Foundations

At their most abstract level, evolutionary multi-objective (EMO) algorithms ([4],[11]) apply stochastic operators to a parent population with the aim of evolving a fitter child population to solve vector valued optimization problems. Subsequently, in the selection phase, a comparison is made between a chromosome x from the parent population and a chromosome y from the child population on the basis of fitness and the weaker of the two is discarded. This is entirely consistent with the principle of survival of the fittest. Given that one of the objectives of EMO is to identify the entire Pareto front [11], fitness is assigned based on Pareto Domination (PD): x Pareto Dominates y if x is strictly no worse off than y in all objectives *and* x is better than y in at least one objective ([11], Definition 2.5, pp. 28).

[33] define a concept analogous to PD called Nash Domination for the NE problem. A chromosome in this context represents the strategies of all N players concatenated into a row vector i.e. a strategy profile. Then instead of using PD to compare two chromosomes i.e. two strategy profiles, Nash Domination operates by counting the number of players that can benefit if each player switches strategies *in turn*. The *fewer* the number of players that can profit from unilaterally deviating from one profile compared to the other, the closer the former is to a NE following Definition 1.

Consider two strategy profiles $\{x, y\} \in S, (x \equiv (x_1, ..., x_n), y \equiv (y_1, ..., y_n))$ and introduce an operator $k : S \times S \to N$ associating the cardinality of a set defined by 2:

$$\{i \in \{1, ..., n\} \,| U_i(y_i, x_{-i}) \geq U_i(x), y_i \neq x_i\} \tag{2}$$

The set thus defined by (2) comprises the players that would potentially benefit by playing y_i when everyone else plays x_{-i}. The total number of players in this set is given by $k(x, y)$. A similar interpretation applies, *mutatis mutandis*, for $k(y, x)$. The procedure is summarized in Algorithm 1. Note that in order to evaluate $k(x, y)$ and $k(y, x)$, the payoff to each player, individually, from deviating has to be computed. Following this procedure outlined in Algorithm 1, one of the following outcomes must be true: ([33], Remark 4, pp. 365)

1. $k(x, y) < k(y, x) \to x$ Nash Dominates y or
2. $k(y, x) < k(x, y) \to y$ Nash Dominates x or
3. $k(x, y) = k(y, x) \to x$ and y are Nash Non Dominated (NND) with respect to each other.

Lemma 1. *All Nash Non Dominated (NND) chromosomes are NE.*

Algorithm 1 Nash Domination Comparison

```
Initialize k(x, y) = 0, k(y, x) = 0
for i = 1 to n do
  if U_i(y_i, x_{-i}) ≥ U_i(x) then
    k(x, y) = k(x, y) + 1
  else if U_i(x_i, y_{-i}) ≥ U_i(y) then
    k(y, x) = k(y, x) + 1
  end if
end for
```

PROOF. See [33], Proposition 9, pp. 366. □

The theoretical basis of the Nash Domination Comparison procedure proposed in [33] and outlined in Algorithm 1 is in fact founded on the Nikaido Isoda (NI) function. This function as given in Eqn. 3 is a mathematical tool that plays a key role in the study of NE problems [5],[12],[20],[24]. Consider again two strategy profiles $\{x, y\} \in S$, then the interpretation of $\Psi(x, y)$ is as follows: each summand shows the increase in payoff a player will receive by unilaterally deviating and playing a strategy y while other players play according to x.

$$\Psi(x, y) = \sum_{i=1}^{n} [U_i(y_i, x_{-i}) - U_i(x)] \tag{3}$$

The interpretation of $\Psi(y, x)$ is analogous: each summand in this case is the increase in payoff a player will receive by unilaterally deviating and playing a strategy x while other players play according to y. $\Psi(x, y)$ is everywhere non-positive for all feasible y when x is a NE profile, a result that follows directly from Definition 1 because at a NE no player can increase their payoff by unilaterally deviating. Thus the NI function plays the role of a "merit function" measuring the proximity of a strategy to NE. In other words, the closer $\Psi(x, y)$ when compared to $\Psi(y, x)$, is to 0, the closer x is to a NE compared to y. Without explicitly using the NI function, the Nash Domination procedure suggested in [33] achieves the same goal by instead *counting* the number of players that can profitably deviate.

5.2. The NDEMO Algorithm

Based on Lemma 1, we can find the NE by checking for Nash Dominance when comparing chromosomes. This replaces the usual Pareto Dominance check when using a standard EMO algorithm. Hence instead of locating the Pareto Front, we collect its analogue: the Non Nash Dominated Front to which the population converges. A proposed Nash Domination Evolutionary Multiplayer Optimization (NDEMO) algorithm is given in Algorithm 2. NDEMO is based on the method of [51] which relies on Differential Evolution (DE) [46]. By modification of this selection criteria, any other EMO algorithm (see [4] or [11] for alternatives) can be used.

NDEMO operates as shown in Algorithm 2. The user specifies the maximum number of iterations Max_{it}, the population size NP, the convergence tolerance, $\epsilon(> 0)$, control parameters required in DE, namely Mutation Factor F and Probability of Crossover CR [46] and a procedure to compute payoffs. Initial parent strategy profiles $\mathcal{P}$ are generated randomly. A hypothetical example of such a profile is shown in Table 1. Each chromosome is a vector in D dimensions with D being equal to the number of strategy variables per player multiplied by the number of players (assuming that every player has the same number of strategy variables). In the hypothetical example given in Table 1 since there are two player with two strategic variables each, we have that D is 4.

Table 1: Example of chromosome encoding of a strategy profile in a hypothetical game with 2 players and 2 strategic variables per player

	Player 1's strategies		Player 2's strategies	
Variable	1	2	1	2
Value	2.75	20.14	0.126	30.133

Child strategy profiles $\mathcal{C}$ are created by applying the DE operators via the stochastic combination of randomly chosen parents as discussed in [46]. Algorithm 3 uses the "DE/rand/1/bin" [46] strategy which means that the child vector is formed by adding 1 difference vector to a randomly selected population member and then undergoes binary crossover. Whilst the mutation procedure (line 11 of Algorithm 3) is performed on the entire vector, it must be noted that crossover (line 13 of Algorithm 3) operates dimension-wise. In lines 14 of Algorithm 3, $rand(0, 1)$ is a pseudo random number between 0 and 1 and $intr(1, D)$ is a pseudo random integer between 1 and D. Finally to

Algorithm 2 Nash Domination Evolutionary Multiplayer Optimization

Input: NP, Max_{it}, ϵ, DE Control Parameters, payoff functions
$it \leftarrow 0$
Randomly initialize a population of NP parent strategy profiles $\mathcal{P}$
Evaluate payoffs to players with $\mathcal{P}$
while $it < Max_{it}$ or $\mathcal{P}$ not converged **do**
  for $j = 1$ to NP **do**
    use Algorithm 3 to create child strategy profiles vector y
    $\mathcal{C}_j^{it} \leftarrow y$
  end for
  Evaluate payoffs to players with $\mathcal{C}$
  $\mathcal{T} \leftarrow \emptyset$
  for $j = 1$ to NP **do**
    $x \leftarrow \mathcal{P}_j^{it}$
    $y \leftarrow \mathcal{C}_j^{it}$
    use Algorithm 1 to carry out Pairwise Nash Domination Comparison between x and y to determine $k(x, y)$ and $k(y, x)$
    if $k(x, y) < k(y, x)$ **then**
      discard y
      $\mathcal{T} \leftarrow x$
    else if $k(y, x) < k(x, y)$ **then**
      discard x
      $\mathcal{T} \leftarrow y$
    else
      $\mathcal{T} \leftarrow x$
      $\mathcal{T} \leftarrow y$
    end if
  end for
  if $|\mathcal{T}| > NP$ **then**
    Randomly trim $\mathcal{T}$ until NP remain
  end if
  Randomly choose a chromosome m from $\mathcal{T}$
  Compute Euclidean norm between m and every member in $\mathcal{T}$
  if maximum of norm $\leq \epsilon$ **then**
    Terminate
  else
    $\mathcal{P}^{(it+1)} \leftarrow \mathcal{T}$
    $it \leftarrow it + 1$
  end if
end while
Output: Nash Non Dominated Solutions

enforce bound constraints, we also utilise the method suggested in [45] (line 16 of Algorithm 3) so that if the child vector produced violates the bound constraints, it is reset to a point half way between its pre-mutation value and the bound violated.

