Technical Companion for the Paper ”Single and Multi-settlement Approaches to Market Clearing Mechanisms under Demand Uncertainty”

Javad Khazaei, Golbon Zakeri, Shmuel Oren

June 4, 2013

1 Computations for proposition 3.4

In[1]:= $c_{s\_} = \frac{Y_s\, cB - cA}{Z\, cB + 1};$

$p_{s\_} = \frac{(Y_s + Z\, cA)}{Z\, cB + 1};$

$y_{i\_,s\_} = p_s B_i - A_i;$

$Q = \frac{EY\, cB - cA}{Z\, cB + 1};$

$f = \frac{(EY + Z\, cA)}{Z\, cB + 1};$

$q_{i\_} = f B_i - A_i;$

$cA = A_i + A_{-i};$

$cB = n B_i;$

$u_{i\_} = \text{FullSimplify}\left[\sum_{s=1}^{S} \theta_s \left(p_s\, y_{i,s} - \left(\alpha\, y_{i,s} + \frac{\beta}{2} y_{i,s}^2 + \frac{\delta}{2}\left(y_{i,s} - q_i\right)^2\right)\right)\right];$

In[11]:= $u_i = \text{FullSimplify}[u_i]$

Out[11]= $$\sum_{s=1}^{S}\left(-\alpha\left(-A_i + \frac{B_i\,(Z\,(A_{-i} + A_i) + Y_s)}{1 + n\,Z\,B_i}\right) + \frac{(Z\,(A_{-i} + A_i) + Y_s)\left(-A_i + \frac{B_i\,(Z\,(A_{-i}+A_i)+Y_s)}{1+n\,Z\,B_i}\right)}{1 + n\,Z\,B_i} - \frac{1}{2}\beta\left(-A_i + \frac{B_i\,(Z\,(A_{-i} + A_i) + Y_s)}{1 + n\,Z\,B_i}\right)^2 - \frac{1}{2}\delta\left(-\frac{(EY + Z\,(A_{-i} + A_i))\,B_i}{1 + n\,Z\,B_i} + \frac{B_i\,(Z\,(A_{-i} + A_i) + Y_s)}{1 + n\,Z\,B_i}\right)^2\right)\theta_s$$

In[12]:= $EY = \sum_{s=1}^{S} \theta_s Y_s$

Out[12]= $\sum_{s=1}^{S} Y_s\, \theta_s$

$\text{FullSimplify}[D[u_i, A_i, A_i]]$

Out[23]= $$-\frac{(1 + (-1 + n)\,Z\,B_i)\,(2\,Z + \beta + (-1 + n)\,Z\,\beta\,B_i)}{(1 + n\,Z\,B_i)^2}$$

2 Proposition 3.5

Proposition. *The equilibrium pre-dispatch and spot production quantities of the firms in the two settlement market are non-negative, i.e.*

$$q_i \geq 0 \qquad \forall i,$$
$$y_{i,s} \geq 0 \qquad \forall i, s.$$

Proof. To prove the proposition, we first show the equilibrium price intercept of the supply function of generators (i.e. $a_i = \frac{A_i}{B_i}$) is less than the price intercept of the demand function (i.e. Y and Y_s). Then, we show this property entails the non-negativity of equilibrium quantities.

Substituting A_i and B_i from proposition 3.4 into $a_i = \frac{A_i}{B_i}$, and then taking the derivative of a_i with respect to Z, we achieve

$$\frac{\partial a_i}{\partial Z} = \frac{2\delta\left((n-2)^2 Z + 2k + n(\beta+\delta+k)\right)(Y-\alpha)}{k((n+2)Z+\beta-\delta+k)^2},$$

where, $k = \sqrt{(n-2)^2 Z^2 + 2nZ(\beta+\delta) + (\beta+\delta)^2}$. Because $n \geq 2$, $Z > 0$, $\beta \geq 0$, $\delta \geq 0$, and $\alpha \leq Y$, we have

$$\frac{\partial a_i}{\partial Z} \geq 0. \tag{1}$$

On the other hand, taking the limit of a_i as Z approaches infinity, we obtain

$$\lim_{Z\to\infty} a_i = \alpha. \tag{2}$$

Equations (1) and (2) yield

$$a_i \leq \alpha.$$

This together with assumption $\alpha \leq Y_s$,$\forall s$ yields

$$a_i \leq Y_s \qquad \forall i, s. \tag{3}$$

Using $a_i = \frac{A_i}{B_i}$, we can rewrite equation (3) as

$$B_i Y_s - A_i \geq 0 \qquad \forall i, s. \tag{4}$$

Also, using the value of B_i from proposition 3.4, we can show $B_i \geq 0$. Thus, we can conclude

$$B \geq 0 \tag{5}$$

On the other hand, embedding p_s into $y_{i,s}$ from proposition 3.2, we obtain

$$y_{i,s} = \frac{B_i Y_s - A_i}{ZB+1} \qquad \forall i, s.$$

This together with equations (4) and (5) and assumption $Z > 0$ gives

$$y_{i,s} \geq 0 \qquad \forall i, s.$$

From propositions 3.1 and 3.2, we achieve $q_i = \sum_s \theta_s y_{i,s}$. As $\theta_s \geq 0$, we obtain

$$q_i \geq 0 \qquad \forall i$$

□

3 The optimal solution to ISOSP problem: proof of proposition 4.4

Proposition. *If* $(\mathbf{q}, \mathbf{x}, \mathbf{f}, \mathbf{p})$ *represents the solution of ISOSP, then we have*

$$\begin{aligned}
q_i &= \frac{(Y + ZA)B_i}{1 + ZB} - A_i && (6)\\
x_{i,s} &= \frac{(Y_s - Y)R_i}{1 + ZR} && (7)\\
f &= \frac{Y + ZA}{1 + ZB}\\
p_s &= \frac{Y + ZA}{1 + ZB} + \frac{Y_s - Y}{1 + ZR}
\end{aligned}$$

Proof. The Lagrangian function of ISOSP can be represented as follows.

$$\begin{aligned}
L \;=\; & -f\left(-Q + \sum_{i=1}^{n} q_i\right)\\
& + \sum_{s=1}^{S} \theta_s \left(-p_s \left(Q - C_s + \sum_{i=1}^{n} x_{i,s}\right)\right.\\
& \left. - Y_s C_s + \frac{Z C_s^2}{2} + \sum_{i=1}^{n} \left(\frac{1}{2} d_i x_{i,s}^2 + a_i \left(q_i + x_{i,s}\right) + \frac{1}{2} b_i \left(q_i + x_{i,s}\right)^2\right)\right)
\end{aligned}$$

Taking derivative with respect to different variables yields to the following equations.

$$\frac{dL}{dq_i} = -f + \sum_s \theta_s \left(a_i + b_i \left(q_i + x_{i,s}\right)\right) \tag{8}$$

$$\frac{dL}{dx_{i,s}} = \theta_s \left(-p_s + a_i + b_i \left(q_i + x_{i,s}\right) + d_i x_{i,s}\right) \tag{9}$$

$$\frac{dL}{dC_s} = \theta_s \left(p_s - Y_s + ZC_s\right) \tag{10}$$

$$\frac{dL}{dQ} = f - \sum_s \theta_s p_s \tag{11}$$

$$\frac{dL}{dp_s} = \theta_s \left(-Q + C_s - \sum_i x_{i,s}\right) \tag{12}$$

$$\frac{dL}{df} = Q - \sum_i q_i \tag{13}$$

The Lagrangian is evidently a convex function. Thus, for finding the solution of the stochastic program, we should set all above derivatives to zero.

From (8)

$$f = a_i + b_i q_i + \sum_s p_s x_{i,s}. \tag{14}$$

From (9) and (14)

$$p_s = f + (b_i + d_i) x_{i,s}, \tag{15}$$

and from (11)

$$f = \sum_s \theta_s p_s. \tag{16}$$

Now (14), (15) and (16) result in the following conclusion, as it is also concluded from lemma 4.1.

$$\sum_s \theta_s x_{i,s} = 0 \tag{17}$$

(14) and (17) lead to

$$f = a_i + b_i q_i. \tag{18}$$

Consequently, forward price is independent of the spot market and is resolved merely by contract quantities. Though, contract quantities are chosen by considering different possible spot scenarios.

From (10),

$$p_s = Y_s - ZC_s, \tag{19}$$

from (12),

$$C_s = Q + \sum_i x_{i,s}, \tag{20}$$

and from (13),

$$Q = \sum_i q_i \tag{21}$$

can be concluded.
(17) and (20) lead to

$$\sum_s \theta_s C_s = Q. \tag{22}$$

(16), (19) and (22) make the following conclusion.

$$f = Y - ZQ \tag{23}$$

Now from (18) and (23) we can conclude

$$q_i = \frac{Y - ZQ - a_i}{b_i}. \tag{24}$$

In consequence, from (21) and summation of q_i from (24) over all firms and by using the transformation (A_i, B_i, R_i), we obtain

$$Q = (Y - ZQ)B - A.$$

Therefore,

$$Q = \frac{YB - A}{1 + ZB}. \tag{25}$$

Now the following inference can be resulted from (24) and (25).

$$q_i = \frac{(Y + ZA)B_i}{1 + ZB} - A_i \tag{26}$$

Now let us find $x_{i,s}$. (15), (19) and (20) give

$$f + (b_i + d_i)x_{i,s} = Y_s - ZQ - Z\sum_i x_{i,s}.$$

By adding (23) to this equation following equation is resulted.

$$x_{i,s} = \frac{Y_s - Y - Z\sum_i x_{i,s}}{b_i + d_i} \tag{27}$$

Now by getting a summation from (27) and simplifying the resulted equation we achieve

$$\sum_i x_{i,s} = \frac{(Y_s - Y)R}{1 + ZR}.$$

By inserting this equation in (27), we obtain

$$x_{i,s} = \frac{(Y_s - Y)R_i}{1 + ZR}, \tag{28}$$

and from (23) and (25), first stage price can be extracted.

$$f = \frac{Y + ZA}{1 + ZB} \tag{29}$$

One observation about this equation is that contract price is independent of R, in other words, it is independent of deviating cost in the spot market.

