Extended cooperative networks games

Alain Quilliot *, Fatiha Bendali, Jean Mailfert

Université BLAISE PASCAL and CNRS, LIMOS Laboratory, LIMOS, UMR CNRS 6158, Bat ISIMA, Campus des Cézeaux BP 10125, 63173 Aubière, France

A R T I C L E   I N F O

Article history:
Received 1 September 2004
Received in revised form 3 July 2007
Accepted 8 April 2008
Available online 11 September 2008

Keywords:
Cooperative games
Pricing
Fixed point
Network design

A B S T R A C T

We present here a pricing model which is an extension of the Cooperative Game concept and which includes a notion of Price-Dependent Demand. We present some existence results as well as some algorithms, and conclude by discussing a specific problem related to Network Pricing.

© 2008 Elsevier B.V. All rights reserved.


1. Introduction

Pricing problems (see [3,6–8,15,20,22,23]) are becoming an important topic in Operation Research, and this is because current deregulation processes turn pricing into a crucial issue for the companies whose activity is related to Transport, Telecommunications and Energy. Until recently, the gap between theoretical models, most of them derived from Game Theory (Cooperative Games [3,6–8,14,20,29], Non-Cooperative Games [2,9,13,21,24], Concurrential Equilibrium Theory [1,2,26]), and potential practical applications, was really huge, essentially due to the difficulties raised by cost and demand measurement. But now, powerful integrated management software and more efficient information systems make possible to provide these models with reliable data. Thus, technological progress turns the idea of applying game theoretical tools to practical pricing and marketing into a more realistic prospect.

Theoretical pricing models may roughly be classified according to the way the pricing decision process works:

– some models basically consider pricing as a cost imputation process or a congestion regulation tool (see [5,10,15,17,18,25,30]), and take competition into account only in an implicit way (Oligopoly models, Cooperative Game models, Aumann–Shapley Pricing…);

– other models see pricing as the main parameter of any equilibrium between demand and production (Ramsay–Boiteux models, Nash equilibria, Value Theory, Non-Cooperative Game models…see [11,14]).

We are first going to propose here a general framework which aims at unifying these two points of view, and whose main characteristic is the introduction of price-dependency into the Cooperative Game framework. Next we shall look at the way some pricing models related to network design may be cast into this framework.

2. A model for pricing with price-dependent demands

2.1. Mathematical notations

Let X be a set. We denote by \( P(x) \) the set of all the subsets of X, and by \( \text{Card}(X) \) the cardinality of X. We call X-indexed vector any real valued vector \( z = (z_x, x \in X) \) whose indexation set is X. We denote by \( 1 \) the X-indexed vector whose all coordinates are equal to 1 and by \( 0 \) the X-indexed vector whose all coordinates are equal to 0.
If \( z \) is a \( X \)-indexed vector and if \( A \) is a subset of \( X \), then we denote by \( z^A \) the Projection of \( z \) on \( A \), that means the \( X \)-indexed vector which is null for any index \( x \) in \( X - A \) and which coincides with \( z \) for any \( x \) in \( A \). We denote by \( z_A \) the Restriction of \( z \) on \( A \), which is the \( A \) indexed vector which coincides with \( z \) on \( A \).

\( Y \) and \( Z \) being two sets, a multiapplication from \( Y \) to \( Z \) is a function \( f \) which, with any element \( y \) in \( Y \), associates some subset \( f(y) \) of \( Z \), i.e., some element of \( P(Z) \).

We denote by \( R \) the set of the real numbers, and by \( R^{\text{Card}(X)} \) the set of all \( X \)-indexed vectors.

If \( X \) and \( Y \) are two sets, we denote by \( X \times Y \) the product of \( X \) and \( Y \), that means the set of all pairs \((x, y)\) such that \( x \in X \), \( y \in Y \).

2.2. Recall: Cooperative games

A Cooperative Game \( J = (X, V) \) is defined by a finite set \( X \) and by a Cost function \( V \) from \( P(X) \) to \( R \), which are such that:

\* \( V(\emptyset) = 0; \)
\* \( V \) is increasing: if \( A \subset B \) then \( V(A) \leq V(B) \).

The set \( X \) is called the Player Set of the game \( J \), and every element of \( X \) is called a player. The Cost function \( V \) is also called the Characteristic Function of the game \( J \).

For such a game \( J \), the set \( P(X) \) is called the Coalition Set of the game \( J \): every subset of \( X \) is called a coalition. Thus, the Cost function \( V \) associates, with any coalition \( A \) of the game \( J \), the cost or characteristic value of the coalition \( A \) for the game \( J \).

\( J \) is said to be sub-additive if for any \( A, B \in P(X) \), we have \( V(A \cup B) \leq V(A) + V(B) \).

Let \( J = (X, V) \) be some cooperative game. Because of the link between Pricing and cooperative games, we call price vector any vector \( X \)-indexed vector \( p = (p_x, x \in X) \) such that \( 0 \leq p \). Such a price vector \( p \) will be told here to be an imputation vector for the game \( J \) if \( \sum_{x \in X} p_x = V(X) \). Then the Core of \( J \) is the set \( CO(J) \) of all the price vectors \( p = (p_x, x \in X) \) such that:

\* \( \sum_{x \in X} p_x = V(X); \)
\* For any coalition \( A \in P(X) \), \( \sum_{x \in A} p_x \leq V(A) \).

Of course, any price vector \( p \) which belongs to the core set \( CO(J) \) is an imputation for \( J \). This core set may be empty.

Remark 0. The above definition of an imputation vector is not standard of Game Theory. The usual definition states that a vector \( p = (p_x, x \in X) \) is an imputation vector of the game \( J \) iff: \( \sum_{x \in X} p_x = V(X) \) and \( p_x \leq V(\{x\}) \) for all \( x \in X \). According to this definition, there is no lower bound on the components \( p_x \). Still, the monotonicity of \( V \) makes that the Core, as it is defined above, coincides for both definitions. So, although our own definition of an imputation vector as a positive vector is not standard, we are going to work with it here, since it will be easier to handle inside our framework and since it is closest to intuition when it comes to pricing.

In the case of a pricing problem, the Core concept reflects a notion of stability, which we may illustrate through two examples.

First example: The player set \( X \) is a customer set.