Algorithm 3 Creating a child vector via Differential Evolution

1: Input: Current Population $\mathcal{P}$
2: Input: Mutation Factor F, Probability of Crossover CR
3: Input: Lower Bounds LB_d and Upper Bounds UB_d in each dimension d
4: Randomly choose 3 integers: $r1, r2, r3$ between 1 and NP such that:
5: $r1 \neq j, r2 \neq r1 \neq j$ and $r3 \neq r2 \neq r1 \neq j$,
6: $x \leftarrow \mathcal{P}_j$
7: $a \leftarrow \mathcal{P}_{r1}$
8: $b \leftarrow \mathcal{P}_{r2}$
9: $c \leftarrow \mathcal{P}_{r3}$
10: Mutation: Produce a mutant vector z via a stochastic combination of donor vectors
11: $z \leftarrow a + F(b - c)$
12: **for** $d = 1$ to D **do**
13: Crossover
14: $y_d \leftarrow \begin{cases} z_d & \text{if } rand(0,1) < CR \vee d = intr(1, D) \\ x_d & \text{otherwise} \end{cases}$
15: Enforce Bound Constraints
16: $y_d \leftarrow \begin{cases} (x_d + LB_d)/2 \text{ if } y_d < LB_d \\ (x_d + UB_d)/2 \text{ if } y_d > UB_d \\ y_d \text{ otherwise} \end{cases}$
17: **end for**
18: Output: child vector y

At each generation, parent and child strategy profiles are compared one by one pairwise, following the Nash Domination Comparison procedure of Algorithm 1. Those chromosomes that are NND are stored in a temporary population $\mathcal{T}$. However, this means that the size of $\mathcal{T}$, shown in line 27 of Algorithm 2 as $|\mathcal{T}|$, may exceed NP. If this happens, we randomly trim $\mathcal{T}$ so that there will always be only NP parents for the next generation (lines 27 to 29 of Algorithm 2). To check convergence to the NE, randomly select a chromosome m and compute the Euclidean norm between m and every member of $\mathcal{T}$ (lines 30 to 31 of Algorithm 2). If the maximum distance is less than ϵ, the population is judged to have converged to a NE and the algorithm can terminate. Otherwise the counter is increased and the process is repeated.

6. Numerical Examples

In this section six numerical examples occurring in three multidisciplinary contexts are provided to demonstrate the applicability of NDEMO to solving realistic problems. Table 2 gives the parameters used for the numerical experiments. Note that though we allowed for a maximum of 400 iterations, all the examples required less than this to meet the specified termination tolerance ϵ of 0.0001. All numerical experiments were conducted using MATLAB^{TM} 7.8 running on a 32 bit Windows^{TM} XP machine with 4 GB of RAM.

Table 2: Parameters used in the NDEMO for all Numerical Experiments

Control Parameter	
Mutation Factor F	0.45
Probability of Crossover CR	0.35
Population Size NP	50
Maximum Number of Iterations Max_{it}	400
Termination Tolerance ϵ	0.0001

6.1. Examples from Production of Homogeneous Product

The first example presented arises when firms compete in the production of a homogeneous product. The purpose of this example is three fold. Firstly, we demonstrate that NDEMO successfully converges to previously reported results for games without any equilibrium constraint (i.e. when the game is not hierarchical in nature) and thus show that NDEMO can be applied to standard Nash games. Secondly, we use this example to demonstrate an instance of an MOEPEC when the players are assumed *to cooperate*. Finally, we wish to compare the solution of the MOEPEC with the NCEPEC for the purpose of emphasising the distinction between a non cooperative Equilibrium and a Cooperative Equilibrium.

The player dependent parameters (ω_i, λ_i and θ_i) shown in Table 3 are found in [34]. These will be the parameters that we will use for the 3 case studies for this example.

Table 3: Production Cost Function Parameter Specification for Players

Firm i	ω_i	λ_i	θ_i
1	10	5	1.2
2	8	5	1.1
3	6	5	1.0
4	4	5	0.9
5	2	5	0.8

6.1.1. Case 1: Example 1 as a Cournot Nash Game

Here we consider the situation in which the firms engage in a Cournot-Nash game amongst themselves. Because of the absence of a hierarchical structure, this game is neither a NCEPEC nor a MOEPEC. However its inclusion serves to demonstrate that the proposed algorithm is able to detect the NE and replicate the reported results in [18] and [30] where deterministic methods were proposed. In this setting, each firm maximises individual profits from the sale of the homogeneous good (given as the difference between revenues and production costs) as given by 4.

$$
\begin{aligned}
U_i(q) = \; & \underbrace{P(q)q_i}_{\text{Revenues}} \; \underbrace{-\,\omega_i q_i + (\frac{\theta_i}{\theta_i+1})\lambda_i^{\frac{-1}{\theta_i}} x_i^{\frac{\theta_i+1}{\theta_i}}}_{\text{Production Costs}} \\
\text{where } & P(q) = 5000^{\frac{1}{1.1}}(\sum_{i=1}^{5} q_i)^{-(\frac{1}{1.1})}
\end{aligned}
\qquad (4)
$$

However the price ($P(q)$) and hence individual firm revenues is dependent not only on their on individual production levels but also on that of their competitors. Using the parameters of NDEMO as mentioned earlier, Table 4 reports results obtained from applying NDEMO to this problem and also compares it against the results published in the literature.

6.1.2. Case 2: Example 1 as a MOEPEC

Let us now consider the example from [37] where a separate problem to be described , using the same parameters as in Case 1, is cast as a MOEPEC. In [37] it was assumed that Firms 1 and 5 become the leaders in the game and cooperate to maximize individual profits again following 4 while the followers (firms 2,3 and 4) play a Nash game amongst themselves. This therefore gives

Table 4: Example 1 Case 1 - Comparison of the Results of NDEMO with results published in literature

Firm	1	2	3	4	5
[18]	36.9318	41.8175	43.706	42.659	39.1800
[30]	36.9325	41.8182	43.7066	42.6593	39.179
Results from NDEMO:					
Mean	36.9368	41.8198	43.7092	42.6612	39.18
Standard Deviation	0.0012	0.0006	0.0008	0.0005	0.0005

rise to a MOEPEC [1]. Given the production levels of the leaders and treating these as exogenous, the followers seek to individually maximize their profits using 4. Assuming that the payoff functions are continuously differentiable (a condition easily verifiable for this example) the first order conditions for a profit maximum for each of the followers are defined by 5:

$$\text{CP} \left\{ \begin{array}{l} f_i = \frac{\partial U_i}{\partial q_i} \geq 0 \\ \frac{\partial U_i}{\partial q_i} q_i = 0 \\ q_i \geq 0 \end{array} \right\} i \in \{2, 3, 4\} \tag{5}$$

It is easy to see that 5 in fact defines a Complementarity Problem (CP) [13],[26],[30] which when written in generic form is to find $q \in \mathbb{R}^n$ where $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ such that:

$$\begin{array}{l} f(q) \geq 0 \\ qf(q) = 0 \\ q \geq 0 \end{array} \tag{6}$$

As the leaders (firms 1 and 5) cooperatively maximise their profits, the actions of the followers leads to 5 which is imposed as an implicit nonlinear constraint on the leaders' actions. The resulting MOEPEC can be written as a vector optimization problem (with T denoting the transpose) in 7.