(25), (28) and (29) determine spot price for each scenario.

$$p_s = \frac{Y + ZA}{1 + ZB} + \frac{Y_s - Y}{1 + ZR} \tag{30}$$

□

4 The equilibrium of the stochastic settlement market: proof of proposition 4.8

Proposition. *The unique symmetric equilibrium quantities of the stochastic settlement market are as follows.*

$$a_i = \frac{\alpha - Y + B_i\left(-Z(Y(n-2) - (2n-1)\alpha) + Y\beta + Z(n-1)(Zn\alpha + Y\beta)B_i\right)}{B_i\left(Z(n+1) + \beta + Y(n-1)(Zn + \beta)B_i\right)} \tag{31}$$

$$d_i = \max\{0, \frac{-Z(n-2) + \beta + \delta + \sqrt{Z^2(n-2)^2 + 2Zn(\beta + \delta) + (\beta + \delta)^2}}{2} - \frac{1}{B_i}\} \tag{32}$$

Proof. As we assumed fixed quantity for all B_i, we have

$$B = nB_i.$$

To find a symmetric equilibrium, we can use

$$A_{-i} = (n-1)A_i,$$

and

$$R_{-i} = (n-1)R_i.$$

By putting these equations in the best response functions (from theorem 4.7) and solving the resulted equations with respect to A_i and R_i, following equilibrium equations is resulted.

$$A_i = \frac{\alpha - Y + B_i\left(-Z(Y(n-2) - (2n-1)\alpha) + Y\beta + Z(n-1)(Zn\alpha + Y\beta)B_i\right)}{Z(n+1) + \beta + Y(n-1)(Zn + \beta)B_i}$$

$$R_i = \min\{B_i, \frac{2}{-Z(n-2) + \beta + \delta + \sqrt{Z^2(n-2)^2 + 2Zn(\beta+\delta) + (\beta+\delta)^2}}\}$$

Let us see why equation (32) implies a true equilibrium quantity. Let $\hat{R}_i = \frac{2}{-Z(n-2)+\beta+\delta+\sqrt{Z^2(n-2)^2+2Zn(\beta+\delta)+(\beta+\delta)^2}}$. If $\hat{R}_i \leq B_i$, it satisfies the best response function for R_i. When $\hat{R}_i > B_i$, we need to show $\frac{1+Z(n-1)B_i}{Z+\beta+\delta+Z(n-1)(\beta+\delta)B_i} \geq B_i$. It means when the other generators j have chosen $R_j = B_j$, the best response for the firm i is also to choose $R_i = B_i$. Note that B_i is a fixed quantity chosen by the ISO, Thus, $B_j = B_i$.

Define $f(x) = \frac{1+Z(n-1)x}{Z+\beta+\delta+Z(n-1)(\beta+\delta)x} - x$. We can easily show that $f(x)$ is a concave function for $x \geq 0$:

$$f^{"}(x) = -\frac{2Z^3(n-1)^2(\beta+\delta)}{(Z+\beta+\delta+Z(n-1)(\beta+\delta)x)^3} < 0$$

Also $f(0) = \frac{1}{Z+\beta+\delta} > 0$ and $f(\hat{R}_i) = 0$. Thus for $0 < B_i < \hat{R}_i$, and by considering concavity of $f(x)$,

$$f(B_i) \geq 0.$$

Therefore,

$$\frac{1+Z(n-1)B_i}{Z+\beta+\delta+Z(n-1)(\beta+\delta)B_i} \geq B_i.$$

□

5 Stochastic settlement yields non-negative equilibria: proof of theorem 4.9

Theorem. *If* $(\mathbf{q}^*, \mathbf{x}^*)$ *represents the equilibrium of the stochastic settlement market, following equations always hold.*

$$\forall i, s: \; q_i^* + x_{i,s}^* \geq 0$$

$$\forall i: \; q_i^* \geq 0$$

Proof. From (26) and (28), the following equation can be resulted.

$$y_{i,s} = q_i^* + x_{i,s}^* = \frac{(Y+ZA)B_i}{1+ZB} - A_i + \frac{(Y_s - Y)R_i}{(1+ZR)}$$

It is obvious that if $y_{i,s}$ is non-negative for the scenario that has the lowest Y_s, it is non-negative for the other scenarios as well. Thus, we prove this only for the scenario s' for which we have $Y_{s'} \le Y_s$ for all s. If we assume having at least two different scenarios with positive probabilities, we have

$$Y_{s'} < Y. \tag{33}$$

Let us first define $\hat{R}_i = \frac{2}{-Z(n-2)+\beta+\delta+\sqrt{Z^2(n-2)^2+2Zn(\beta+\delta)+(\beta+\delta)^2}}$, as we defined in the proof of proposition 4.8. Now consider $y^{'}_{i,s'} = \min_{\alpha,\delta} y_{i,s'}$. Obviously if we prove that $y^{'}_{i,s'}$ is non-negative, we have also proven the non-negativity of $y_{i,s}$. $y_{i,s}$ can be divided to two separate functions of α and δ, such that

$$\frac{dy_{i,s'}}{d\delta} = \begin{cases} if\ \hat{R}_i \le B_i: \\ \frac{2\left(Zn+\beta+\delta+\sqrt{Z^2(n-2)^2+2Zn(\beta+\delta)+(\beta+\delta)^2}\right)(Y-Y_s)}{\sqrt{Z^2(n-2)^2+2Zn(\beta+\delta)+(\beta+\delta)^2}\left(Z(n+2)+\beta+\delta+\sqrt{Z^2(n-2)^2+2Zn(\beta+\delta)+(\beta+\delta)^2}\right)^2} \\ \\ Otherwise: \\ 0 \end{cases}$$

$$\frac{dy_{i,s'}}{d\alpha} = -\frac{1+ZB_1(n-1)}{Z(n+1)+\beta+ZB_1(n-1)(Zn+\beta)}$$

The parameters Z, β, and δ are non-negative. Thus, from (33), we can conclude

$$\frac{dy_{i,s}}{d\delta} \ge 0,$$

$$\frac{dy_{i,s}}{d\alpha} \le 0.$$

Consequently, $\delta = 0$ and $\alpha = Y_{s'}$ minimize $y_{i,s'}$. Note that we have assumed in this chapter, that y-intercept of cost function (α) is less than y-intercept of the demand scenarios $(Y_{s'})$. Thus, we prove that $y^{'}_{i,s'} = y_{i,s'}(\delta = 0, \alpha = Y_{s'})$ gets non-negative values.

When $\delta = 0$, at $\hat{\beta} = \frac{1+ZB_i(n-2)}{B_i(1+ZB_i(n-1))}$, we have $\hat{R}_i = B_i$. By applying the fact that $\hat{R}_i$ is a decreasing function of β, we can conclude,

$$R_i = \begin{cases} B_i & \beta < \hat{\beta} \\ \hat{R}_i & \beta \ge \hat{\beta} \end{cases}$$

and

$$y^{'}_{i,s'} = \begin{cases} \frac{(Y+ZA)B_i}{1+ZB} - A_i + \frac{(Y_s-Y)B_i}{(1+ZB)} & \beta < \hat{\beta} \\ \frac{(Y+ZA)B_i}{1+ZB} - A_i + \frac{(Y_s-Y)\hat{R}_i}{(1+Z\hat{R})} & \beta \ge \hat{\beta} \end{cases}$$

We can also show that equation $y^{'}_{i,s'} = 0$ only holds at $\beta = \hat{\beta}$. In addition, $y^{'}_{i,s'}$ is a continuous function. These mean $y^{'}_{i,s'}$ is either entirely positive or entirely negative in each of $[0, \hat{\beta}]$ or $[\hat{\beta}, \infty)$. Firstly, we prove that it is positive in $[0, \hat{\beta}]$.

We see that $\frac{dy'_{i,s'}}{dA_i} < 0$. On the other hand,

$$\frac{dA_i}{d\beta} = \frac{(Y-\alpha)\left(1+Z(n-1)B_i\right)^2\left(1+ZnB_i\right)}{\left(Z(n+1)+\delta+Z(n-1)(Zn+\beta)B_i\right)^2} \geq 0$$

Therefore, for $\beta < \hat{\beta}$, $\frac{dy'_{i,s'}}{d\beta} = \frac{dy'_{i,s'}}{dA_i}\frac{dA_i}{d\beta}$ is not positive. It means $y_{i,s'}$ is a non-increasing function of β in this interval. Considering the fact that $y'_{i,s'}(\hat{\beta}) = 0$, we can conclude

$$y'_{i,s'} \geq 0 \;\text{ if }\; \beta \leq \hat{\beta}. \tag{34}$$

Right derivative of $y'_{i,s'}$ at $\hat{\beta}$ also has a positive value of $\frac{Z^2(Y-Y_s)B_i(n-1)(1+Z(n-1)B_i)(1+ZnB_i)^2}{\sqrt{Z^2(n-2)^2+2Zn\beta+\delta^2}(Z(n+1)+\delta+ZB_i(-\beta+2n(Zn+\beta)+Z(n-1)n(Zn+\beta)B_i))^2}$.