Let us suppose that a Telecommunication operator \( O1 \) is settling an infrastructure \( I \), which is likely to be used by a customer set \( X \). Then a coalition \( A \) may be viewed as a market share, and the cost value \( V(A) \) is the cost which would be induced by an infrastructure \( I_A \) endowed with the same technical characteristics as \( I \), and dedicated to the members of \( A \) by a competitor \( O2 \) of \( O1 \). The eventual non-vacuity of the Core will allow \( O1 \) to protect itself from losing such a market share.

Second example: The player set \( X \) is a product or service set.

Let us suppose now that \( X \) is a set of products which are going to be put on sale by a producer \( P \). Then the cost value \( V(A) \), \( A \subset X \), is the cost that \( P1 \) would have to face if he decided only to produce \( A \). Pricing \( X \) according to a price vector in the Core \( CO(J) \) prevents any competitor \( P2 \), submitted to the same cost constraints as \( P1 \), from proposing better prices than \( P1 \) while limiting his production to \( A \).

Remark 1. The Core concept does not take into account competitors in an explicit way. In order to clearly make appear the various firms and goods operating on a given market, one needs to introduce non-cooperative game models \([1,2,12,21,24,26]\).

2.3. Recall: Ramsay–Boiteux model for pricing

This model takes into account the fact that demands are price-dependent. It considers an operator which is able to put on the market a set \( I \) of goods and services. The production cost of a production vector \( d = (d_i, i \in I) \) is a quantity \( C(d) \) and the demand induced by a price vector \( p = (p_i, i \in I) \) is a function \( D(p) = (D_i(p), i \in I) \) of \( p \). Then the Ramsay–Boiteux pricing process works through a formula which computes the prices as an inverse function of the \( D_i \) functions.
2.4. A mixed model: Cooperative games with price-dependent demands

The price dependency of the demands, which is present in the Ramsay–Boiteux model as well as in the Non-Cooperative Games models, needs to be taken into account as soon as strategic decision related to market segmentation is at stake. Practically, trouble comes from the fact that modelling price dependency is very difficult. Still, recent advances in Data mining and in the structuration of industrial information systems make dealing with this problem easier than before, and provide the motivation of the approach which are going to present here.

In order to better explain our approach, let us come back to the two above examples.

**First example**: The player set is a customer set.

Let us consider operator O1 of our previous example, and let us suppose that X is a set of origin/destination pairs asking for connections. Then any connection involving such a pair x will be affected with some price \( p_x \). This price will induce a demand level \( d_x \), which will be a fraction of all the potential connections related to x. This demand will depend on the whole vector \( p = (p_y, y \in X) \), and satisfying the demand vector \( d = (d_y, y \in X) \) will in turn induce a cost \( C(d) \) for the operator O1. Then O1 will have to fix \( p \) in such a way that financial equilibrium is ensured and that no operator O2, which would work under similar technological constraints may propose, while limiting itself to some well-choosen subset (coalition) A of the customer set X, a cheaper service.

**Second example**: The player set is a service set.

By the same way, we may imagine that, for any service proposed by operator P1, the demand \( d_x \) generated by service x depends on the unit price vector \( p = (p_x, x \in X) \), and that the demand vector \( d = (d_x, x \in X) \) induces in turn a production cost \( C(d) \). Then, a stable price system \( p \) will be such that no alternative product or P2 may cut off some market share from P1, while limiting its activity to some well-choosen service subset A (coalition) of X and while proposing better prices for those services.

This leads us to introduce the following general model of Cooperative Game with Price-Dependent Demands, which tries to unify both previous models:

An instance of such a Cooperative Game with Price-Dependent Demands will be defined by a triple \( G = (X, C, D) \) in such a way that:

- \( X \) is a finite set (customers or services), which will be called the Player Set:
  - any subset \( A \) of \( X \) will be called a coalition;
  - we shall associate with the player set \( X \) two specific sets of \( X \)-indexed vectors: the Price Set \( \text{PRICE}(X) \) of the unit price vectors \( p = (p_x, x \in X) \geq 0 \), and the Production Set \( \text{PROD}(X) \) of the production vectors \( d = (d_x, x \in X) \geq 0 \);
  - \( C \) is a continuous increasing Cost or Characteristic function, which makes correspond to any production vector \( d \in \text{PROD}(X) \) a cost \( C(d) \), in such a way that \( C(0) = 0 \) (E1);
  - \( D \) is a Demand function, which makes correspond, to any unit price vector \( p = (p_x, x \in X) \geq 0 \) in \( \text{PRICE}(X) \), a production vector \( D(p) = (D(p)_x, x \in X) \geq 0 \) in \( \text{PROD}(X) \), in such a way that any component function \( D_x \) be continuous and decreasing (E2)

**Remark 2.** According to the above definition, the \( d_x \) quantities must be interpreted here as fractional market shares. Thus, our model is going to be a non-discrete one, and it will eventually allow null prices. Also, one must be careful with the unit price notion: we mean here any vector \( p = (p_x, x \in X) \geq 0 \), in which, for any \( x \in X \), \( p_x \) is the price of one unit of the commodity \( x \).

Then we shall say that a pair \( p = (p_x, x \in X) \geq 0 \), \( d = (d_x, x \in X) \geq 0 \), is in the Core of \( G \) iff:

- \( p \) is a price vector in \( \text{PRICE}(X) \) and \( d \) is a production vector in \( \text{PROD}(X) \);
- \( d = D(p) \);
- \( p.d = C(d) \); (Financial Global Equilibrium Constraint) (E3)
- there does not exist (Stability Constraint) any coalition \( A \subset X \), any price vector \( p' \) and any production vector \( d' \) such that:
  - \( p'_x \geq p_x \);
  - \( d'_x = 0; d'_A \geq d_A; \)
  - \( C(d') = p'_A, d'_A; \)
  - \( D(p') = d'_A \).