[1]This is in fact a MultiObjective Equilibrium Problem with Complementarity Constraints (MOEPCC) but the MOEPCC is a special case of the MOEPEC and the distinction does not affect our ensuing discussions.

$$
\begin{aligned}
&\max_{q_1,q_5}[U_1(q_1,q_{-1}),U_5(q_5,q_{-5})]^T \\
&\text{subject to} \\
&q_1,q_5 \geq 0 \\
&\{q_2,q_3,q_4\} \rightarrow \text{sol CP}
\end{aligned} \tag{7}
$$

In 7 "sol CP" emphasises that the production levels of the followers is the solution of the (nonlinear) CP given by 5. The Multiobjective Self Adaptive Differential Evolution (MOSADE) [23] algorithm was used to generate the Pareto Front corresponding to 7 and the resulting front is shown in Figure 3. In doing so, we integrated within MOSADE, the PATH Solver from [9] to solve the CP (i.e. 5) for each vector of the production levels of the leaders. On Figure 3, the two points marked with a $\star$ correspond to the two solutions reported in [37] (see Table 5) which were obtained using a deterministic non smooth method. If negotiations between the leaders were allowed under prevailing anti-trust legislation, we conjecture that this Pareto Front would play a key role in these negotiations.

Table 5: The two solutions reported in [37] and indicated on Figure 3 with $\star$

	Solution 1	Solution 2
Profit of Leader 1/Firm 1	840.86	978.89
Profit of Leader 2/Firm 5	485.63	410.97

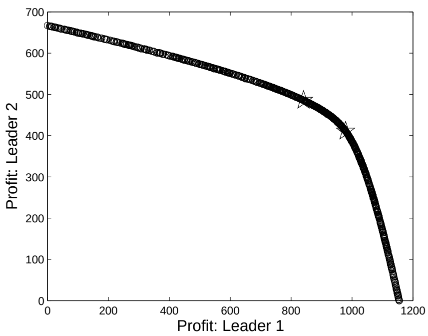

Figure 3: Example 1-Case 2 Pareto Front for MOEPEC

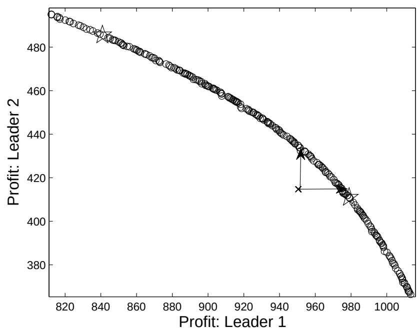

Figure 4: Example 1-Case 3 The NCEPEC solution ($\times$) is *not* on the Pareto Front

6.1.3. Case 3: Example 1 as a NCEPEC

Using the same parameters as in the previous two cases, assume that these same leaders, firms 1 and 5 *do not cooperate* as they were assumed to do in Case 2 but instead play a Nash game amongst themselves. The optimization problem facing each leader individually is given in 8 and 9 respectively. As the leaders (firms 1 and 5) individually maximise their profits, the production levels of the followers leads to the complementarity problem which is imposed as an implicit nonlinear constraint on the leaders' actions.

$$\text{Player 1 /Leader 1} \begin{cases} \max\limits_{q_1} U_1(q_1, q_{-1}) \\ \text{subject to} \\ q_1 \geq 0 \\ q_5 = \bar{q_5} \\ \{q_2, q_3, q_4\} \rightarrow \text{sol CP} \end{cases} \tag{8}$$

$$\text{Player 5 /Leader 2} \begin{cases} \max\limits_{q_5} U_5(q_5, q_{-5}) \\ \text{subject to} \\ q_1 \geq 0 \\ q_1 = \bar{q_1} \\ \{q_2, q_3, q_4\} \rightarrow \text{sol CP} \end{cases} \tag{9}$$

Critically compared to the MOEPEC, in the NCEPEC, when optimising their individual profits, each leader searches for the best response to the other firm's production level. Integrating NDEMO with the PATH Solver [9] to resolve the CP as before, we apply NDEMO to solve the resulting NCEPEC.

The result produced by the NDEMO algorithm is given in Table 6 and this solution (in profit space) is marked with $\times$ on Figure 4. As illustrated in Figure 4, the non cooperative outcome is not Pareto Optimal. It is obvious that any one of the leaders can be made better off (i.e. increase individual profits) *without* making the other worse off. For example, holding the profit from Leader 1 fixed at $\times$ of 950.56, one can move upwards (in the direction of the arrow) towards the Pareto Front and hence increase the profit of Leader 2 without reducing the profit accruing to Leader 1. This outcome highlights the key difference between the MOEPEC and the NCEPEC and the proposed NDEMO algorithm is designed for the latter. Our finding is similar to that concluded in [21] [2] who also found that the result obtained by

[2] Figure 2 in [21] is analogous to Figure 4 in this paper.

the Nash Genetic Algorithm [53] *lies inside* the Pareto Front generated by a conventional EMO algorithm when optimizing a problem arising in the steel forging industry. In our example, the primary reason that the Nash point lies inside the Pareto Front is attributable to the assumption of non-cooperative behaviour between the leaders.

Table 6: Example 1 Case 3 - Production levels and Profits for Leaders in NCEPEC

Leader 1		Leader 2	
Production	Profit	Production	Profit
97.70	950.56	42.14	414.72

6.2. Examples from Private Sector Participation in the Operation of Toll Roads

The next three examples presented are typical of situations when private profit maximizing firms compete with one another in the operation of toll roads. The private firms, acting as leaders, set their strategic decision variables and the followers (who are in effect the highway users) optimize their route choice according to Wardrop's Equilibrium Condition [58]. We seek therefore to compute the Nash Equilibrium strategic variables of these games [3].

We define the notation for a mathematical statement of the problem:

A: the set of directed links in a traffic network,

B: the set of links which are subject to tolls $B \subset A$,

Q: the set of origin destination (O-D) pairs in the network,

v: the vector of link flows $v = [v_a]\ a \in A$,

τ: the vector of link tolls with $\tau_a = 0, \forall a \notin B$,

$c(v)$: the vector of monotonically non decreasing travel costs as a function of link flows on that link only,

$c = [c_a(v_a)]\ a \in A$,

μ: the vector of generalized travel cost for each OD pair $\mu = [\mu_q],\ q \in Q$,

δ: the continuous and monotonically decreasing demand function for each O-D pair as a function of the generalized travel cost between OD pair q alone, $\delta = [\delta_q],\ q \in Q$,

[3]We measure tolls in units of seconds and revenues/profits in seconds following conventions in the transportation planning literature [40].

δ^{-1}: the inverse demand function giving the highway travel cost as a function of the demands and

Ω: feasible region of flow vectors, defined by a linear equation system of flow conservation constraints.

For simplicity suppose that each player is able to set tolls only on a single link in the network then each seeks to maximize its individual revenue (payoff) as given in 10

$$\underset{\tau_i}{\text{Max}} \quad U_i(\tau) = v_i(\tau)\tau_i, \forall i \in 1, 2...n \tag{10}$$

Where v_i is obtained by solving the variational inequality (11) i.e.

$$c(v^*, \tau)^T (v - v^*) - \delta^{-1}(\delta^*)^T(\delta - \delta^*) \geq 0, \quad \forall (v, \delta) \in \Omega \tag{11}$$

Hence for a specified toll vector τ, the solution of the Variational Inequality defined by 11 results in a vector of link flows and demands ($v^*, \delta^* \in \Omega$) that satisfies Wardrop's Equilibrium principle [58] of route choice (see [8, 55]). It is well known that the variational inequality in 11 can be solved by means of a standard traffic assignment algorithm once the vector of tolls have been input [8].