If we add this to the facts that $y'_{i,s'}(\hat{T}) = 0$ and $y'_{i,s'}$ is either entirely non-negative or entirely non-positive for $\beta > \hat{\beta}$, we can conclude that

$$y'_{i,s'} \geq 0\,\text{if}\,\beta \geq \hat{\beta} \tag{35}$$

(34) and (35) can be gathered to conclude

$$y'_{i,s'} \geq 0.$$

Therefore,

$$y_{i,s} = q_i^* + x_{i,s}^* \geq 0$$

We know from Lemma 4.1 that $x_{i,s}^*$ is non-positive for at least one-scenario. Thus,

$$q_i^* \geq 0$$

□

6 Equilibrium of the stochastic settlement mechanism with non-negativity constraints: theorem 4.11

6.1 SP clearing problem with non-negativity constraints

The SP clearing problem with non-negativity constraints is

ISOSP :

$$
\begin{aligned}
\min z \quad &= \sum_{s=1}^{S} \theta_s \left(\sum_{i=1}^{n} \left[a_i(q_i + x_{i,s}) + \frac{b}{2}(q_i + x_{i,s})^2 + \frac{d_i}{2} x_{i,s}^2 \right] - (Y_s C_s - \frac{Z}{2} C_s^2) \right) \\
s.t. \quad & \sum_i q_i - Q = 0 \\
& Q + \sum_i x_{i,s} - C_s = 0 \quad \forall s \in \{1, \dots, S\} \\
& q_i + x_{i,s} \geq 0 \quad \forall i, s \in \{1, \dots, S\}
\end{aligned}
$$

ISOSP is a convex optimization problem as the objective function of ISOSP is a convex function, and its constraints are linear. Therefore, solving the KKT conditions of this problem is equivalent to solving ISOSP.

6.1.1 KKT of ISOSP

To find the KKT conditions we can use the Lagrangian function

$$
\begin{aligned}
L = \sum_{s=1}^{S} \Bigg(\theta_s \Bigg(\sum_{i=1}^{n} \left(a_i \left(x_{i,s} + q_i \right) + \frac{b}{2} \left(x_{i,s} + q_i \right)^2 + \frac{d_i}{2} x_{i,s}^2 \right) \\
- \left(C_s Y_s - \frac{Z C_s^2}{2} \right) + p_s \left(Q + \sum_{i=1}^{n} x_{i,s} - C_s \right) \Bigg) \\
- \sum_{i=1}^{n} e_{i,s} \left(x_{i,s} + q_i \right) \Bigg) - f \left(\sum_{i=1}^{n} q_i - Q \right).
\end{aligned}
$$

To produce the building blocks of the KKT condition, we can use the partial derivations of L with respect to the decision variables.

$$\frac{dL}{dq_i} = -f - \sum_{s=1}^{S} e_{i,s} + (a_i + bq_i) + b\sum_{s=1}^{S} \theta_s x_{i,s}$$
$$\frac{dL}{dx_{i,s}} = -e_{i,s} + \theta_s\left(-p_s + a_i + bq_i + (b + d_i)x_{i,s}\right)$$
$$\frac{dL}{dC_s} = \left(p_s + ZC_s - Y_s\right)\theta_s$$
$$\frac{dL}{dQ} = f - \sum_{s} \theta_s p_s$$
$$\frac{dL}{dp_s} = \theta_s\left(C_s - \left(Q + \sum_{i=1}^{n} x_{i,s}\right)\right)$$
$$\frac{dL}{df} = Q - \sum_{i=1}^{n} q_i$$
$$\frac{dL}{de_{i,s}} = -q_i - x_{i,s}$$

Thus, KKT of this problem can be represented as

$$-f - \sum_{s=1}^{S} e_{i,s} + (a_i + bq_i) + b\sum_{s=1}^{S} \theta_s x_{i,s} = 0 \qquad \forall i \in \{1,\dots,n\} \qquad [C1]$$
$$Q = \sum_{i=1}^{n} q_i \qquad [C2]$$
$$C_s = \left(Q + \sum_{i=1}^{n} x_{i,s}\right) \qquad \forall s \in \{1,\dots,S\} \qquad [C3]$$
$$p_s = (Y_s - ZC_s) \qquad \forall s \in \{1,\dots,S\} \qquad [C4]$$
$$f = \sum_{s=1}^{S} \theta_s p_s \qquad [C5]$$
$$e_{i,s} = \theta_s\left(-p_s + a_i + bq_i + (b + d_i)x_{i,s}\right) \qquad \forall i \in \{1,\dots,n\},\ \forall s \in \{1,\dots,S\} \qquad [C6]$$
$$e_{i,s}(q_i + x_{i,s}) = 0 \qquad \forall i \in \{1,\dots,n\},\ \forall s \in \{1,\dots,S\} \qquad [C7]$$
$$e_{i,s} \geq 0 \qquad \forall i \in \{1,\dots,n\},\ \forall s \in \{1,\dots,S\} \qquad [C8]$$
$$q_i + x_{i,s} \geq 0 \qquad \forall i \in \{1,\dots,n\},\ \forall s \in \{1,\dots,S\}. \qquad [C9]$$

If we replace the value of f and $e_{i,s}$ from [C5] and [C6] into [C1], constraint [C1] can be replaced with $\sum_{s=1}^{S} \theta_s x_{i,s} = 0$.

6.1.2 Firms' optimisation problem

Problem WNN[j] represents the optimization problem solved by firm j to maximize its profit, subject to KKT conditions of ISO's optimization problem.

WNN[j]:

$$\max u_j = \sum_{s=1}^{S} \theta_s \Bigg(p_s(q_j + x_{j,s}) - \left(\alpha_j \left(q_j + x_{j,s} \right) + \frac{\beta_j}{2} \left(q_j + x_{j,s} \right)^2 + \frac{\delta_j}{2} {x_{j,s}}^2 \right) \Bigg)$$

$$\text{s.t.} \quad \sum_{s=1}^{S} \theta_s x_{i,s} = 0 \qquad \forall i \in \{1, \ldots, n\} \quad \text{[C1]}$$

$$Q = \sum_{i=1}^{n} q_i \qquad \text{[C2]}$$

$$C_s = \left(Q + \sum_{i=1}^{n} x_{i,s} \right) \qquad \forall s \in \{1, \ldots, S\} \quad \text{[C3]}$$

$$p_s = (Y_s - ZC_s) \qquad \forall s \in \{1, \ldots, S\} \quad \text{[C4]}$$

$$f = \sum_{s=1}^{S} \theta_s p_s \qquad \text{[C5]}$$

$$e_{i,s} = \theta_s \left(-p_s + a_i + bq_i + (b + d_i) x_{i,s} \right) \qquad \forall i \in \{1, \ldots, n\} \quad \forall s \in \{1, \ldots, S\} \quad \text{[C6]}$$

$$e_{i,s}(q_i + x_{i,s}) = 0 \qquad \forall i \in \{1, \ldots, n\} \quad \forall s \in \{1, \ldots, S\} \quad \text{[C7]}$$

$$e_{i,s} \geq 0 \qquad \forall i \in \{1, \ldots, n\} \quad \forall s \in \{1, \ldots, S\} \quad \text{[C8]}$$

$$q_i + x_{i,s} \geq 0 \qquad \forall i \in \{1, \ldots, n\} \quad \forall s \in \{1, \ldots, S\} \quad \text{[C9]}$$

To make the optimization problem look simpler, we can replace the values of Q, C_s, and f from [C2], [C3], and [C5] in the other equations. This simplifies

WNN to the following shape.

WNN[j]:

$$\max\ u_j = \sum_{s=1}^{S} \theta_s \Bigg(p_s(q_j + x_{j,s}) - \left(\alpha_j\,(q_j + x_{j,s}) + \frac{\beta_j}{2}\ (q_j + x_{j,s})^{\,2} + \frac{\delta_j}{2} {x_{j,s}}^2 \right) \Bigg)$$

$$\begin{aligned}
\text{s.t.}\quad & \sum_{s=1}^{S} \theta_s x_{i,s} = 0 && \forall i \in \{1,\dots,n\} && [\text{C1}]\\
& p_s = Y_s - Z\left(\sum_{h=1}^{n} q_h + \sum_{h=1}^{n} x_{h,s}\right) && \forall s \in \{1,\dots,S\} && [\text{C4}]\\
& e_{i,s} = -\theta_s\,(-p_s + a_i + bq_i + (b+d_i)x_{i,s}) && \forall i \in \{1,\dots,n\} && [\text{C6}]\\
& && \forall s \in \{1,\dots,S\} &&\\
& e_{i,s}(q_i + x_{i,s}) = 0 && \forall i \in \{1,\dots,n\} && [\text{C7}]\\
& && \forall s \in \{1,\dots,S\} &&\\
& e_{i,s} \geq 0 && \forall i \in \{1,\dots,n\} && [\text{C8}]\\
& && \forall s \in \{1,\dots,S\} &&\\
& q_i + x_{i,s} \geq 0 && \forall i \in \{1,\dots,n\} && [\text{C9}]\\
& && \forall s \in \{1,\dots,S\} &&
\end{aligned}$$

With a similar process, the optimization problem of firm j in a stochastic market clearing mechanism without non-negativity constraints can be found as

WONN[j]:

$$\max\ u_j = \sum_{s=1}^{S} \theta_s \Bigg(p_s(q_j + x_{j,s}) - \left(\alpha_j\,(q_j + x_{j,s}) + \frac{\beta_j}{2}\ (q_j + x_{j,s})^{\,2} + \frac{\delta_j}{2} {x_{j,s}}^2 \right) \Bigg)$$

$$\begin{aligned}
\text{s.t.}\quad & \sum_{s=1}^{S} \theta_s x_{i,s} = 0 && \forall i \in \{1,\dots,n\} && [\text{C1}]\\
& p_s = Y_s - Z\left(\sum_{h=1}^{n} q_h + \sum_{h=1}^{n} x_{h,s}\right) && \forall s \in \{1,\dots,S\} && [\text{C4}]\\
& e_{i,s} = \theta_s\,(-p_s + a_i + bq_i + (b+d_i)x_{i,s}) && \forall i \in \{1,\dots,n\} && [\text{C6}]\\
& e_{i,s} = 0 && \forall i \in \{1,\dots,n\} && [\text{C11}]\\
& && \forall s \in \{1,\dots,S\}. &&
\end{aligned}$$