**Remark 3.** In the case of fixed (non-elastic) demands, i.e., in case any quantity \( D(p)_x, x \in X \) is fixed and equal to some number \( D^*_x \), one easily checks that the above model contains the usual Cooperative Game model, and that the above Core notion coincides with the usual Core notion for cooperative games. In such a case, an imputation vector \( p^* = (p^*_x, x \in X) \) of the cooperative game which derives from \( G \) is such that every price \( p^*_x \) must be understood as the product \( p_x D^*_x \) of some unit price \( p_x \) with the production \( D^*_x \). This vector is in the Core of the related cooperative game if and only if the pair \( (p = (p_x, x \in X) \geq 0, D^* = (D^*_x, x \in X) \geq 0) \) is in the Core of the the Cooperative Game with Price-Dependent Demands \( G \).
3. An existence result

Let \( G = (X, C, D) \) be a cooperative game with price-dependent demands. We say that the Demand function \( D \) is regular if it is strictly positive and if the vector product \( p.D(p) \) is a strictly increasing function of \( p \). Then we may set:

**General Existence Theorem 1.** Given a finite player set \( X \) and a Cost function \( C \) which satisfies the above (E1) hypothesis. We suppose that there exists some number \( \lambda \geq 0 \) such that for any production vector \( d \geq 0 \), \( C(d) \leq \lambda.N(d) \), where \( N \) denotes the Euclidean norm. Then the two statements (1) and (2) below are equivalent:

1. for any regular demand function \( D \) which satisfies (E2), the Core of the Cooperative Game with Elastic Demands \( G = (X, C, D) \) is not empty;
2. for any vector \( \mu = (\mu_A, A \in P(X)) \geq 0 \), which is balanced, that means which is such that for any \( x \) in \( X \), we have \( \Sigma_{A \in P(X)} \mu_A = 1 \), and for any production vector \( d = (d_x, x \in X) \geq 0 \), we have:

\[
C(d) = \Sigma_{A \in P(X)} \mu_A.C(d_A). \quad (E5)
\]

**Remark 4.** The hypothesis involving the quotient \( C(d)/N(d) \) is natural since it may be viewed as a consequence of the usual degressivity of the marginal production costs. But it implies that no fixed costs related to the initialization of the system are taken into account.

**Proof of Theorem 1.** We start by proving the following lemmas:

**Lemma 1.** Let \( C \) be a Cost function which satisfies the (E1) hypothesis, and let \( D \) be a Demand function which satisfies the (E2) hypothesis. Then any element which is in the core of the Cooperative Game with Time-Dependent Demands \( G = (X, C, D) \) is also a solution of the following program:

\[
Core(G): \text{(Compute } p = (p_x, x \in X) \geq 0 \text{ in PRICE}(X), \text{and } d = (d_x, x \in X) \geq 0 \text{ in PROD}(X), \text{such that):}
\]

- for any coalition subset \( A \) of \( X \), \( \Sigma_{x \in A} p_x.d_x \leq C(d_A) \);
- \( p.d = C(d) \);
- \( d = D(p) \).

**Proof of Lemma 1.** Let us check that if a pair \((p, d)\) in \( \text{PRICE}(X) \), \( \text{PROD}(X) \) is in the core of the Cooperative Game with Price-Dependent Demand \( G = (X, C, D) \), then it satisfies the constraints of the program \( \text{Core}(G) \). We only have to prove that if \( A \) is some coalition subset of \( X \), then:

\[
\Sigma_{x \in A} p_x.d_x \leq C(d_A).
\]

Let us suppose that the converse is true and let us consider a coalition \( A \subset X \) such that: \( \Sigma_{x \in A} p_x.d_x > C(d_A) \). Since \( C(d_A) \geq 0 \) and since the cost function \( C \) is monotonic, we may suppose, without any loss of generality, that \( p_x > 0 \) for any player \( x \) in \( X \). So we may consider, for any number \( \tau \geq 0 \), the price vector \( p' \) defined by:

\[
\begin{align*}
p'_{x-A} &= p_{x-A}; \\
p'_{x} &= \tau \cdot p_x.
\end{align*}
\]

Then we see that there exists \( \tau^* \) in the \([0, 1]\) interval such that \( p' = \tau^* \cdot D(p'^*) \).

Then it becomes sufficient to set:

\[
\begin{align*}
p' &= p'^*; \\
d'_{x-A} &= 0 \quad \text{and} \quad d'_x = D(p'^*)_A;
\end{align*}
\]

in order to get the (E4) relationships of the previous section, and to deduce that the pair \((p, d)\) cannot belong to the core of the cooperative game with time-dependent demands \( G \).

**End-Lemma.**

**Lemma 2.** Let \( D \) some regular Demand function. Then the core of the cooperative game with price-dependent demands \( G = (X, C, D) \) is exactly the set of the solution set related to the constraints of \( \text{Core}(G) \).

**Proof of Lemma 2.** Lemma 1 tells us that any element of the core of \( G \) is also a solution of the program \( \text{Core}(G) \). Conversely, let \( p \) be some price vector and \( d \) be some production vector such that the pair \((p, d)\) is a solution of the constraint set \( \text{Core}(G) \). Let suppose that there exists a coalition subset \( A \) of \( X \) and a price vector \( p' \) which satisfy (E4), that means which are such that:

- \( p'_{x-A} = p_{x-A}; \quad p'_{x} < p_x; \)
- \( C(D(p')^A) = p'_A.D(p')_A. \)

We must have (since the demand function \( D \) is decreasing) \( D(p')_A \geq D(p)_A \), which implies (since \( C \) is increasing) that \( C(D(p')^A) \geq C(D(p)^A) \). We also must have (because of the constraints of \( \text{Core}(G) \)): \( \Sigma_{x \in A} p_x.d_x \leq C(D(p)^A) \). But the Regularity hypothesis tells us that the scalar product \( q.D(q) \) is a strictly increasing function of the unit price \( q \), and we also know that the demand \( D(q) \) is a decreasing function of the unit price \( q \). This allows us to deduce the inequality \( p'_x.D(p')_A < p_x.D(p)_A \), which induces a contradiction.

**End-Lemma.**
Proof of Theorem 1

Let us first define the Cooperative Game $J(G, d) = (X, V_{G,d})$ which may be associated with $G$ and with any fixed production vector $d$ in $\text{PROD}(X)$, by setting:

- for any coalition subset $A$ of $X$, the characteristic value $V_{G,d}(A)$ is defined $V_{G,d}(A) = C(d^A)$.

Let us also recall the Bondareva–Shapley Theorem (see [4,27]): the core of a cooperative game $(X, V)$ is non-empty if and only if for any balanced vector $\mu = (\mu_A, A \in P(X)) \geq 0$, we have: $V(X) \leq \sum_{A \in P(X)} \mu_A V(A)$.

Then, let us prove the implication $(1) \Rightarrow (2)$.