6.2.1. Example 2

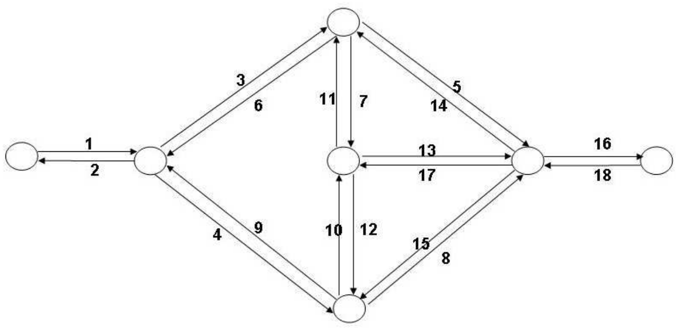

Figure 5: Highway Network with 18 directed links from [28] for Examples 1 and 2 (links labeled are road numbers)

This example is taken from [28]. Let roads number 7 and 10 shown in Figure 5 be the only toll roads operated by two independent players. This

Table 7: Example 2 - NCEPEC Tolls (seconds)

Firm	Road	NDEMO	[27]	[28]
1	7	141.36	141.36	141.37
2	10	138.28	138.29	138.29

example was solved as a complementarity problem in [28] and via a Coevolutionary Particle Swarm Algorithm in [27]. We employed NDEMO with the parameters given in Table 2 and NDEMO took 12 minutes to converge to the tolls shown in Table 7 which agrees with previous results. The standard deviation of the variables at convergence were both less than 0.0001.

6.2.2. Example 3

Next we consider the situation when, in addition to roads 7 and 10 being tolled as in Example 2, another player maximizes payoffs by charging tolls on road 17. The results are reported in Table 8. The standard deviation of the variables at convergence were both less than 0.0001. Although NDEMO again successfully converged to the NE (as verified by solving it as a complementarity problem following the method described in [28]), this time NDEMO took 19 minutes to meet the same convergence criteria. Thus with one additional player, the time taken has increased by 40% over the 2 player case. The increase in computing time stems from the domination checking procedure of Algorithm 1 combined with the hierarchical nature of the game. This arises from the need to solve a traffic assignment problem for each unilateral deviation (so as to obtain $k(x, y)$ and $k(y, x)$).

Table 8: Example 3 - NCEPEC Tolls (seconds)

Firm	Road	NDEMO	Method of [28]
1	7	140.94	140.96
2	10	137.51	137.56
3	17	711.25	712.88

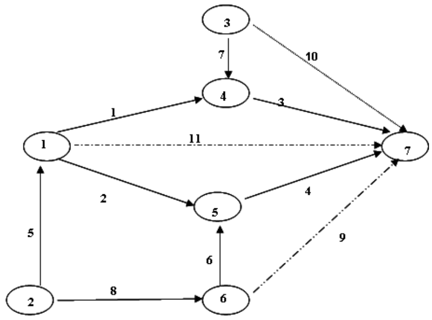

Figure 6: Highway Network with 11 directed links from [60] for Example 3 (links labeled are road numbers), Dash lines indicate links (9,10 and 11) which are subject to Tolls and Capacity Expansion

6.2.3. Example 4

In this example we consider the 11 link network from [60] where each player has two strategic variables. In this case, beyond collecting the toll revenues, each player also has to finance capacity expansion of the network link operated. In addition to the notation introduced earlier, we redefine B to be the set of links which are subject to both tolls and capacity enhancements, $B \subset A$. Further let β represent the vector of link capacity enhancements with $\beta_a = 0, \forall a \notin B$.

The payoff to player $i, i \in N$ is the difference between the toll revenue obtained by collecting tolls from traffic using the link and the amortized cost of providing the capacity enhancements, $I(\beta_i)$. Thus α is a parameter that transfers the costs of the project into unit period costs. Mathematically, the resulting choice of the strategic variables for each player may be represented by the optimization problem in 12:

$$\underset{\tau_i,\beta_i}{\text{Max}} \;\; U_i(\tau, \beta) = v_i(\tau, \beta)\tau_i - \alpha I(\beta_i), \forall i \in N \tag{12}$$

Where v_i is obtained by solving the variational inequality representing Wardrop's User Equilibrium Condition (13)

$$c(v^*, \tau, \beta)^T(v - v^*) - \delta^{-1}(\delta^*)^T(\delta - \delta^*) \geq 0, \quad \forall (v, \delta) \in \Omega \tag{13}$$

The three links (numbered 9,10 and 11) subject to tolls and capacity enhancements are shown as dashed lines in Figure 6. Details of the link parameters and the functional forms of the travel demand relationships can be found in [60] where this was solved via a heuristic gradient based procedure.

The results in [60] are compared with those produced by NDEMO in Table 9. At convergence, the standard deviation of the population of toll and capacity enhancement variables are all less than 0.0001. Although NDEMO took about 18 minutes to converge to the specified tolerance (which is similar to the time taken in Example 2) this finding suggests that increasing the number of strategic variables per player did not have a significant effect on the performance of the algorithm.

Table 9: Example 4 - NCEPEC Tolls and Capacities

		NDEMO			[60]		
Firm	Link	Toll (secs)	Capacity (vehicles)	Profit (secs/hr)	Toll (secs)	Capacity (vehicles)	Profit (secs/hr)
1	9	4.53	151.74	302.04	4.52	151.60	301.43
2	10	4.76	193.01	417.63	4.76	193.04	417.14
3	11	2.97	61.29	25.98	2.97	61.88	25.92

We provide plots of the mean and standard deviation of the population at each iteration to illustrate the convergence of NDEMO for this problem. The left and right panes of Figures 7 to 9 show the means and standard deviation of the population of toll variables for each player over the 300 iterations of the algorithm. Similar plots are provided in Figures 10 to 12 of the population of the capacity variables for each player.

6.3. Examples from the Electricity Generation Industry

In deregulated electricity markets, electricity generating companies ("GENCOs") submit bids of the quantities of electricity they propose to supply to meet the market demand to maximize their profits. These bids are then cleared by the Independent System Operator (ISO). However the price and individual profits are not only dependent on their individual bids but also that of their competitors and the prices are not known until the market clearing is performed by the ISO [1],[15], [17]. This results in a NCEPEC and NDEMO can be applied to such "pool based bidding" games. The last two examples in this paper illustrate the performance of NDEMO in this context.

6.3.1. Example 5: Two Bus Model with 2 players

Two players, indexed by i, $i \in \{1, 2\}$, submit bids to generate electricity to maximize individual profits. As shown in 14, this is given by the difference between revenues and costs from production of q_i units of electricity.