Also, we introduce a relaxation to WNN, which we use later in proofs of our theorems. We eliminate constraint [C7]: $e_{i,s}(q_i + x_{i,s}) = 0$, and limit the constraint [C9]: $\forall i, q_i + x_{i,s} \geq 0$ to the optimizer generator j to obtain a relaxation problem

RWNN:

$$\max\ u_j = \sum_{s=1}^{S} \theta_s \Bigg(p_s(q_j + x_{j,s}) - \left(\alpha_j \left(q_j + x_{j,s}\right) + \frac{\beta_j}{2} \ \left(q_j + x_{j,s}\right)^2 + \frac{\delta_j}{2} {x_{j,s}}^2 \right) \Bigg)$$

$$\begin{aligned}
\text{s.t.} \quad & \sum_{s=1}^{S} \theta_s x_{i,s} = 0 && \forall i \in \{1,\ldots,n\} && \text{[C1]}\\
& p_s = Y_s - Z \left(\sum_{h=1}^{n} q_h + \sum_{h=1}^{n} x_{h,s} \right) && \forall s \in \{1,\ldots,S\} && \text{[C4]}\\
& f = \sum_{s=1}^{S} \theta_s p_s && && \text{[C5]}\\
& e_{i,s} = \theta_s \left(-p_s + a_i + bq_i + (b + d_i) x_{i,s}\right) && \forall i \in \{1,\ldots,n\} && \text{[C6]}\\
& && \forall s \in \{1,\ldots,S\} && \\
& e_{i,s} \geq 0 && \forall i \in \{1,\ldots,n\} && \text{[C8]}\\
& q_j + x_{j,s} \geq 0 && \forall s \in \{1,\ldots,S\}. && \text{[C12]}
\end{aligned}$$

Now, we prove three lemmas which help us to demonstrate the final theorem.

Lemma 6.1. *If for every $i \neq j$(j is the optimizer generator), a_i and d_i has the same value, then the constraint $e_{i,s} \geq 0$ (for every $i \neq j$) in RWNN can be replaced with $e_{i,s} = 0$ without reducing the optimal value of RWNN.*

Proof. We prove the lemma by contradiction. Assume there exist a point $\nu = (a_j, d_j, \boldsymbol{q}, \boldsymbol{x}, \boldsymbol{p}, \boldsymbol{e})$ with at least one $e_{i',s'} > 0$ $(i \neq j)$ and higher objective value than any feasible solution with $\boldsymbol{e} = \boldsymbol{0}$.

Consider $\nu' = (a'_j, d'_j, \boldsymbol{q}', \boldsymbol{x}', \boldsymbol{p}', \boldsymbol{e}')$ defined as follows.

$$q'_i = \begin{cases} q_i & i = j \\ q_i + \frac{\frac{Z\sum_{h\neq j}\sum_w e_{h,w}}{Z(n-1)+b} - \sum_w e_{i,w}}{b} & i \neq j \end{cases} \tag{36}$$

$$x'_{i,s} = \begin{cases} x_{i,s} & i = j \\ x_{i,s} + \frac{\sum_w e_{i,w} - \frac{e_{i,s}}{\theta_s}}{b+d_i} - \frac{Z(\sum_{h\neq j}\sum_w e_{h,w} - \sum_{h\neq j}\frac{e_{h,s}}{\theta_s})}{(Z(n-1)+b+d_i)(b+d_i)} & i \neq j \end{cases} \tag{37}$$

$$a'_j \geq \max_s \left\{ Z\left(\sum_{h\neq j}\sum_w e_{h,w}\left(\frac{1}{Z(n-1)+b} - \frac{1}{Z(n-1)+b+d_i}\right) + \frac{\sum_{h\neq j}\frac{e_{h,s}}{\theta_s}}{Z(n-1)+b+d_i}\right) + a_j \right\} \tag{38}$$

$$d'_j = d_j \tag{39}$$

Firstly, we show this is a feasible solution.

$$\sum_s \theta_s x'_{i,s} = \begin{cases} \sum_s \theta_s x_{i,s} & i = j \\ \sum_s \theta_s x_{i,s} + \frac{\sum_w e_{i,w} - \sum_s \theta_s \frac{e_{i,s}}{\theta_s}}{b+d_i} - \frac{Z(\sum_{h\neq j}\sum_w e_{h,w} - \sum_s \theta_s \sum_{h\neq j}\frac{e_{h,s}}{\theta_s})}{(Z(n-1)+b+d_i)(b+d_i)} & i \neq j \end{cases}$$

Extra simplifications yields to

$$\forall i : \sum_s \theta_s x'_{i,s} = 0 \tag{40}$$

After substituting the value of q'_h from (36) into $\sum_{h\neq j} q'_h$ and slightly simplifying the resulted equation, we get

$$\sum_{h\neq j} q'_h = \sum_{h\neq j} q_h - \frac{\sum_{h\neq j}\sum_w e_{h,w}}{Z(n-1)+b} \tag{41}$$

The same analysis on equation (37) gives us the following equation.

$$\sum_{h\neq j} x'_{h,s} = \sum_{h\neq j} x_{h,s} + \frac{\sum_{h\neq j}\sum_w e_{h,w} - \sum_{h\neq j}\frac{e_{h,s}}{\theta_s}}{Z(n-1)+b+d_i} \tag{42}$$

p'_s can be obtained combining equations [C4], (41), and (42).

$$\begin{aligned} p'_s =& p_s - Z\left(-\frac{\sum_{h\neq j}\sum_w e_{h,w}}{Z(n-1)+b} + \frac{\sum_{h\neq j}\sum_w e_{h,w} - \sum_{h\neq j}\frac{e_{h,s}}{\theta_s}}{Z(n-1)+b+d_i}\right) \\ =& p_s + Z\left(\sum_{h\neq j}\sum_w e_{h,w}\left(\frac{1}{Z(n-1)+b} - \frac{1}{Z(n-1)+b+d_i}\right) \right. \\ &\left. + \frac{\sum_{h\neq j}\frac{e_{h,s}}{\theta_s}}{Z(n-1)+b+d_i}\right) \end{aligned} \tag{43}$$

Considering the fact that $e_{i,s}$, Z, b, and d_i have non-negative values,

$$p'_s \geq p_s \tag{44}$$

From (36), (37), (43), and [C6], $e_{i,s}$ can be obtained as follows.

$$e'_{i,s} = \begin{cases} e_{i,s} + \theta_s(-p'_s + p_s + a'_j - a_j) & i = j \\ e_{i,s} + \theta_s\Bigg(- Z \sum_{h \neq j} \sum_w e_{h,w} \left(\frac{1}{Z(n-1)+b} - \frac{1}{Z(n-1)+b+d_i}\right) & i \neq j \\ \quad -Z \frac{\sum_{h \neq j} \frac{e_{h,s}}{\theta_s}}{Z(n-1)+b+d_i} + \frac{Z \sum_{h \neq j} \sum_w e_{h,w}}{Z(n-1)+b} - \sum_w e_{i,w} & \\ \quad + \sum_w e_{i,w} - \frac{e_{i,s}}{\theta_s} - \frac{Z(\sum_{h \neq j} \sum_w e_{h,w} - \sum_{h \neq j} \frac{e_{h,s}}{\theta_s})}{Z(n-1)+b+d_i} \Bigg) & \end{cases}$$

This simplifies to

$$e'_{i,s} = \begin{cases} e'_{j,s} \geq 0 & i = j \\ 0 & i \neq j \end{cases}$$

Thus, the constraint [C8] is also satisfied. As $q'_j = q_j$, $x'_{j,s} = x_{j,s}$, and ν is a feasible solution, constraints [C12] are also fulfilled.

In sum, ν' is a feasible solution.

On the other hand, a comparison between the u'_j and u_j demonstrates that ν' gives a better objective:

$$u'_j - u_j = \sum_s \theta_s(p'_s - p_s)(q_j + x_{j,s}).$$

With $q_j + x_{j,s} \geq 0$, as concluded from [C12], and $p'_s - p_s \geq 0$ as resolved in (44)

$$u'_j \geq u_j$$

This contradicts the initial assumption, which proves the lemma. □

Lemma 6.2. *RWNN can be simplified to the following optimization problem.*

RWNN:

$$\max\ u_j = fq_j + \sum_{s=1}^{S} \theta_s (p_s - f) x_{j,s} - \left(\alpha_j q_j + \frac{\beta_j}{2} q_j^2 + \frac{\beta_j + \delta_j}{2} \sum_{s=1}^{S} \theta_s {x_{j,s}}^2 \right)$$

$$\text{s.t.} \quad \sum_{s=1}^{S} \theta_s x_{i,s} = 0 \qquad \forall i \in \{1, \ldots, n\} \qquad \textit{[C1]}$$

$$p_s = Y_s - Z \left(\sum_{h=1}^{n} q_h + \sum_{h=1}^{n} x_{h,s} \right) \qquad \forall s \in \{1, \ldots, S\} \qquad \textit{[C4]}$$

$$f = \sum_{s=1}^{S} \theta_s p_s \qquad \textit{[C5]}$$

$$e_{i,s} = \theta_s \left(-p_s + a_i + bq_i + (b + d_i) x_{i,s} \right) \qquad \forall i \in \{1, \ldots, n\} \quad \forall s \in \{1, \ldots, S\} \qquad \textit{[C6]}$$

$$e_{i,s} \geq 0 \qquad \forall i \in \{1, \ldots, n\} \qquad \textit{[C8]}$$

$$q_j + x_{j,s} \geq 0 \qquad \forall s \in \{1, \ldots, S\} \qquad \textit{[C12]}$$