In order to do it, let us proceed by refutation and let us suppose that $(2)$ is false, which means that there exists some production vector $d = (d_\alpha, \alpha \in X) \geq 0$ and some balanced vector $\mu = (\mu_A, A \in P(X)) \geq 0$, such that the inequality $(E5)$ is not satisfied. Then we may consider some demand functions $D_\alpha, \alpha \in X$, which are constant and equal to $d_\alpha$. Those functions are regular (since $d$ is strictly positive) and they satisfy $(E2)$. If a pair $(p^*, d^*)$ is in the core of the cooperative game with elastic demands $X = (X, C, D)$, then the price vector $q^*$ which is defined by:

- for any player $x$ in $X$, $q^*_x = p^*_x d_x$,

must be in the core of the cooperative game $J(G, d)$. But the Bondareva–Shapley Theorem tells us that this core must be empty if $(E5)$ is not satisfied, which means that the pair $(p^*, d^*)$ cannot exist.

Let us now prove the converse implication $(2) \Rightarrow (1)$. Let us first recall that a convex, compact, and upper semi-continuous multiapplication from $R^m$ to $R^m$ is a function $\Gamma$ which with any $x$ in $R^m$ associates some convex and compact subset $\Gamma(x)$ of $R^m$ in such a way that, for any open subset $U$ of $R^m$, the subset $\Gamma^[U] = \{x \in R^m \text{ such that } \Gamma(x) \subset U\}$ is an open subset of $R^m$. We know (Tychonoff–Kakutani Theorem [28]), that if $A$ is some convex compact subset of $R^m$, and if $\Gamma$ is some convex, compact, and upper semi-continuous multiapplication from $A$ into itself, then $G$ admits some fixed point, which means that there exists $x \in A$ such that $x = \Gamma(x)$.

So, let us consider now some player set $X$ and some cost function $C$, which satisfy the hypotheses of Theorem 1, as well as some demand function $D$, which we suppose to be regular and to satisfy $(E2)$. Let us denote by $G$ the cooperative game with elastic demands $(X, C, D)$. For any unit price vector $p \geq 0$, the core of the cooperative game $J(G, D(p))$ is non-empty as soon as side (2) of Theorem 1 is true. In such a case, this core subset is a compact and convex subset of $K(\text{Card}(X))$ and we may define a convex, compact and upper semi-continuous multiapplication $K$ by setting, for any unit price vector $p = (p_\alpha, \alpha \in X) \geq 0 : K(p) = \text{Core of the cooperative game } J(G, D(p))$. Through composition, and taking into account the fact that the demand function $D$ is strictly positive (because of the regularity hypothesis), we see that the multiapplication $K^*$ which is defined, for any unit price vector $p = (p_\alpha, \alpha \in X) \geq 0$, by:

- $K^*(p) = \{u = (u_\alpha, \alpha \in X), \text{ such that } u_\alpha \text{ may be written } u_\alpha = q^*_\alpha / d_\alpha \text{ with } d_\alpha = D(p)_\alpha \text{ and } q_\alpha \in K(p)\};$

is also convex, compact and upper semi-continuous from $P^{\text{Card}(X)}$ into itself. Moreover, we also supposed the existence of a number $\lambda \geq 0$ such that for any production vector $d \geq 0$, $C(d) \leq \lambda N(d)$, where $N$ denotes the Euclidean norm. It comes that, if $p$ is some unit price vector, if $q$ is in $K(p)$ and if $x$ is in $X$, then the inequality $q_\alpha \leq C(d^{\alpha}) = C(D(p)_\alpha)$ also yields $q_x / d_x \leq \lambda$. This means that if $u$ is in $K^*(p)$, then we have $u_\alpha \leq \lambda$. We deduce that if $A$ is the subset of $\text{PRICE}(X)$ defined by:

- $A = \{p = (p_\alpha, \alpha \in X) \geq 0 \text{ such that } \sup_{x \in X} p_\alpha \leq \lambda\},$

then $K^*$ is a convex, compact and upper semi-continuous multiapplication from $A$ into itself. It follows that there exists a unit price vector $p^* \in A$ which is a fixed point for $K^*$. So we set $d^* = D(p^*)$, and we check that the price (pre-imputation) vector $q^*$ which is defined, for any $x$ in $X$, by: $q^*_x = p^* d^*_x$ is in the core of the cooperative game $J(G, d^*)$, and that the following relations are true:

- $1.q^* = C(d^*)$;
- for any coalition subset $A$ of $X$, $\sum_{x \in A} p^*_x d^*_x \leq C(d^A)$.

It comes that the unit price vector $p^*$ and the production vector $d^*$ define a solution of the constraint set $\text{Core}(G)$. We conclude by using the fact that the game $G(X, C, D)$ is regular and by applying Lemma 2. End-Theorem.

4. The case of a network design game

We are now going to present an adaptation of the above model to the case of a specific network design problem, according to which demand depends not only on prices but also on the Quality of Service (QoS) of the routing process.

In order to do it, we must first recall some definitions about flows and multicommodity flows: (see for instance [16] or [19]):

- let $H = (Z, E)$ some network: then a flow $F$ is defined on $H$ as an $E$-indexed vector such that for every vertex $z$ of $Z$, we have: $\sum_{e \in \text{extremity}(e) = z} F_e = \sum_{e \in \text{origin}(e) = z} F_e$ (Kirshoff Law);
- if $o$ and $s$ are two specific vertices (origin and destination) of $H$, and if $d$ is some number, we say that a $E$-indexed vector $f$ is a $d$-routing flow from origin $o$ to destination $y$, if:
  - for every vertex $z$ different from $o$ and $s$, we have: $\sum_{e \in \text{extremity}(e) = z} F_e = \sum_{e \in \text{origin}(e) = z} F_e$.
  - $\sum_{e \in \text{extremity}(e) = o} F_e - \sum_{e \in \text{origin}(e) = o} F_e = d$;
Then we may consider a strongly connected communication or transit network \( H = (Z, E) \) as well as a specific arc set \( U \subset E \), which is likely to support some high QoS (speed) transportation mobile infrastructure (for instance a shuttle fleet) \( F \), run by some transportation operator. We suppose that any arc \( e \in E \) is endowed with a financial cost \( c_e \) and with a routing QoS cost \( t_e \), in such a way that the routing QoS cost \( t_e \) is significantly smaller if \( e \in U \) than if \( e \notin U \). We say that the vector \( c = (c_e, e \in E) \) is the financial cost vector related to the network \( H \) and that the vector \( t = (t_e, e \in E) \) is the routing cost vector related to the network \( H \).