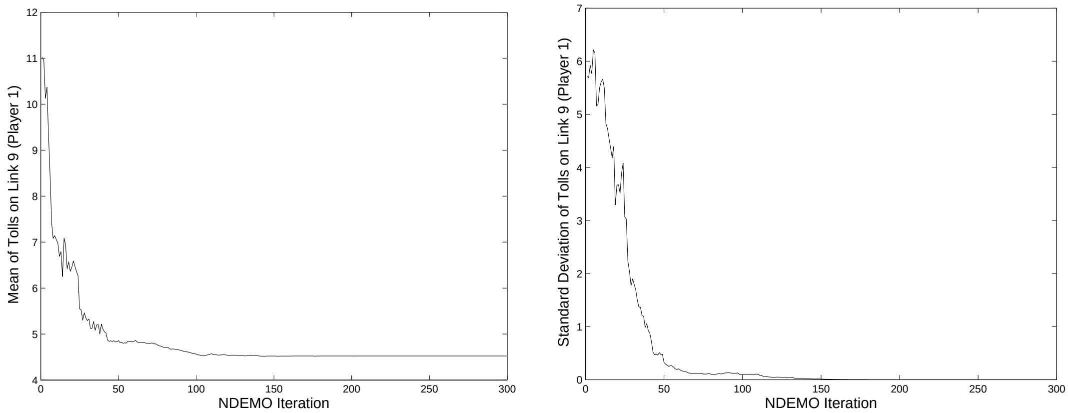

Figure 7: Example 4-Mean and Standard Deviation of Toll for Player 1 on Link 9

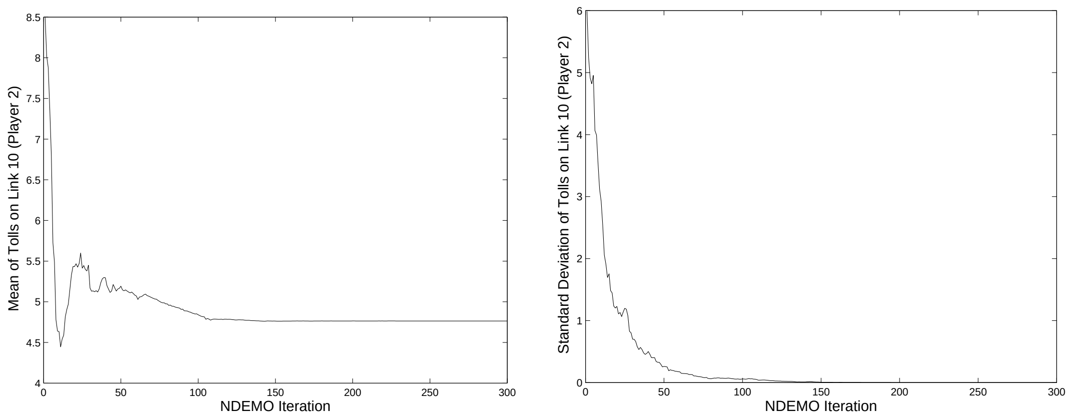

Figure 8: Example 4-Mean and Standard Deviation of Toll for Player 2 on Link 10

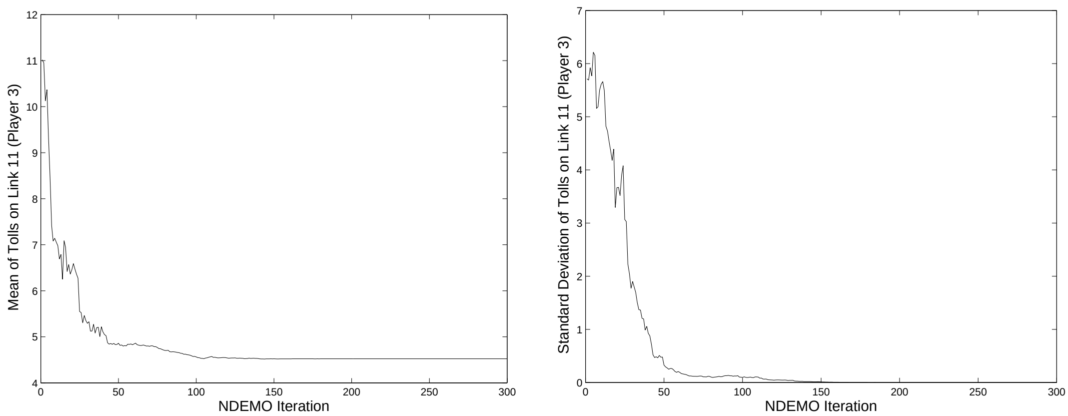

Figure 9: Example 4-Mean and Standard Deviation of Toll for Player 3 on Link 11

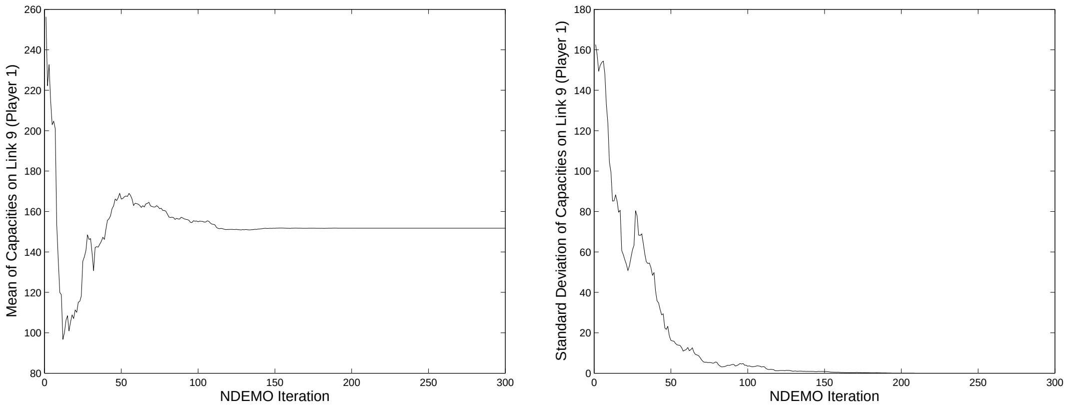

Figure 10: Example 4-Mean and Standard Deviation of Capacity for Player 1 on Link 9

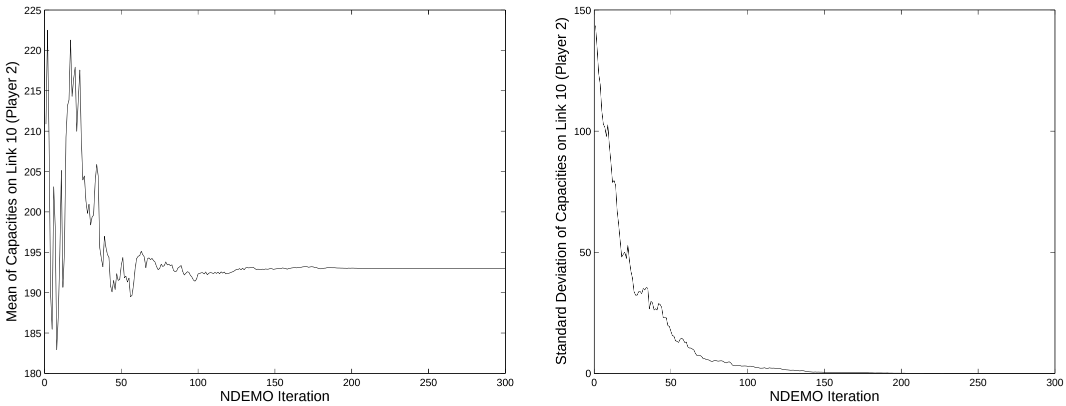

Figure 11: Example 4-Mean and Standard Deviation of Capacity for Player 2 on Link 10

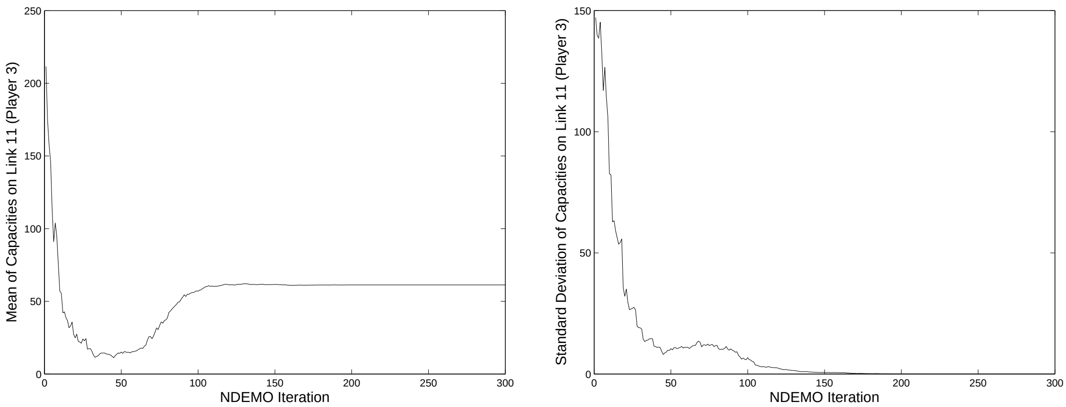

Figure 12: Example 4-Mean and Standard Deviation of Capacity for Player 3 on Link 11

$$U_i(q_i, q_{-i}) = \lambda_i^* q_i - c_i(q_i), i \in \{1, 2\} \tag{14}$$

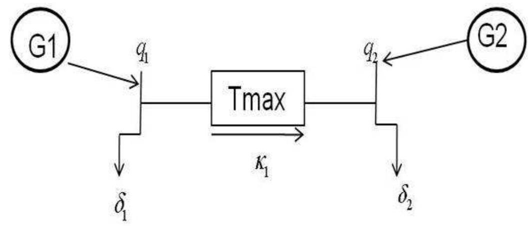

Figure 13: Two bus network model from [54], Players 1 and 2 are located at G1 and G2 respectively.