Proof. The first part of the objective function is the optimizer's income, which is equal to

$$\begin{aligned}
\sum_{s=1}^{S} \theta_s p_s (q_j + x_{j,s}) &= \sum_{s=1}^{S} \theta_s p_s q_j + \sum_{s=1}^{S} \theta_s p_s x_{j,s} \\
&= fq_j + \sum_{s=1}^{S} \theta_s f x_{j,s} + \sum_{s=1}^{S} \theta_s (p_s - f) x_{j,s} && \text{From [C5]} \\
&= fq_j + \sum_{s=1}^{S} \theta_s (p_s - f) x_{j,s} + f \sum_{s=1}^{S} \theta_s x_{j,s} \\
&= fq_j + \sum_{s=1}^{S} \theta_s (p_s - f) x_{j,s} && \text{From [C1]}
\end{aligned}$$

The rest of the objective function can also be simplified similarly, as follows.

$$
\begin{aligned}
\text{Generating Cost} = & \sum_{s=1}^{S} \theta_s \left(\alpha_j \left(q_j + x_{j,s} \right) + \frac{\beta_j}{2} \ \left(q_j + x_{j,s} \right)^2 + \frac{\delta_j}{2} {x_{j,s}}^2 \right) \\
= & \alpha_j q_j + \frac{\beta_j}{2} q_j^2 + \frac{\beta_j + \delta_j}{2} \sum_{s=1}^{S} \theta_s {x_{j,s}}^2 \\
& + \left(\alpha_j + \beta_j q_j \right) \sum_{s=1}^{S} \theta_s x_{j,s} \\
= & \alpha_j q_j + \frac{\beta_j}{2} q_j^2 + \frac{\beta_j + \delta_j}{2} \sum_{s=1}^{S} \theta_s {x_{j,s}}^2 \qquad \text{From [C1]}
\end{aligned}
$$

□

Lemma 6.3. *If for every $i \neq j$ (j is the optimizer generator), a_i and d_i has the same value, then the optimal solution to WONN is at least as good as the optimal value to RWNN.*

Proof. To prove the lemma, we find the optimal solution to RWNN, while we ignore the non-negativity constraint $q_j + x_{j,s} \geq 0$. Thus, this point gives an objective value as good as (possibly better than) the optimal point. Then we show this point is a feasible solution to WONN, which proves the lemma.

From lemma 6.2 we have

$$
e_{i,s} = \theta_s \left(-Y_s + Z \left(\sum_{h=1}^{n} q_h + \sum_{h=1}^{n} x_{h,s} \right) + a_i + b q_i + (b + d_i) x_{i,s} \right).
$$

To simplify the equations we use some transformations. Let $R_i = \frac{1}{(b+d_i)}$, and $A_i = \frac{a_i}{b}$. Also, let A and R denote $\sum_{h=1}^{n} A_h$, and $\sum_{h=1}^{n} R_h$ respectively. Then, constraint [C6] looks like

$$
e_{i,s} = \theta_s \left(-Y_s + Z \left(\sum_{h=1}^{n} q_h + \sum_{h=1}^{n} x_{h,s} \right) + \frac{1}{R_i} x_{i,s} + b \left(A_i + q_i \right) \right). \tag{45}
$$

A summation over different scenarios gives

$$
\sum_{w=1}^{S} e_{i,w} = -Y + Z \sum_{h=1}^{n} q_i + \left(A_i + q_i \right) b \tag{46}
$$

From lemma 6.1, the constraints $e_{i,s} = 0$ for every $i \neq j$ and s can be replaced with $e_{i,s} \geq 0$ in RWNN. On the other hand, from the assumption we know that A_i has a fixed value for every $i \neq j$. As a result, equation (46) is used to show that q_i must have a fixed value for every $i \neq j$. Thus, equation (46) can be re-written as

$$0 = -Y + Z\left((n-1)q_i + q_j\right) + \left(A_i + q_i\right)b \tag{47}$$

With a similar argument, we can show that $x_{i,s}$ also has the same value for every $i \neq j$. Equation (45), thus, can be represented as

$$0 = \theta_s\left(-Y_s + Z\left((n-1)q_i + q_j + (n-1)x_{i,s} + x_{j,s}\right) + \frac{1}{R_i}x_{i,s} + b\left(A_i + q_i\right)\right) \tag{48}$$

Solving equations (47) and (48), we find the values of q_i and $x_{i,s}$ as functions of q_j and $x_{j,s}$.

$$\begin{aligned} q_i =& \frac{Y - bA_i - Zq_j}{b + (n-1)Z} \\ x_{i,s} =& -\frac{R_i\left(Y - Y_s + Zx_{j,s}\right)}{1 + (n-1)ZR_i} \end{aligned} \tag{49}$$

From (49) we can also calculate the values of f and $p_s - f$ as functions of q_j and $x_{j,s}$.

$$\begin{aligned} f =& \frac{b\left(Y + (n-1)ZA_i - Zq_j\right)}{b + (n-1)Z} \\ p_s - f =& \frac{-Y + Y_s - Zx_{j,s}}{1 + (n-1)ZR_i} \end{aligned} \tag{50}$$

Inserting these values into the utility function from lemma 6.2 simplifies the utility function to

$$\begin{aligned} u_j = & \left(\frac{b\left(Y + (n-1)ZA_i - Zq_j\right)}{b + (n-1)Z} - \alpha_j - \frac{\beta_j}{2}q_j\right)q_j \\ & + \sum_{s=1}^{S}\theta_s\left(\frac{-Y + Y_s - Zx_{j,s}}{1 + (n-1)ZR_i} - \frac{\beta_j + \delta_j}{2}x_{j,s}\right)x_{j,s} \end{aligned}$$

As Z, α_j, β_j, and R_i have non-negative values, u_j is a concave function of q_j and $\boldsymbol{x}_j$. Therefore, ignoring the rest of the constraints, the optimal value of q_j and $x_{j,s}$ can be found using first order conditions.

First order conditions for q_j and $x_{j,s}$ gives

$$q_j^* = \frac{bY + (n-1)bZA_i - (b + (n-1)Z)\alpha_j}{2bZ + (b + (n-1)Z)\beta_j} \tag{51}$$

$$x_{j,s}^* = \frac{Y_s - Y}{2Z + (1 + (n-1)ZR_i)(\beta_j + \delta_j)}. \tag{52}$$

Now we need to show that we can always find A_j and R_j, so that this value is a feasible solution to WONN and yields $e_{j,s} = 0$. To do so, we first calculate $\frac{e_{j,s}}{\theta_s} - \sum_w e_{j,w}$ for all s. From (45), (46), and (51)

$$\begin{aligned}\frac{e_{j,s}}{\theta_s} - \sum_{w=1}^{S} e_{j,w} =& Y - Y_s + \frac{x_{j,s}}{R_j} + Z\left((n-1)x_{i,s} + x_{j,s}\right)\\ =& \frac{(Y - Y_s)\left(-1 + R_j\left(Z + \beta_j + \delta_j\right) + (n-1)ZR_i\left(-1 + R_j\left(\beta_j + \delta_j\right)\right)\right)}{R_j\left(1 + (n-1)ZR_i\right)\left(2Z + \left(1 + (n-1)ZR_i\right)\left(\beta_j + \delta_j\right)\right)}\end{aligned} \tag{53}$$

It is always possible to choose R_j as follows to ensure that $\frac{e_{j,s}}{\theta_s} - \sum_w e_{j,w} = 0$. Note that this does not change either of production quantities or prices. This value of R_j is

$$R_j = \frac{1 + (n-1)ZR_i}{Z + (1 + (n-1)ZR_i)(\beta_j + \delta_j)}$$

We can also choose A_j so that $\sum_w e_{j,w} = 0$ without changing any production quantity and thus any prices. From (46) and (49)

$$\begin{aligned}\sum_{w=1}^{S} e_{j,w} =& -Y + Z\left((n-1)q_i + q_j\right) + b\left(A_h + q_h\right)\\ =& -Y + b\left(A_j + q_j\right) + \frac{(n-1)Z\left(Y - bA_i\right) + bZq_j}{b + (n-1)Z}\end{aligned} \tag{54}$$

Solving $\sum_w e_{j,w} = 0$ for A_j gives

$$\begin{aligned}A_j =& \frac{-bY(b + (n-2)Z) + (b + (n-1)Z)\left((b + nZ)\alpha_j + Y\beta_j\right)}{(b + (n-1)Z)\left(2bZ + (b + (n-1)Z)\beta_j\right)}\\ &+ \frac{-(n-1)ZA_i\left(b(b + (n-2)Z) - (b + (n-1)Z)\beta_j\right)}{(b + (n-1)Z)\left(2bZ + (b + (n-1)Z)\beta_j\right)}\end{aligned}$$

These A_j and R_j ensures $\begin{aligned}\sum_{w=1}^{S} e_{j,w} =& 0\\ \frac{e_{j,s}}{\theta_s} - \sum_{w=1}^{S} e_{j,w} =& 0\end{aligned} \Rightarrow \forall s : e_{j,s} = 0$

Thus, constraints [C6] and [C8] are met in WONN and RWNN.

From (51) we derive $\sum_s \theta_s x_{j,s} = 0$. We can use the fact that $\sum_s \theta_s x_{j,s} = 0$ to show that for $i \neq j$ also $\sum_s \theta_s x_{i,s} = 0$ (in equation (49)). So, this optimal point is feasible in [C1].

In sum, the constructed point is feasible to WONN, and gives an objective value at least as good as RWNN. □

Now, we can use the above lemmas to prove a theorem that shows using the equilibrium of the simplifies game without the non-negativity constraints instead of the equilibrium of the original game is justifiable.