The Player (Customer) Set \( X \) is defined here together with a family of origin/destination pairs of vertices \( (o_x, s_x), x \in X \). Its meaning is that any player \( x \in X \) must be viewed as some customer group which is a potential user of the mobile infrastructure \( F \), and which will use it in order to route goods, passengers, luggages or messages from \( o_x \) to \( s_x \). The demand \( d_x \) of this player for such a routing service involving \( F \), is going to depend on both the unit price \( p_x \) which is asked to the customer group (player) \( x \) for the access to \( F \), and on the routing QoS (Quality of Service) \( T_x \) of the induced connection. The vectors \( p = (p_x, x \in X) \), \( T = (T_x, x \in E) \) and \( d = (d_x, x \in X) \) will, respectively, define a unit price vector, a routing QoS cost vector and a demand vector.

Infrastructure Operator Decisions: Let us think into \( H \) as into a transit network, which means that coefficients \( t_e, e \in E \), are time coefficients. Then, a infrastructure operator decision consists, for a given operator who manages a shuttle fleet, in determining routes and traffic volumes. This decision might be summarized as some infrastructure flow \( F \), with support on the arc set \( U \). This infrastructure flow \( F \) being fixed, every player \( x \) will be proposed a route from \( o_x \) to \( s_x \), whose mathematical description will come as a flow \( f(x) \) which will ensure the routing of some demand \( d_x \) from \( o_x \) to \( s_x \). In many cases, it will happen that part of the moves performed by \( x \) will not involve the shuttle fleet \( F \). the role of \( U \) is here to distinguish, inside the transit network \( H \), the player moves which involve \( F \), from those which do not. The way \( F \) will be designed and priced will induce, for any player \( x \in X \), some running time (QoS level) \( T_x \), and, consequently, some demand level \( d_x \) as well as some routing decision which might be summarized by some flow \( f(x) \). So, infrastructure and pricing decisions should be simultaneously taken, while meeting the induced demands and ensuring a financial equilibrium. But, if we suppose that demand levels \( d_x \) and QoS levels \( T_x, x \in X \), are already known, the operator only must run its fleet under the smallest possible costs, while ensuring, for any player \( x \in X \), to go from origin \( o_x \) to destination \( s_x \) in no more than \( T_x \) time units. In such a case, the pair \( (d = (d_x, x \in X) = \text{demand vector}, T = (T_x, x \in X) = \text{routing QoS cost vector}) \) will be called, by analogy with Sections 2 and 3, a Production Pair, and our operator should solve the following program NETWORK\((d, T)\):

**Linear Program NETWORK** \((d, T)\): (Find a flow \( F \geq 0 \) and a multicommodity flow \( f = f(x), x \in X \), both defined on \( H \), such that:

- the flow \( F \) represents the mobile infrastructure decision of the operator (the mean traffic of the shuttles), and only involves arcs of \( U \), which means: for any \( e \in U \), \( F_e = 0 \);
- every flow \( f(x) \) is a \( d_x \)-routing flow which represents the routing of the demand (passengers, goods, luggage...) \( d_x \) from \( o_x \) to \( s_x \) and which is submitted to the following QoS constraint: \( d_x T_x \geq f(x) \);

- \( F \) and \( f \) are tied by a capacity constraint on the arcs of \( U \) for any \( e \in U \), \( F_e \geq \sum_{x \in X} f(x) \cdot \).

(Comment: this constraint is an equality, since \( F \) is a mobile infrastructure, which needs to follow its route, even when it is not fully loaded)

and which minimize the quantity \( c.F \), where \( c \) is the financial cost vector which we previously defined).

This program roughly summarizes the infrastructure decision problem which has to be solved by the shuttle fleet operator if the production pair \((d, T)\) is such that the demand vector \( d = (d_x, x \in X) \) and the related QoS cost bounds \( T_x, x \in X \) are considered as being fixed. We denote by \( W(d, T) \) the optimal value of this program and we call it **Production Cost** of the production pair \((d, T)\) defined by the demand vector \( d \) under the routing QoS cost vector \( T \).

We also denote by **FLOW**(\(H\)) the set of all flow \( F \geq 0 \) which are defined on \( H \), and which are such that \( F_e = 0 \) if \( e \notin U \), and by **MULT-FLOW**(\(H\)) the set of all multicommodity flows \( f = f(x), x \in X \) \( \geq 0 \) such that flow \( f(x) \) is a \( d_x \)-routing flow which represents the routing of the demand (passengers, goods, luggages...) \( d_x \) from \( o_x \) to \( s_x \).

**Remark 5.** We should think in a player \( x \) as in an aggregate of users of the mobile infrastructure \( F \). Thus, flow \( f(x) \) does need to involve a single path from \( o_x \) to \( s_x \), which means that it may takes values on the arc set \( E \) which are different from \( 0 \) and \( d_x \).

**An example**

As we just told, a good way to understand the meaning of the program NETWORK\((d, k)\) is to suppose that \( H \) is a transportation network, that the flow \( F \) represents the routes which are going to be periodically run by a fleet of vehicles, and that every demand \( d_x, x \in X \) is related to a quantity of goods which are required to be transported during every period from an origin vertex \( o_x \) to a destination vertex \( s_x \). According to such an interpretation, a routing QoS cost \( t_e \) may be associated with the time required to run along an arc \( e \). Then, for any vertex \( x \in X \), the QoS inequality \( d_x T_x \geq f(x) \) expresses the fact that the average time required to route demand \( d_x \) from \( o_x \) to \( s_x \) should not exceed some threshold \( T_x \). Under those assumptions, solving the NETWORK\((d, T)\) program globally means, for the transportation operator who manages the vehicle fleet, determining the activity level of this fleet, in such a way that it meets, under the smallest running cost \( c.F \) and while taking into account the QoS constraints, the routing demands \( d_x, x \in X \).
Remark 6. Other examples could be drawn from contexts related to production management or to telecommunication systems. Still, for such examples, we should rather interpret coefficient \( T_k \) as a lower bound for an average delay or a congestion rate, and then express QoS constraints under a non-linear form. Also, we should in some cases model the infrastructure decision as an integral vector instead of representing it as a flow. It would not change anything in the way we set our problem and in its true meaning, but it would make more difficult the obtention of theoretical results similar to the forthcoming Theorem 2.