However, as mentioned before, the prices λ_1^*, λ_2^* can only be determined by the solution of a market clearing problem carried out by the ISO [15], [17]. Specific to the Two Bus Model shown in Figure 13 with player 1 and 2 located at buses G1 and G2 respectively, the ISO's market clearing task is embodied in the solution of the system of equations in 15 for given bid submissions i.e. $\{q_1, q_2\}$.

$$\max_{\delta_1,\delta_2,\kappa_1} \quad B_1(\delta_1) + B_2(\delta_2) \tag{15a}$$

$$\text{subject to}$$

$$\delta_1 - q_1 + \kappa_1 = 0 \tag{15b}$$

$$\delta_2 - q_2 - \kappa_1 = 0 \tag{15c}$$

$$-T^{Max} \leq \kappa_1 \leq T^{Max} \tag{15d}$$

The objective function 15a of the market clearing problem is maximization of the total benefits, the equality constraints in 15b and 15c represent Kirchoff's Law and the inequality constraint 15d represents the transmission limits on the line. The prices, λ_1^* and λ_2^*, are given by the Lagrange multipliers of the equality constraints in the market clearing problem 15b and 15c respectively. Within the scope of this paper it is the market clearing problem 15 that represents the equilibrium constraint facing each player which is a function not only of their own bids but also that of their competitor's.

The benefit functions at each node/bus and the cost functions for each player from [54] are shown in Table 10. The line transmission limit T^{Max} is 80. Table 11 compares the results from [54] with that obtained by NDEMO where 200 iterations were required to achieve the tolerance ϵ of 1e-4.

Table 10: Parameters of the Demand Function for Two Bus Model [54]

Bus/Player	Benefit Function $B_i(\delta_i)$	Cost $c_i(q_i)$
1	$-0.08(\delta_1)^2 + 50(\delta_1)$	$0.01(q_1)^2 + 10(q_1)$
2	$-0.04(\delta_2)^2 + 30(\delta_2)$	$0.01(q_2)^2 + 10(q_2)$

Table 11: Example 5 - 2 Bus Model NCEPEC Bid Quantities (Megawatts/Hr)

Player	1	2
[54]	148	148
Results from NDEMO:		
Mean	148.1267	148.1542
Standard Deviation	0.000289	0.000097

6.3.2. Example 6: Three Bus Model with 3 players

In this example, we consider the three player model from [7] and the 3 bus network used is shown in Figure 14. Three players submit bids to generate electricity to maximize individual profits from the generation of electricity according to 16.

$$U_i(q_i, q_{-i}) = \lambda_i^* q_i - c_i(q_i), i \in \{1, 2, 3\} \tag{16}$$

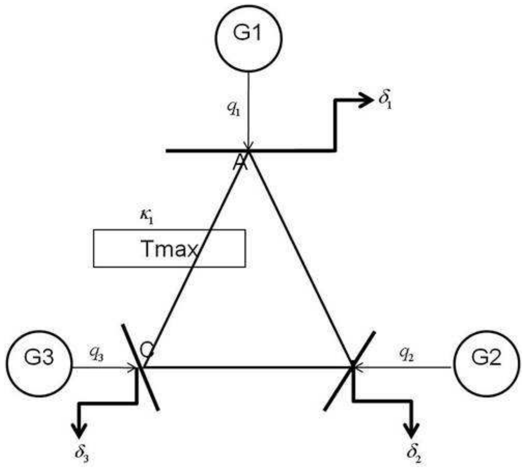

Figure 14: Three bus network model from [7],[54]. Players 1, 2 and 3 are located at G1,G2 and G3 respectively.

Once again the prices are determined by the market clearing problem given in 17. The equality constraints 17b, 17c and 17d represent the dc powerflow equations. The market clearing price tuple (the so-called locational marginal prices) $\lambda_i^*, i \in \{1, 2, 3\}$ is, as before, given by the Lagrange multiplier of these equality constraints. In this system of equations 17, θ_1 and θ_2 represent the powerflows on lines AC and -AC respectively. The last constraint 17e is the transmission limit on the line.

$$\max_{\delta_1,\delta_2,\delta_3} \quad B_1(\delta_1) + B_2(\delta_2) + B_3(\delta_3) \tag{17a}$$

subject to

$$2\theta_1 - \theta_2 = q_1 - \delta_1 \tag{17b}$$

$$-\theta_1 + 2\theta_2 = q_2 - \delta_2 \tag{17c}$$

$$-\theta_1 - \theta_2 = q_3 - \delta_3 \tag{17d}$$

$$-T^{Max} \leq \kappa_1 \leq T^{Max} \tag{17e}$$

The line transmission limit T^{Max} is 100 and the individual benefit functions at each node/bus and the cost functions for each player are shown in Table 12 as reported in [7].

Table 12: Parameters of the Demand Function for Three Bus Model [7]

Bus/Player	Benefit Function $B_i(\delta_i)$	Cost $c_i(q_i)$
1	$-0.0278(\delta_1)^2 + 108.4096(\delta_1)$	$0.0079(q_1)^2 + 1.360575(q_1) + 9490.366$
2	$-0.0335(\delta_2)^2 + 103.8238(\delta_2)$	$0.0105(q_2)^2 - 2.07808(q_2) + 11128.95$
3	$-0.0319(\delta_3)^2 + 105.6709(\delta_3)$	$0.0065(q_3)^2 + 8.105354(q_3) + 6821.482$

Table 13 compares the results from [7] with that obtained by NDEMO where 300 iterations were required to achieve the tolerance ϵ of 1e-4. Figures 15,16 and 17 show the mean and standard deviation of the population of each player's bids, $q_i, i \in \{1, 2\}$, over the iterations.

7. Conclusions

In this paper, we proposed modifying an evolutionary algorithm for solving EPECs by extending the procedure suggested in [33]. The resulting