Theorem. *The equilibrium of the symmetric SFSP game without the non-negativity constraints in ISO's problem is also the equilibrium of SFSP game with the non-negativity constraints.*

Proof. To prove the theorem, we should show that if all generators offer the equilibrium values of a_i and d_j none of them are willing to deviate from it. Equivalently, if in WNN a_i and d_i are equal to the equilibrium of the SFSP game without the non-negativity constraints for all $i \neq j$, then optimal a_j and d_j are also equal to equilibrium values of this game.

The equilibrium of SFSP without non-negativity constraints is equal to the optimal value of WONN when every non-optimizer generator has offered the equilibrium values of the game. Thus, we prove that the optimal value of WONN is also optimal to WNN.

Firstly, lemma 6.3 states that if the optimal solution to WONN is feasible to RWNN, then, it is also the optimal solution to RWNN. In our problem, from theorem 4.9 we know that the optimal solution to WONN holds both $q_i \geq 0$ and $q_i + x_{i,s} \geq 0$. The other constraints of RWNN are shared between these two models. Thus, it is feasible and optimal in RWNN.

On the other hand, every feasible solution to WNN is feasible in RWNN. So, if this solution (which is the optimal solution to RWNN) is feasible to WNN, then it is also optimal to WNN. From the theorem 4.9, we know that $q_i \geq 0$ and $q_i + x_{i,s} \geq 0$ for all i, as it is the equilibrium of the game without non-negativity constraints. This means this point is feasible in [C8] and [C9]. On the other hand, we know that $e_{i,s} = 0$ for all i, as it is the optimal solution to WONN. This shows it also holds [C7]. The other constraints are common and thus met. In sum, This point is feasible and therefore optimal to WNN.

Thus, no generator is willing to deviate from this point unilaterally, and this is the equilibrium of WNN. □

7 Stochastic settlement outperforms the conventional settlement: proof of proposition 5.1

Proposition. *When the parameter b_i is chosen less than the threshold value of $\hat{b}$, where*

$$\hat{b} = \frac{-Z(n-2) + \beta + \delta + \sqrt{Z^2(n-2)^2 + 2Zn(\beta+\delta) + (\beta+\delta)^2}}{2},$$

social welfare in the stochastic settlement market is higher than that in the two-settlement market.

Proof. To prove the proposition, we show when $b_i = \hat{b}$, we can conclude $\mathrm{SW}^{\mathrm{SS}} = \mathrm{SW}^{\mathrm{TS}}$. Then, we demonstrate $\mathrm{SW}^{\mathrm{SS}}$ is a decreasing function of b_i, and therefore, $\mathrm{SW}^{\mathrm{SS}} \geq \mathrm{SW}^{\mathrm{TS}}$, when $b_i \leq \hat{b}$.

When $b_i = \hat{b}$, it is easy to show that equilibrium quantities are identical in the stochastic settlement and two settlement markets. (equations (31), (**??**), (**??**))

$$\begin{aligned} B_i^{\mathrm{SS}} &= B_i^{\mathrm{TS}} \\ A_i^{\mathrm{SS}} &= A_i^{\mathrm{TS}} \\ R_i^{\mathrm{SS}} &= B_i^{\mathrm{SS}} \end{aligned}$$

Under this situation we can show that $y_{i,s}$ and q_i formulae (from propositions **??**, **??**, and **??**) simplifies to

$$\begin{aligned} q_i^{\mathrm{SS}} &= q_i^{\mathrm{TS}} = \frac{YB_i - A_i}{1+ZB} \\ y_{i,s}^{\mathrm{SS}} &= y_{i,s}^{\mathrm{TS}} = \frac{Y_s B_i - A_i}{1+ZB} \end{aligned}$$

Therefore social welfare of these models (equation **??**) are the same providing $b_i = \hat{b}$.

Note that we can rewrite social welfare formula (**??**) as

$$SW = \sum_{s=1}^{S} \theta_s \left(Y_s \left(\sum_{i=1}^{n} y_{i,s} \right) - \frac{Z}{2} \left(\sum_{i=1}^{n} y_{i,s} \right)^2 - \sum_{i=1}^{n} \left(\alpha y_{i,s} + \frac{\beta}{2} {y_{i,s}}^2 + \frac{\delta}{2} {x_{i,s}}^2 \right) \right).$$

Note that $x_{i,s}$ is independent of b, and therefore,

$$\frac{dW}{db_i} = -\frac{1}{b_i^2} \sum_{i,s} \frac{dW}{dy_{i,s}} \frac{dy_{i,s}}{dB_i}.$$

On the other hand, we show (in the technical companion [**?**]) that

$$\frac{dy_{i,s}}{dB_i} = \frac{(Y-\alpha)(n-1)Z^2}{(Z+nZ+\beta+(n-1)Z(nZ+\beta)B_i)^2} \geq 0.$$

Note that according to our assumptions $\forall s$, $\alpha \leq Y_s$. Also, this derivative is a fixed number independent of i and s. Thus,

$$\frac{dW}{db_i} = -\frac{1}{b_i^2}\frac{dy_{i,s}}{dB_i}\sum_{i,s}\frac{dW}{dy_{i,s}}.$$

On the other hand,

$$\frac{dW}{dy_{i,s}} = \theta_s\left(Y_s-\alpha-(Zn+\beta)y_{i,s}\right).$$

Hence,

$$\begin{aligned}\sum_s \frac{dW}{dy_{i,s}} &= Y-\alpha-(Zn+\beta)\,q_i \\ &= \frac{bZ(Y-\alpha)}{Z(n-1)(nZ+\beta)+b((n+1)Z+\beta)} \geq 0.\end{aligned}$$

In sum, we can conclude that,

$$\frac{dW}{db_i} \leq 0.$$

□

8 Computations of firms and equilibrium values for the SP mechanism

Best response curves

From propositions 3.1 and 3.2, we have

$$Q = \frac{EY\ cB - cA}{1 + Z\ cB}$$

$$q_{i\_} = \frac{(EY + Z\ cA)\ B_i}{(1 + Z\ cB)} - A_i$$

$$x_{i\_,s\_} = \frac{(Y_s - EY)\ R_i}{1 + Z\ cR}$$

$$y_{i\_,s\_} = q_i + x_{i,s}$$

$$f = \frac{EY + Z\ cA}{1 + Z\ cB}$$

$$p_{s\_} = f + \frac{Y_s - EY}{1 + Z\ cR}$$

$$\theta_S = 1 - \sum_{s=1}^{S-1} \theta_s$$

$$cA = A_i + A_{-i}$$

$$cB = B_i + B_{-i}$$

$$cR = R_i + R_{-i}$$

$$u_{i\_} = \text{Simplify}\left[f\ q_i + \sum_{s=1}^{S} \theta_s \left(p_s\ x_{i,s} - \left(\alpha\ (q_i + x_{i,s}) + \frac{\beta}{2}\ (q_i + x_{i,s})^2 + \frac{\delta}{2}\ x_{i,s}{}^2\right)\right)\right];$$

$$\text{welfare} = \sum_{s=1}^{S} \theta_s \left(Y_s\ (n\ y_{1,s}) - \frac{Z}{2}\ (n\ y_{1,s})^2 - n\left(\alpha\ y_{1,s} + \frac{\beta}{2}\ y_{1,s}{}^2 + \frac{\delta}{2}\ x_{1,s}{}^2\right)\right);$$

$$\text{FullSimplify}\left[D[u_i, R_i, A_i], \left\{EY == \sum_{s=1}^{S} \theta_s\ Y_s,\ \theta_S = 1 - \sum_{s=1}^{S-1} \theta_s\right\}\right]$$

0

$$\text{FullSimplify}\left[D[u_i, R_i, B_i], \left\{EY == \sum_{s=1}^{S} \theta_s\ Y_s,\ \theta_S = 1 - \sum_{s=1}^{S-1} \theta_s\right\}\right]$$

0

Therefore, $u_i(A_i, B_i, R_i) = g_i(A_i, B_i) + h_i(R_i)$.

$$\text{FullSimplify}[D[u_i, A_i, A_i]]$$

$$-\frac{(1 + Z\ B_{-i})\ (2\ Z + \beta + Z\ \beta\ B_{-i})}{(1 + Z\ (B_{-i} + B_i))^2}$$

```
FullSimplify[Solve[D[u_i, A_i] == 0, A_i], {EY == Sum[θ_s Y_s, {s, 1, S}], θ_S = 1 - Sum[θ_s, {s, 1, S-1}]}]
```

$$\left\{\left\{A_i \to \frac{(1+Z B_{-i})(-EY+\alpha-Z A_{-i}+Z\alpha B_{-i})+(Z\alpha+EY(Z+\beta)+Z(Z\alpha+EY\beta)B_{-i}+Z A_{-i}(Z+\beta+Z\beta B_{-i}))B_i}{(1+Z B_{-i})(2Z+\beta+Z\beta B_{-i})}\right\}\right\}$$

$$A_i = \frac{(1+Z B_{-i})(-EY+\alpha-Z A_{-i}+Z\alpha B_{-i})+(Z\alpha+EY(Z+\beta)+Z(Z\alpha+EY\beta)B_{-i}+Z A_{-i}(Z+\beta+Z\beta B_{-i}))B_i}{(1+Z B_{-i})(2Z+\beta+Z\beta B_{-i})}$$

$$\frac{(1+Z B_{-i})(-EY+\alpha-Z A_{-i}+Z\alpha B_{-i})+(Z\alpha+EY(Z+\beta)+Z(Z\alpha+EY\beta)B_{-i}+Z A_{-i}(Z+\beta+Z\beta B_{-i}))B_i}{(1+Z B_{-i})(2Z+\beta+Z\beta B_{-i})}$$

```
FullSimplify[D[u_i, B_i]]
```