The network design multicriterion game

We may now define an associated Network Design Multicriterion Game by considering that the access demands \( d_x, x \in X \), depend on both the unit access prices \( p_x, x \in X \), and on the routing QoS costs \( T_x = (t.f(x))/d_x, x \in X \), where \( t \) is the routing cost vector which we defined at the beginning of this section. That means that we suppose that for any player \( x \in X \), \( d_x \) may be written \( d_x = D_x(p_x, T_x) \), where every function \( D_x \) is continuous and decreasing.

By analogy with sections II and III, we set:

- \( \text{DEM}(X) = \) the set of all \( X \)-indexed demand vectors;
- \( \text{QS}(X) = \) the set of all \( X \)-indexed routing QoS cost vectors;
- \( \text{P-PROD}(X) = \) the set of all production pairs;
- \( \text{U-PRICE}(X) = \) the set of all \( X \)-indexed unit price vectors.

Then the previous model of Core of a Cooperative Game with Time-Dependent Demands may be extended by telling that a pair \( (p, (d, T)) \) defined by:

- a unit price vector \( p = (p_x, x \in X) \geq 0 \) in \( \text{U-PRICE}(X) \);
- a production pair \( (d, T) \in \text{P-PROD}(X) \), made of a routing QoS cost vector \( T = (T_x, x \in X) \geq 0 \) in \( \text{QS}(X) \) and of a demand vector \( d = (d_x, x \in X) \geq 0 \) in \( \text{DEM}(X) \);

is in the Core of the Network Design Multicriterion Game defined by the program NETWORK\((d, T)\) and by the demand functions \( (D_x, x \in X) \), iff:

- \( d = D(p,T); \)
- there exists an optimal solution \( (F,f) \) in \( \text{FLOW}(H), \text{MULT-FLOW}(H) \) of the NETWORK\((d, T)\) program which is such that:
  - \( c.F = p.d; \)
  - there does not exist any coalition subset \( A \subset X \) and any pair \( (p', (d', T')) \) (price vector, production pair) = (demand vector, QoS vector) such that:
    - \( (p', T')_A < (p, T)_A; d'_A \geq d_A; \)
    - \( p'_{X-A} = d'_{X-A} = 0; T'_{X-A} = 0; \)
    - \( d'_A = D(p', T')_A; \)
    - \( W(d', T') = \text{Optimal Value of the Program NETWORK}(d', T') = d'.p'. \)

We say that our Network Design Multicriterion Game is regular if any function \( D_x \) is strictly positive and if for any value \( k \geq 0 \), every quantity \( (p_x + k.T_x).D_x(p_x, T_x) \) is a strictly increasing function of the pair \( (p_x, T_x) \).

Then we get:

**Theorem 2.** If the above Network Design Multicriterion Game is regular then its core is non-empty.

**Proof of Theorem 2.** Theorem 2 can be proved through techniques which are the same as those which we used in order to prove Theorem 1. Given some positive multicriterion coefficient \( k > 0 \) together with some demand vector \( d \in \text{DEM}(X) \), let us first introduce the following auxiliary program NETWORK\(1(d, k)\):

**NETWORK1\((d, k)\):** [Find a flow \( F \geq 0 \) and a multicommodity flow \( f = f(x), x \in X \), both defined on \( H \), such that:

- every flow \( f(x) \) represents the routing of the demand \( d_x \) from \( f_x \) to \( s_x \);
- for any \( e \in U \), \( F_e \geq \Sigma_{x \in H} f(x)_e \);
- for any \( e \notin U \), \( F_e = 0; \)
- and which minimizes the quantity \( c.F + k.t.(\Sigma_{x \in X} f(x)) \), where \( c \) and \( t \) are, respectively, the financial cost vector and the routing cost vector which were defined at the beginning of Section 4.]

We set \( V(d, k) = \text{Optimal Value of the program NETWORK1}(d, k) \).

For any coalition \( A \subset X \), we set: \( V(d, k)^A = \text{optimal value of the program NETWORK1}(d, k)^A \) which is the restriction of the program \( \text{NETWORK}(d, k) \) to the variables and constraints which involve the players of \( A \).

Doing this allows to define a cooperative game \( f-\text{NETWORK}(d, k) = (X, V(d, k)) \), with player set \( X \) and characteristic (cost) function \( V(d, k) \).

**Lemma 1.** For any strictly positive value of \( k \), the function \( V \) which, with any demand vector \( d \in \text{DEM}(X) \), associates the value \( V(d, k) \), is continuous, and increasing. Besides, there exists \( \lambda \geq 0 \) such that for any demand vector \( d \geq 0 \) in \( \text{DEM}(X) \), we have \( V(d, k) \leq \lambda.N(d) \), where \( N \) is the Euclidean norm.
Proof of Lemma 1. The fact that \( V \) is continuous and increasing is obvious. In order to get the part of the Lemma which is related to the coefficient \( \lambda \), let us consider some demand vector \( d \) in \( \text{DEM}(X) \) and some coefficient value \( k \), and let us denote by \( \Gamma^* \) the polyhedron defined by the dual program of \( \text{NETWORK1}(d, k) \).

Let us denote by \( S \) the vertex set of \( \Gamma^* \). Since the program \( \text{NETWORK1}(d, k) \) is bounded, any value \( V(d, k) \) may be written (duality) \( V(d, k) = \sup_{x \in S} d^T x \), where \( d^* \) is the bound vector of the program \( \text{NETWORK1}(d, k) \). We notice that this bound vector \( d^* \) is the image of \( d \) through some linear function and we get the result. \( \text{End-Lemma.} \)

In order to keep on with the proof of Theorem 2, we consider now some demand vector \( d \) in \( \text{DEM}(X) \), some coefficient value \( k \), together with some pair \( (F, f) \) in \( \text{FLOW}(H) \).\( \text{MULTI-FLOW}(H) \) which is an optimal solution of the linear program \( \text{NETWORK1}(d, k) \). We suppose that the constraint matrices which express the flow conservation rules inside the \( \text{NETWORK1}(d, k) \) program are written in such a way that the bound vector \( d^* \) respects the following convention:

\[-d^*_{Ax} = 1; d^*_{Ax} = -1.\]