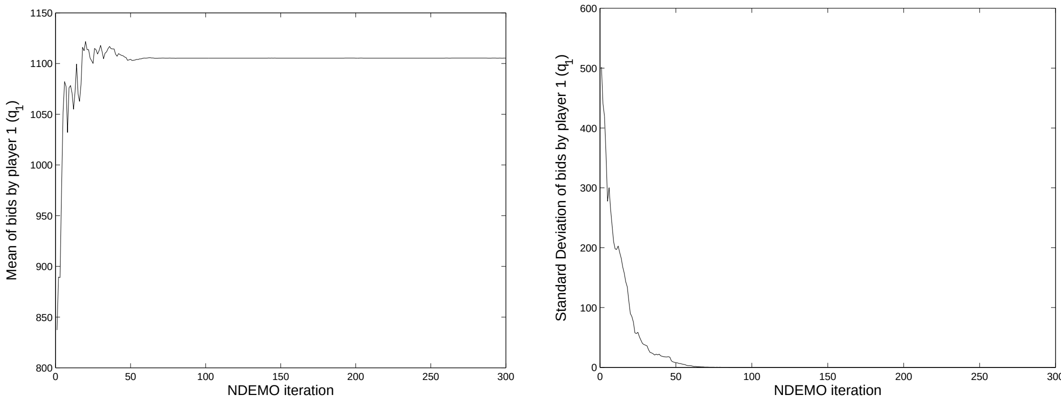

Figure 15: Example 6-Mean and Standard Deviation of Bids for Player 1 on Bus 1

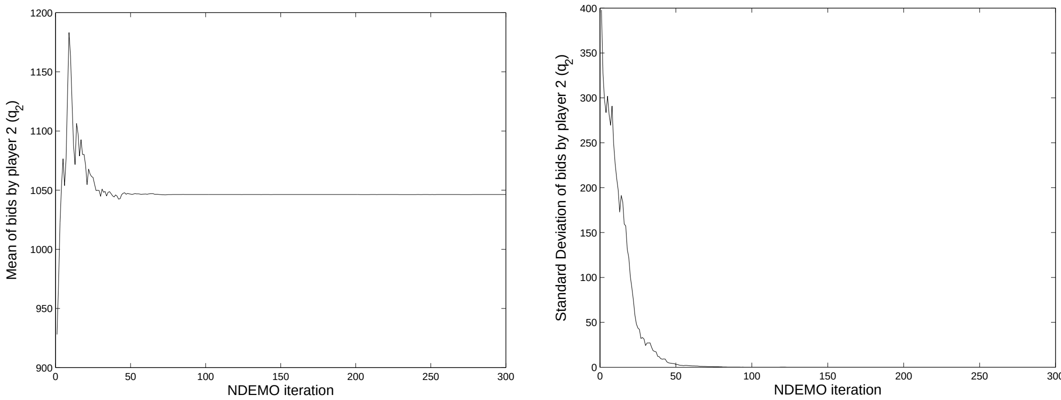

Figure 16: Example 6-Mean and Standard Deviation of Bids for Player 2 on Bus 2

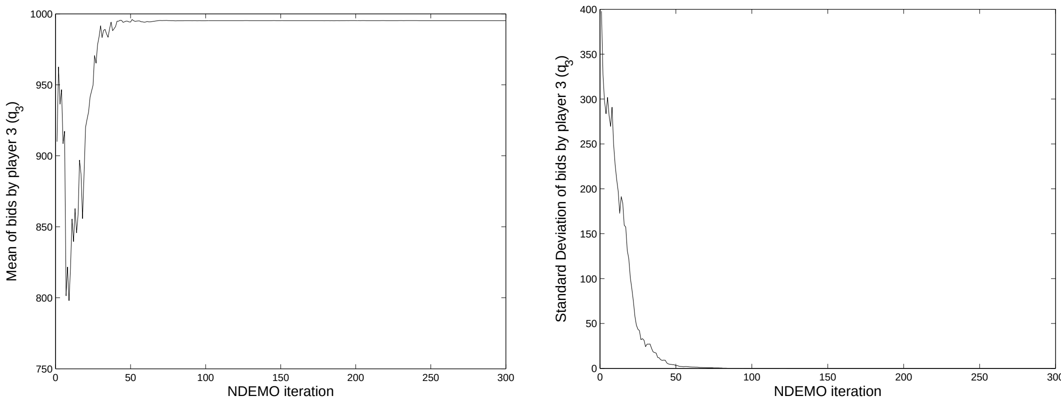

Figure 17: Example 6-Mean and Standard Deviation of Bids for Player 3 on Bus 3

Table 13: Example 6 - 3 Bus Model NCEPEC Bid Quantities (Megawatts/Hr)

Player	1	2	3
[7]	1105	1046	995
Results from NDEMO:			
Mean	1105.396	1046.238	995.177
Standard Deviation	0.00104	0.00053	0.0008815

Nash Domination Evolutionary Multiplayer Optimization (NDEMO) algorithm enabled us to handle Nash games where players encounter a system equilibrium constraint. We highlighted the fact that the critical Nash Domination procedure used in NDEMO to select between parent and child chromosomes is in fact theoretically rooted in the well established Nikaido Isoda function extending the original contribution of [33].

To assess the performance of NDEMO, six examples were given in this paper. The first,broken down into three case studies, used parameters from a well documented 5 player Cournot Nash model. The three case studies of the first example were given to underline the salient points of the market structure of competition assumed. In the first case study, we assumed that the players were competing non cooperatively but on an equal footing and this resulted in a standard Cournot Nash game for which NDEMO could be applied. In the second and third case studies, two players presented themselves as "market leaders " and this results in either the cooperative EPEC which is a MultiObjective Equilibrium Problem with Equilibrium Constraints (MOEPEC) (second case study) or the Non Cooperative Equilibrium Problem with Equilibrium Constraints (NCEPEC) (third case study). The proposed algorithm, NDEMO, is designed for the latter case and conventional evolutionary multiobjective optimization (EMO) algorithms could be used for the former. This example highlights the difference between a MOEPEC and a NCEPEC, with the former arising from the assumption of *cooperative* behaviour amongst the leaders and the latter stems from assuming that the leaders engage in a Nash game amongst themselves. In both cases the strategies the leaders can play are subject to the actions of the followers which manifests in the form of an implicit nonlinear constraint on the actions of the leaders.

Three numerical examples illustrating competition in private sector participation in highway transportation and two examples drawn from pool based bidding in the the electricity generation industry were further used to demonstrate the performance of NDEMO. In all instances, it was clear that NDEMO successfully converged to previously reported results in the literature and underscores the fact that the proposed algorithm is suitable for multidisciplinary applications.

While the examples suggest that this could be a potentially useful method for EPECs, we stress the need, in the pairwise comparison, to compute the payoff to each player, one by one, from deviating. This implies that the computational complexity of NDEMO increases significantly as the number of players increase as evidenced by the increase in computational times required in our examples. However, increasing the strategic variables available to each player did not significantly increase the time taken to solve the problem.

Further research would consider the effects of the control parameters of NDEMO on the speed of convergence to NND solutions. In this research we have used control parameters of the embedded Differential Evolution operators suggested in [51]. Nevertheless these parameters are in no way regarded as perfect and it is hypothesized that well chosen parameters may reduce the run time of the NDEMO algorithm.

Acknowledgement

The research reported here is funded by the Engineering and Physical Sciences Research Council of the UK under the "Competitive Cities" Grant EP/H021345/1. The author would like to thank two anonymous referees for their comments and suggestions on improving the presentation of an earlier draft.

References

[1] Bower J., Bunn DW Model-Based Comparisons of Pool and Bilateral Markets for Electricity, Energy Journal 21(3), 1–29 (2000)

[2] Červinka M: Hierarchical structures in equilibrium problems. PhD Thesis, Charles University, Prague, Czech Republic (2008)

[3] Chellapilla K, Fogel D: Evolving neural networks to play checkers without expert knowledge, IEEE Transactions on Neural Networks, 10(6), 1382–1391 (1999)

[4] Coello-Coello C, Lamont G: Applications of multi-objective evolutionary algorithms. World Scientific, Singapore (2004)

[5] Contreras J, Klusch M, Krawzyck JB Numerical solutions to Nash-Cournot equilibrium in electricity Markets. IEEE Transactions on Power Systems 19(1), 195–206 (2004)

[6] Curzon Price T: Using co-evolutionary programming to simulate strategic behavior in markets. Journal of Evolutionary Economics 7(3), 219–254 (1997)

[7] Cunningham L B, Baldick R, Baughman M L: An Empirical Study of Applied Game Theory: Transmission Constrained Cournot Behavior, IEEE Transactions on Power Systems 22(1), 166–172 (2002)

[8] Dafermos S C: Traffic Equilibrium and Variational Inequalities. Transportation Science 14(1),42–54 (1980)

[9] Dirkse S P, Ferris M C: The PATH solver: A non-monotone stabilization scheme for mixed complementarity problems, Optimization Methods and Software, 5 (2), 123-156 (1995)

[10] Ehrenmann A: Manifolds of multi-leader Cournot equilibria. Operations Research Letters 32(2), 121–125 (2004)

[11] Deb K: Multi-objective optimization using evolutionary algorithms. John Wiley, Chichester (2001)

[12] Facchinei F, Kanzow C: Generalized Nash equilibrium problems. Annals of Operations Research 175(1), 177–211 (2010)

[13] Facchinei F, Pang J S: Finite Dimensional Variational Inequalities and Complementarity Problems Volume 1. Springer, New York (2003)

[14] Facchinei F, Pang J S: Finite Dimensional Variational Inequalities and Complementarity Problems Volume 2. Springer, New York (2003)