0

The fact that derivative of u_i with respect to B_i is zero means $u_i(A_i^*(B_i), B_i, R_i)$ and $g_i(A_i^*(B_i), B_i)$ is independant of B_i. Therefore, $g_i(A_i^*(B_i), B_i)$ is a constant dependant on the cost and demand parameters.

```
FullSimplify[D[u_i, R_i], {EY == Sum[θ_s Y_s, {s, 1, S}], θ_S = 1 - Sum[θ_s, {s, 1, S-1}]}]
```

$$\frac{(-1+(Z+\beta+\delta)R_i+Z R_{-i}(-1+(\beta+\delta)R_i))\left(EY^2-\sum_{s=1}^{S} Y_s^2\theta_s\right)}{(1+Z(R_{-i}+R_i))^3}$$

The expression $(-1+(Z+\beta+\delta)R_i+Z R_{-i}(-1+(\beta+\delta)R_i))$ is a linear increasing function of R_i. Thus, it is negative bellow its zero and is positive after its zero. The denominator $(1+Z(R_{-i}+R_i))^3$ is positive, and $\left(EY^2-\sum_{s=1}^{S}\theta_s Y_s^2\right)$ is negative (because of Jensen's inequality). In sum, $\frac{du_i}{dR_i}$ is positive before its zero and is negative after this point. Thus, it is a quasi-concave function of R_i.

```
FullSimplify[Solve[D[u_i, R_i] == 0, R_i], {EY == Sum[θ_s Y_s, {s, 1, S}], θ_S = 1 - Sum[θ_s, {s, 1, S-1}]}]
```

$$\left\{\left\{R_i \to \frac{1+Z R_{-i}}{Z+\beta+\delta+Z(\beta+\delta)R_{-i}}\right\}\right\}$$

R_i must be less than B_i, and u_i is a quasi-concave function of R_i. Therefore, the optimal R_i is

$$R_i = \mathrm{Min}\left[B_i, \frac{1+Z R_{-i}}{Z+\beta+\delta+Z(\beta+\delta)R_{-i}}\right]$$

Finding a symmetric equilibrium

```
FullSimplify[Solve[{A_i ==
    ((1 + Z B_-i) (-EY + α - Z A_-i + Z α B_-i) + (Z α + EY (Z + β) + Z (Z α + EY β) B_-i + Z A_-i (Z + β + Z β B_-i))
      B_i) / ((1 + Z B_-i) (2 Z + β + Z β B_-i)), A_-i == (n - 1) A_i}, {A_i, A_-i}]]
```

$$\left\{\left\{A_i \to \frac{(1+Z B_{-i})(-EY+\alpha+Z\alpha B_{-i})+(Z\alpha+EY(Z+\beta)+Z(Z\alpha+EY\beta)B_{-i})B_i}{(1+Z B_{-i})(Z+nZ+\beta+Z\beta B_{-i})-(-1+n)Z(Z+\beta+Z\beta B_{-i})B_i},\right.\right.$$

$$\left.\left.A_{-i} \to \frac{(-1+n)(-(1+Z B_{-i})(-EY+\alpha+Z\alpha B_{-i})-(Z\alpha+EY(Z+\beta)+Z(Z\alpha+EY\beta)B_{-i})B_i)}{-(1+Z B_{-i})(Z+nZ+\beta+Z\beta B_{-i})+(-1+n)Z(Z+\beta+Z\beta B_{-i})B_i}\right\}\right\}$$

```
n =.
FullSimplify[Solve[{R_i == (1 + Z R_-i)/(Z + β + δ + Z (β + δ) R_-i), R_-i == (n - 1) R_i}, {R_i, R_-i}]]
```

$$\left\{\left\{R_i \to -\frac{2}{(-2+n)\,Z-\beta-\delta+\sqrt{(-2+n)^2\,Z^2+2\,n\,Z\,(\beta+\delta)+(\beta+\delta)^2}},\right.\right.$$

$$\left.R_{-i} \to -\frac{-(-2+n)\,Z+\beta+\delta+\sqrt{(-2+n)^2\,Z^2+2\,n\,Z\,(\beta+\delta)+(\beta+\delta)^2}}{2\,Z\,(\beta+\delta)}\right\},$$

$$\left\{R_i \to \frac{2}{-(-2+n)\,Z+\beta+\delta+\sqrt{(-2+n)^2\,Z^2+2\,n\,Z\,(\beta+\delta)+(\beta+\delta)^2}},\right.$$

$$\left.\left.R_{-i} \to \frac{(-2+n)\,Z-\beta-\delta+\sqrt{(-2+n)^2\,Z^2+2\,n\,Z\,(\beta+\delta)+(\beta+\delta)^2}}{2\,Z\,(\beta+\delta)}\right\}\right\}$$

The expression $-\frac{2}{(-2+n)\,Z-\beta-\delta+\sqrt{(-2+n)^2\,Z^2+2\,n\,Z\,(\beta+\delta)+(\beta+\delta)^2}}$ is negative. However, $\hat{R}_i = \frac{2}{-(-2+n)\,Z+\beta+\delta+\sqrt{(-2+n)^2\,Z^2+2\,n\,Z\,(\beta+\delta)+(\beta+\delta)^2}}$ is positive and acceptable. As we show in the paper, the equilibrium R_i is $\mathrm{Min}\left[B_i, \frac{2}{-(-2+n)\,Z+\beta+\delta+\sqrt{(-2+n)^2\,Z^2+2\,n\,Z\,(\beta+\delta)+(\beta+\delta)^2}}\right]$.