Then we suppose that the demand vector \( d \) is strictly positive and we set, for any player \( x \in X \):

\[-T_x = (t.f(x))/d_x = \text{Routing QoS cost of } f \text{ for the player } x; \]

\[-\text{We say that } T = (T_x, x \in X) \text{ is the Routing QoS Cost Vector associated with } f. \]

Then we may assert:

Lemma 2. Let \( u = (u_z, z \in Z) \), \( v = (v_z, z \in Z, x \in X) \), \( w = (w_e, e \in U) \geq 0 \) be some optimal solution of the dual program \( \text{NETWORK1}^*(X) \) of the network \( \text{NETWORK1}(d, k) \). Then for any \( x \in X \), we have \( k.T_x \leq d_x.(v_{Ax} - u_{Ax}). \)

Proof of Lemma 2. The vector \( v \) may be chosen in such a way that for any \( x \), \( v_{Ax} = 0 \). Let us consider some player \( x_o \in X \), together with some positive number \( \delta < d_{oA} \). We know that \( v_{oA}, \delta \geq V(d, k) - V(d', k) \), where the demand vector \( d' \) is obtained from \( d \) by subtracting \( \delta \) from \( d_{oA} \) (subdifferentiability property of the dual solution of the linear program \( \text{NETWORK1}(d, k) \)). But we get a feasible solution \( (F, f) \) in \( \text{FLOW}(H) \).\( \text{MULTI-FLOW}(H) \) of the linear program \( \text{NETWORK1}(d', k) \) by removing some amount \( \delta \) from the flow vector \( f(x_o) \), in such a way that \( k.(\Sigma_{x \in X} f(x)) - k.T \geq V(d', k) - k.\delta.T_{xo} \). We deduce that \( V(d', k) \leq V(d, k) - k.\delta.T_{xo} \) and the result. \( \text{End-Lemma.} \)

Lemma 3. Given some optimal solution \( (F, f) \) in \( \text{FLOW}(H) \).\( \text{MULTI-FLOW}(H) \) of the linear program \( \text{NETWORK1}(d, k) \), and \( T = (T_x = t.f(x))/d_x; x \in X \) the related QoS Vector. Then there exists some unit price vector \( p \geq 0 \), such that the vector \( q = (d_x(p_x + k.T_x), x \in X) \) is in the core of the cooperative game \( J\text{-NETWORK}(d, k) \).

Proof of Lemma 3. Let us consider, as in Lemma 2, \( u = (u_z, z \in Z) \), \( v = (v_z, z \in Z, x \in X) \), \( w = (w_e, e \in A) \geq 0 \), which define some optimal solution of the dual program \( \text{NETWORK1}^*(X) \) of the linear program \( \text{NETWORK1}(d, k) \). We may set, for any player \( x \in X \), \( p_x = v^X_1 \cdot k.T_x \) where \( v^X_1 \) is the \( Z \) indexed vector which is equal to 1 in \( oA \), to \(-1 \) in \( s_x \) and \( \od \) elsewhere. Lemma 2 tells us that the vector \( \delta = (p_x, x \in X) \geq 0 \). Therefore, this vector \( \delta \) may be considered as a unit price vector in \( \text{U-PRICE}(X) \). If \( A \subset X \) represents some coalition, then we denote by \( \text{NETWORK1}(d, k)^A \) the restriction of the program \( \text{NETWORK1}(d, k) \) to the variables and constraints which involve the players of \( A \). Then we see that the restrictions of \( u, v \) and \( w \) to the constraints of the linear program \( \text{NETWORK1}(d, k) \) which only involve the players of \( A \), define a feasible solution of the dual program of the network \( \text{NETWORK1}(d, k)^A \). We deduce the following relations:

\[-\Sigma_{x \in A} d_x(p_x + k.T_x) \leq \text{Optimal Value } V(d, k)^A \text{ of the program } \text{NETWORK1}(d, k)^A; \]

\[-\Sigma_{x \in A} d_x(p_x + k.T_x) = \text{Optimal Value } V(d, k)^A \text{ of the program } \text{NETWORK1}(d, k)^A. \]

As a matter of fact, relations (E6) mean that the vector \( p + k.d.T \) is an imputation vector and belongs to the core of the cooperative game \( J\text{-NETWORK}(d, k) \) related to the linear program \( \text{NETWORK1}(d, k) \). \( \text{End-Lemma.} \)

End of the Proof of Theorem 2

Let us denote by \( \text{Core}(d, k) \) the set of the pairs \( (p, T) \geq 0 \) which are such that:

\[-p \text{ in } \text{U-PRICE}(X) \text{ and } T \text{ in } \text{QS}(X); \]

\[-\text{there exists some vector } q \text{ in the core of the cooperative game } J\text{-NETWORK}(d, k) \text{ and some optimal solution } (F, f) \text{ of the linear program } \text{NETWORK1}(d, k), \text{ such that } q \text{ may be written:} \]

\[-q = (d_x(p_x + k.T_x), x \in X), \]

\[-\text{where } T = T_x = (t.f(x))/d_x; x \in X \text{ is the routing QoS Cost Vector related to } f. \]

Then, the end of the proof of Theorem 2 comes as a mere copy of the proof of Theorem 1. While \( k \) remains fixed, we first consider the domain \( A \) defined by: \( A = \{ (p, T) \geq 0 \text{ in } \text{U-PRICE}(X), \text{QS}(X) \text{ such that } \sup_{x \in X} (p_x + k.d_x.T_x) \leq \lambda \}, \) and the multiapplication \( \Gamma' \) which to any pair \( (p, T) \in A \) makes correspond \( \Gamma'(p, T) = \text{Core}(D(p, T), k) \). Next we check that any pair \( (p, T) \) which is a fixed point for this multiapplication, is also in the core of the Network Design Multicriterion Game defined by the program \( \text{NETWORK1}(d, T) \) and by the demand functions \( (D_x, x \in X) \). We conclude through application to \( \Gamma' \) of the Tychonoff–Kakutani Fixed Point Theorem. \( \text{End-Theorem.} \)
Corollary: A computing scheme