[15] Gan D, Bourcier, D V: A single-period auction game model for modeling oligopolistic competition in pool-based electricity markets, IEEE Power Engineering Society Winter Meeting, 101–106 (2002)

[16] Gabay D, Moulin H: On the uniqueness and stability of Nash-equilibria in non cooperative games, In: Bensoussan A, et al, (eds) Applied Stochastic Control in Econometrics and Management Science. North Holland, Amsterdam, 271–293 (1980)

[17] Glover J D, Sarma M S, Overbye T, Power Systems Analysis and Design, Thomson: Toronto (2008)

[18] Harker P T: A variational inequality approach for the determination of Oligopolistic market equilibrium. Mathematical Programming 30(1), 105–111 (1984)

[19] Harker P T: Generalized Nash games and quasi-variational inequalities. European Journal of Operational Research 54(1), 81–94 (1991)

[20] Haurie A, Krawczyk J: Optimal charges on river effluent from lumped and distributed sources. Environ Model Assess 2(3), 177-189 (1997)

[21] Hodge B M, Pettersson F, Chakraborti N: Re-evaluation of the optimal operating conditions for the primary end of an integrated steel plant using multi-objective Genetic Algorithms and Nash Equilibrium, Steel Research International 77(7), 459–461 (2006)

[22] Hu X, Ralph D: Using EPECs to model bilevel games in restructured electricity markets with locational prices. Operations Research 55(5), 809–827 (2007)

[23] Huang, V L, Qin, A K, Suganthan P N, Tasgetiren M F:Multi-objective optimization based on self-adaptive differential evolution algorithm. Proceedings of IEEE CEC, 3601–3608 (2007)

[24] von Heusinger A, Kanzow C: Relaxation methods for generalized Nash equilibrium problems with inexact line search. Journal of Optimization Theory and Applications 143(1), 159–183 (2009)

[25] Judd K: Numerical methods in Economics, MIT Press, Cambridge,MA (1998)

[26] Karamardian S: Generalized complementarity problems. Journal of Optimization Theory and Applications 8(3), 161–168 (1971)

[27] Koh A: Coevolutionary particle swarm algorithm for bi-level variational inequalities: applications to competition in highway transportation networks in Chiong R, Dhakal S(eds) Natural intelligence for scheduling, planning and packing problems. Springer, Berlin, 195–217 (2010)

[28] Koh A, Shepherd S: Tolling, collusion and equilibrium problems with equilibrium constraints. European Transport/Trasporti Europei 43:3–22 (2010)

[29] Koh A: Nash Dominance with Applications to Equilibrium Problems with Equilibrium Constraints in Xiao-Zhi Gao, Antnio Gaspar-Cunha, Mario Kppen, Gerald Schaefer and Jun Wang (eds) Soft Computing in Industrial Applications: Algorithms, Integration, and Success Stories, Advances in Intelligent and Soft Computing Volume 75 Springer Verlag: Berlin, 71–79 (2010)

[30] Kolstad M, Mathisen L: Computing Cournot-Nash equilibrium. Operations Research 39(5), 739–748 (1991)

[31] Luo Z, Pang J, Ralph D: Mathematical Programs with Equilibrium Constraints. Cambridge University Press, Cambridge

[32] Leyffer S, Munson T: Solving multi-leadercommon-follower games. Optimization Methods and Software 25(4), 601-623 (2010)

[33] Lung R I, Dumitrescu D: Computing Nash equilibria by means of evolutionary computation. International Journal of Computers, Commmunications and Control III, 364–368 (2008)

[34] Murphy F H, Sherali H D, Soyster A L: A mathematical programming approach for determining oligopolistic market equilibrium. Mathematical Programming 24(1), 92–106 (1982)

[35] Mordukhovich B S: Optimization and equilibrium problems with equilibrium constraints. Omega 33(5), 379–384 (2005)

[36] Mordukhovich B S: Variational analysis and generalized Differentiation, II: Applications. Grundlehren der mathematischen wissenschaften, Vol 331, Springer, Berlin (2006)

[37] Mordukhovich B S, Outrata J V,Červinka M: Equilibrium problems with complementarity constraints: Case study with applications to oligopolistic markets. Optimization 56(4), 479–494 (2007)

[38] Nagurney A: Network Economics: A Variational Inequality Approach. Advances in Computational Economics, Kluwer, Boston (1999)

[39] Nash J: Equilibrium points in N-person games. Proceedings of the National Academy of Science USA 36(1), 48–49 (1950)

[40] Ortuzar J, Willumsen L: Modelling Transport. Wiley, Chichester(1990)

[41] Outrata J, Kočvara M, Zowe J: Nonsmooth Approach to Optimization Problems with Equilibrium Constraints. Kluwer, Boston (1998)

[42] Outrata J V: A note on a class of equilibrium problems with equilibrium constraints. Kybernetika 40(5), 585-594

[43] Pedroso J P: Numerical solution of Nash and Stackelberg equilibria: an evolutionary approach. Proceedings of SEAL'96, 151–160 (1996)

[44] Potter M A, De Jong K: A cooperative coevolutionary approach for function optimization. In: Proceedings of PPSN III, Springer, Berlin, 249–257 (1994)

[45] Price K (1999), An Introduction to Differential Evolution in Corne D, Dorigo M, Glover F(eds) New Techniques in Optimization. London: McGraw Hill, 79–108.

[46] Price K, Storn R, Lampinen J: Differential evolution: a practical approach to global optimization. Springer, Berlin (2005)

[47] Protopapas M, Kosmatopoulos E: Determination of sequential best replies in n-player games by genetic algorithms. International Journal of Applied Mathematics and Computer Science 5(1), 19–24 (2009)

[48] Rajabioun R, Atashpaz-Gargari E, Lucas C: Colonial competitive algorithm as a tool for Nash equilibrium point achievement. Proceedings of ICCSA, LNCS 5073, Springer, Berlin, 680–695 (2008)

[49] Rapoport A, Chammah A: Prisoner's Dilemma. University of Michigan Press, Ann Arbour, Michigan (1965)

[50] Razi K, Shahri S H, Kian A R: Finding Nash equilibrium point of nonlinear non-cooperative games using coevolutionary strategies. Proceedings of ISDA, 875–882 (2007)

[51] Robič T, Filipič B: DEMO: differential evolution for multiobjective problems. Proceedings of EMO2005, LNCS 3410,Springer, Berlin, 520–533 (2005)

[52] Rosen J B: Existence and uniqueness of equilibrium point for concave N person games. Econometrica 33(3), 520–534 (1965)

[53] Sefrioui M, Periaux J: Nash genetic algorithms: examples and applications. Proceedings of IEEE CEC, 509-516 (2000)

[54] Son Y, Baldick R: Hybrid coevolutionary programming for Nash equilibrium search in games with local optima. IEEE Transactions on Evolutionary Computation 8(4), 305–315 (2004)

[55] Smith M J The existence, uniqueness and stability of traffic equilibria. Transportation Research Part B 13(4),295–304 (1979)

[56] von Stackelberg, H H: The theory of the market economy. William Hodge, London (1952)

[57] Su C: Equilibrium problems with equilibrium constraints: stationarities, algorithms and applications. PhD Thesis, Stanford University, California, USA (2005)

[58] Wardrop J G: Some theoretical aspects of road traffic research. Proceedings of Institution of Civil Engineers Part II, 1(36), 325-378 (1952)

[59] Webb J N: Game theory: decisions, interaction and Evolution. Springer, London (2007)

[60] Yang H, Feng X, Huang H: Private road competition and equilibrium with traffic equilibrium constraints. Journal of Advanced Transportation 43(1), 21–45 (2009)

[61] Ye J J, Zhu Q J: Multiobjective optimization problem with variational inequality constraints. Mathematical Programming Series A 96(1), 139–160 (2003)

[62] Zubeita L: A network equilibrium model for oligopolistic competition in city bus services. Transportation Research Part B 32(6), 413-422 (1998)