An element of the Core of this Network Design Multicriterion Game may be obtained by solving, for any value \( k \), the following constraint system \( \text{CORE-NETWORK}(k) \):

\[
\{ \text{Find a unit price vector } p > 0 \text{ in } \text{U-PRICE}(X), \text{ and a production pair } (d, T) \geq 0 \text{ in } \text{P-PROD}(X), \text{ a primal (optimal) solution } (F, f) \text{ in FLOW}(H) \text{.MULT-FLOW}(H) \text{ and a dual solution } u = (u_z, z \in Z), v = (v_{z,x}, z \in Z, x \in X), w = (w_e, e \in U) \geq 0, \text{ of the NETWORK1}(d, k) \text{ linear program, such that:} \\
\bullet \ d = D(p, T); \\
\bullet \ \text{for any player } x \in X, d_x, T_x = t.f(x); \\
\bullet \ \text{for any player } x \in X, p_x = v^x.1^x - k.T_x, \text{ where } 1^x \text{ is the } Z \text{ indexed vector which is equal to } 1 \text{ in } o_{x}, \text{ to } -1 \text{ in } s_x \text{ and } 0 \text{ elsewhere.} \}
\]

Proof of the Corollary

We get it in a straightforward way by noticing that Lemma 2 allows us to reinforce the definition of the multiapplication \( \Gamma \) which is used at the end of the proof of Theorem 2, by setting, for any demand vector \( d \) and any coefficient value \( k > 0 \):

- \( \text{Core}^*(d, k) = \{ \text{the pairs } (p = \text{unit price vector}, T = \text{routing QoS cost vector}) \geq 0 \text{ which are such that there exists a primal solution } (F, f) \text{ in FLOW}(H) \text{.MULT-FLOW}(H) \text{ and a dual solution } u = (u_z, z \in Z), v = (v_{z,x}, z \in Z, x \in X), w = (w_e, e \in U) \geq 0 \text{ of the linear program NETWORK1}(d, k) \text{ such that:} \\
\bullet \ T = T_x = (t.f(x))/d_x, x \in X; \\
\bullet \ \text{for any player } x \in X, p_x = v^x.1^x - k.T_x, \text{ where } 1^x \text{ is the } Z \text{ indexed vector which is equal to } 1 \text{ in } o_{x}, \text{ to } -1 \text{ in } s_x \text{ and } 0 \text{ elsewhere; } \\
\bullet \ \Gamma^*(d, k) = \text{Core}^*(D(p, T), k) \}
\]

while applying to the multiapplication \( \Gamma^* \) the same fixed point argument as in the proof of Theorem 2 in order to conclude.

End-Corollary.

Impact of the parameter \( k \) on the result produced by the Resolution of CORE-NETWORK(k)

As a matter of fact, we have been able to prove the existence of an element in the core of the Network Design Multicriterion Game related to the linear program NETWORK(d, T) by using the auxiliary program NETWORK1(d, k). We checked that, for any \( k > 0 \), computing some solution of the constraint system CORE-NETWORK(k) yields an element of this core. Now we may ask ourselves about the influence of the parameter \( k \) on the quality of this solution. More specifically we want to know more about the relation between the value of \( k \) and the global activity level supported by the network, which is induced by the resolution of the program CORE-NETWORK(k), and which may be summarized by the global demand \( \Sigma_{x \in X} d_x \).

In order to deal with this point, we present here a numerical experiment on the constraint system CORE-NETWORK(k), which has been performed on a small network \( G = (Z, E) \) with 12 vertices and 30 arcs. The player set \( X \) contains 10 elements.

Cost vectors \( c \) and \( t \) are satisfying the following properties:

- the mean value of \( c \) is equal to 1;
- the mean value of \( t \) for the arcs of \( U \) is equal to 2;
- the mean value of \( t \) of the arcs of \( E - U \) is equal to 1/2.

The demand function \( d_x \) can be expressed as function \( D_x(p_x, T_x) = D_x^* \Phi(p_x) \Pi(T_x) \) where the quantities \( D_x^* \) are fixed coefficient with mean value equal to 1, and where \( \Phi \) and \( \Pi \) are piecewise linear decreasing functions.

Then we observe an evolution of the quantity \( GD = \text{GLOBAL-DEMAND}(k) = \Sigma_{x \in X} d_x \) as a function of the parameter \( k \) as follows:

<table>
<thead>
<tr>
<th>( k )</th>
<th>0.2</th>
<th>0.4</th>
<th>0.6</th>
<th>0.8</th>
<th>1.0</th>
<th>1.2</th>
<th>1.4</th>
<th>1.6</th>
<th>1.8</th>
<th>2.0</th>
<th>2.2</th>
<th>2.4</th>
</tr>
</thead>
<tbody>
<tr>
<td>GD</td>
<td>2.5</td>
<td>2.6</td>
<td>2.8</td>
<td>3.1</td>
<td>3.5</td>
<td>3.9</td>
<td>4.4</td>
<td>4.8</td>
<td>4.9</td>
<td>4.6</td>
<td>4.4</td>
<td>4.3</td>
</tr>
</tbody>
</table>

Those results make appear that the way \( k \) is chosen has a strong influence on the global utility of the network. If \( k \) is too small, then prices are lowered at the expense of the QoS (Quality of Service) of the transportation network defined by the flow \( F \), and consequently, the global demand is rather low. If \( k \) is very large, then the QoS of the network defined by the flow \( F \) is very good, but the global demand is lowered by excessively high prices. A convenient value of \( k \) seems to exists here close to \( k = 1.7 \), which makes mean values \( \Phi(p_x) \) and \( \Pi(T_x) \) be rather close and which allows the \( \text{GLOBAL-DEMAND}(k) \) quantity to reach a top.

5. Conclusion

We just proposed here a model which aims at providing a deeper understanding of pricing mechanisms and proposing some new tools for the management of related decision problems. This model involves the Cooperative Game framework and tries to take into account the way users are likely to react to tariff policies. Still, one must keep in mind that Pricing remains a very hard issue, mainly due to essential difficulties related to cost and demand measurement.

As for the models and the theoretical issues which we discussed here, several points might deserve a deeper study. For instance, what could be told about an eventual link between pricing according to the above model and doing it according to
a classical bilevel optimization scheme? What about the structure of the Core of the games which we studied here and about the cases when this core is empty? Also, how should we proceed in order to take into account the existence of several well identified operators, whose relationship includes at the same time competition and partnership? How should we proceed in order to model the true segmentation of the market, which involves three layers, respectively, made with a small number of infrastructure operators, with a large number of service operators, and with users?

References