9 The LINGO model used to find the equilibrium of the two settlement mechanism with asymmetric firms

```
!The two settlement model with asymmetric generators and non-negativity constraints.;
MODEL:

DATA:
NumProblems= @OLE('TS1.xls', 'GENERATORS!K16');
ENDDATA

!The similar parameters to the parameters defined in the original paper have a similar
definition.
The rest of parameters are defined as comments.;
SETS:
GENERATORS :  b , alpha , beta , delta , q , a ,d, a_fixed,b_fixed,d_fixed, Opt,
optimizer, profit,lambda;
!
***a_fixed, b_fixed, d_fixed: The offered parametters of the generators in the last run.
***opt: If the current decision of the optimizing generator is similar (with a precision)
to its decision in the last run, it is 1, otherwise it is zero.
***optimizer: in each round it is one for the optimizing generator and zero for the
others.
***Lambda: The dual variable of the non-negativity constraint q_{i}>=0.
;
SCENARIOS : Y, theta, transCoef, p, C;
GEN_SCEN (GENERATORS, SCENARIOS): x, e, boundary ;
!
***e: The dual variable of the non-negativity consttraint q_i+x_{i,s}>=0.
***boundary: A binary variable to linearize the orthogonality constraint e_{i,s}(q_{i}+x_
{i,s})=0.
;
OPTIMIZERS  (GENERATORS);
!
The set of the optimizer generator in each step of the dynamic process.
;
FIXEDGENS(GENERATORS) | #NOT# @IN( OPTIMIZERS, &1) ;
!All non-optimizer generators;
ROWS /1..100/:alp,bet,del,op,a_f,b_f,d_f ,tet,coe,Y_f,Z_f,walpha ,wbeta ,wdelta
,woptimizer ,wa_fixed ,wb_fixed,wd_fixed ,wb ,wa ,wd ,wq ,wx1 ,wx2
,wprofit,wf,wp1,wp2,wwelfare,wrep,wst1,wst2,wtet,wcoe,wY_f,wZ_f;
!Degined for the purpose of collecting result of different runs of the model, and
outputting the results.;

ENDSETS

! Here is the data.
The data is read from an Excel file.
;
DATA:
GENERATORS, OPTIMIZERS= @OLE( 'TS1.xls', 'GENERATORS','OPTIMIZERS');
SCENARIOS = @OLE('TS1.xls','SCENARIOS');
theta, transCoef, Y = @OLE('TS1.xls','SCENSDATA');
Z, MyBigM = @OLE('TS1.xls' , 'Z' , 'MyBigM');
alp,bet,del,op,a_f,b_f,d_f = @OLE('TS1.xls','GENERATORS!D16:J116');
tet,coe,Y_f,Z_f = @OLE('TS1.xls','GENERATORS!N16:Q116');

precision =@OLE('TS1.xls','GENERATORS!R18');
!A tolerance that determines the smallest value that we consider as a change in strategy.
In other words, if the change in a firm's strategy is less than this, we count that as a
```

```
no change in the strategy.;

maxRep=@OLE('TS1.xls','GENERATORS!R19');
!If we do not find an equilibrium after "maxRep" steps, we stop searching for it.;

ENDDATA

SUBMODEL TS1:
!This is the optimization model solved by a firm to maximize profit, assuming that the
strategy set of all other firms are fixed. ;
@FOR(GENERATORS: @FREE(a));
@FOR(SCENARIOS: @FREE(p));
@FOR(GEN_SCEN: @FREE(x));
@FREE(f);

[obj] MAX = @sum (GENERATORS(i): optimizer(i)* (  f * q(i)+ @sum (SCENARIOS(s): (theta(s)*
(p(s)* x(i,s)-(alpha(i) * (q(i)+x(i,s))+ beta(i)/2 * (q(i)+x(i,s))^2 + delta(i)/2 *x(i,s)^
2)))  ) ));
! The objective;

!The constraints include constraints of a generator on his offered supply function and KKT
consitions of the ISO's optimization problem;
@FOR (GENERATORS(i):
      -f+a(i)+b(i)*q(i)-lambda(i)=0;
      q(i)*lambda(i)=0;
);

@FOR(GEN_SCEN(i,s):
       a(i)-p(s)-e(i,s)+b(i)*(q(i)+x(i,s))=0;
      q(i) + x(i,s) >= 0;
      [Const_ebin]      e(i,s)  <= boundary(i,s)*MyBigM;
      [Const_qxbin]     q(i)+x(i,s)  <= (1-boundary(i,s))*MyBigM;
      @BIN(boundary(i,s));
);


@FOR(SCENARIOS(S):
[Const_p_demand]  theta(s) * (p(s)+ Z* C(s)-Y(s))   = 0;
[Const_C]   theta(s)*(-cQ+C(s)-@sum (GENERATORS(i): x(i,s))) = 0;
);

!Non-optimizing generators should offer their previous offered parameters;
@FOR (GENERATORS(k)|optimizer(k) #EQ# 0 :
      a(k)=a_fixed(k);
      b(k)=b_fixed(k);
);

-@sum(SCENARIOS(s):theta(s)*Y(s))+f+cQ*Z=0;
cQ - @sum(GENERATORS(h): q(h))= 0;

ENDSUBMODEL

!Calculations and procedure of the dynamic process to find an equilibrium for each of the
market settings.;
CALC:
@for(ROWS(k):

walpha(k)=0;
wbeta(k)=0;
wdelta(k)=0;
woptimizer(k)=0;
wa_fixed(k)=0;
wb_fixed(k)=0;
wd_fixed(k)=0;
wb(k)=0;
wa(k)=0;
```

```
wd(k)=0;
wq(k)=0;
wx1(k)=0;
wx2(k)=0;
wprofit(k)=0;
wf(k)=0;
wp1(k)=0;
wp2(k)=0;
wwelfare(k)=0;
wrep(k)=0;
wst1(k)=0;
wst2(k)=0;
wtet(k)=0;
wcoe(k)=0;
wY_f(k)=0;
wZ_f(k)=0;
);

!Reading different market settings (i.e. case studies or examples).;
ind=@OLE('TS1.xls','GENERATORS!L16');
@WHILE (ind #LE# NumProblems:
eq=0;
rep=0;
alp1=alp(2*(ind-1)+1);
alp2=alp(2*(ind-1)+2);
bet1=bet(2*(ind-1)+1);
bet2=bet(2*(ind-1)+2);
del1=del(2*(ind-1)+1);
del2=del(2*(ind-1)+2);
op1=op(2*(ind-1)+1);
op2=op(2*(ind-1)+2);
a_f1=a_f(2*(ind-1)+1);
a_f2=a_f(2*(ind-1)+2);
b_f1=b_f(2*(ind-1)+1);
b_f2=b_f(2*(ind-1)+2);
d_f1=d_f(2*(ind-1)+1);
d_f2=d_f(2*(ind-1)+2);
tet1=tet(2*(ind-1)+1);
tet2=tet(2*(ind-1)+2);
coe1=coe(2*(ind-1)+1);
coe2=coe(2*(ind-1)+2);
Y_f1=Y_f(2*(ind-1)+1);
Y_f2=Y_f(2*(ind-1)+2);
Z_f1=Z_f(2*(ind-1)+1);

@OLE('TS1.xls','GENERATORS!D2:j2')=alp1,bet1,del1,op1,a_f1,b_f1,d_f1;
@OLE('TS1.xls','GENERATORS!D3:j3')=alp2,bet2,del2,op2,a_f2,b_f2,d_f2;

@OLE('TS1.xls','SCENARIOS!C2:E2')=tet1,coe1,Y_f1;
@OLE('TS1.xls','SCENARIOS!C3:E3')=tet2,coe2,Y_f2;

@OLE('TS1.xls','OtherParams!B2')=Z_f1;


@for( GENERATORS(i):
      Opt(i)=0;
);
! st1 and st2 records the status of the optimization problems i.e. whether it is found a
global optimal solution or a local optima. These are importnt to ensure that we actually
find a true equilibrium.;
st1=1000;
st2=1000;
@WHILE (eq #LE# 1 #AND# rep#LE#maxRep:
      st1=st2;
      alpha, beta, delta, optimizer, a_fixed, b_fixed,d_fixed = @OLE( 'TS1.xls',
'GENSDATA');
      theta, transCoef, Y = @OLE('TS1.xls','SCENSDATA');
```

```
        Z = @OLE('TS1.xls' , 'Z');
        @SOLVE( TS1);

        @for(GENERATORS(i)| optimizer(i) #EQ# 1 :
              @ifc( a(i) #GE# a_fixed(i)-precision  #AND#  a(i) #LE# a_fixed(i)+precision
#AND# b(i) #GE# b_fixed(i)-precision  #AND#  b(i) #LE# b_fixed(i)+precision:
                    Opt(i)=1;
              @else
                    Opt(i)=0;
               );

              a_fixed(i) = a(i);
              b_fixed(i) = b(i);
        );
        @for(GENERATORS(i):
              @ifc( optimizer(i) #EQ# 1:
                    optimizer(i)=0;
              @else
                    optimizer(i)=1;
               );
        );
        st2=@STATUS();
        eq = @sum(GENERATORS(i): Opt(i));
        @OLE( 'TS1.xls', 'GENSDATA') = alpha, beta, delta, optimizer, a_fixed, b_fixed,
d_fixed;
        rep=rep+1;
        @for(GENERATORS(i):
              profit(i) =  f * q(i)+ @sum (SCENARIOS(s): (theta(s)*(p(s)* x(i,s)-(alpha(i) *
(q(i)+x(i,s))+ beta(i)/2 * (q(i)+x(i,s))^2 + delta(i)/2 *x(i,s)^2)))  ) ;
        );

        !Intermediate output;
        welfare = @sum(SCENARIOS(s): theta(s)*(Y(s)*C(s)-Z/2*C(s)^2-@sum(GENERATORS(i):
alpha(i)*(q(i)+x(i,s))+beta(i)/2*(q(i)+x(i,s))^2+delta(i)/2*x(i,s)^2)));
        @OLE( 'TS1.xls', 'GENERATORS!L2:N3') = a, d, q;
        @OLE( 'TS1.xls', 'GENERATORS!O2:O3') = @writefor(GEN_SCEN(i,s)|s #EQ# 1: x(i,s));
        @OLE( 'TS1.xls', 'GENERATORS!P2:P3') = @writefor(GEN_SCEN(i,s)|s #EQ# 2: x(i,s));
        @OLE( 'TS1.xls', 'GENERATORS!Q2:Q3') =profit;
        @OLE( 'TS1.xls', 'GENERATORS!R2:R2') = f;
        @OLE( 'TS1.xls', 'GENERATORS!S2:T2') = p;
        @OLE( 'TS1.xls', 'GENERATORS!U2:U2') = welfare;
        @OLE( 'TS1.xls', 'GENERATORS!V2:V2') = rep;
        @OLE( 'TS1.xls', 'GENERATORS!W2:W2') = st1;
        @OLE( 'TS1.xls', 'GENERATORS!X2:X2') = st2;
        @OLE( 'TS1.xls', 'GENERATORS!Y2:Y2') = @write('WNN');
);

!Final output;
welfare = @sum(SCENARIOS(s): theta(s)*(Y(s)*C(s)-Z/2*C(s)^2-@sum(GENERATORS(i): alpha(i)*
(q(i)+x(i,s))+beta(i)/2*(q(i)+x(i,s))^2+delta(i)/2*x(i,s)^2)));
@OLE( 'TS1.xls', 'GENERATORS!L2:N3') = a, b, q;
@OLE( 'TS1.xls', 'GENERATORS!O2:O3') = @writefor(GEN_SCEN(i,s)|s #EQ# 1: x(i,s));
@OLE( 'TS1.xls', 'GENERATORS!P2:P3') = @writefor(GEN_SCEN(i,s)|s #EQ# 2: x(i,s));
@OLE( 'TS1.xls', 'GENERATORS!Q2:Q3') = profit;
@OLE( 'TS1.xls', 'GENERATORS!R2:R2') = f;
@OLE( 'TS1.xls', 'GENERATORS!S2:T2') = p;
@OLE( 'TS1.xls', 'GENERATORS!U2:U2') = welfare;
@OLE( 'TS1.xls', 'GENERATORS!V2:V2') = rep;
@OLE( 'TS1.xls', 'GENERATORS!W2:W2') = st1;
@OLE( 'TS1.xls', 'GENERATORS!X2:X2') = st2;
@OLE( 'TS1.xls', 'GENERATORS!Y2:Y2') = @write('WNN');
i=1;
        walpha(2*(ind-1)+i)=alpha(i);
        wbeta(2*(ind-1)+i)=beta(i);
        wdelta(2*(ind-1)+i)=delta(i);
        woptimizer(2*(ind-1)+i)=optimizer(i);
        wa_fixed(2*(ind-1)+i)=a_f(2*(ind-1)+i);
```

```
        wb_fixed(2*(ind-1)+i)=b_f(2*(ind-1)+i);
        wd_fixed(2*(ind-1)+i)=d_f(2*(ind-1)+i);
        wa(2*(ind-1)+i)=a(i);
        wb(2*(ind-1)+i)=b(i);
        wd(2*(ind-1)+i)=d(i);
        wq(2*(ind-1)+i)=q(i);
        wx1(2*(ind-1)+i)=x(i,1);
        wx2(2*(ind-1)+i)=x(i,2);
        wprofit(2*(ind-1)+i)=profit(i);
        wf(2*(ind-1)+i)=f;
        wp1(2*(ind-1)+i)=p(1);
        wp2(2*(ind-1)+i)=p(2);
        wwelfare(2*(ind-1)+i)=welfare;
        wrep(2*(ind-1)+i)=rep;
        wst1(2*(ind-1)+i)=st1;
        wst2(2*(ind-1)+i)=st2;
        wtet(2*(ind-1)+i)=theta(i);
        wcoe(2*(ind-1)+i)=transCoef(i);
        wY_f(2*(ind-1)+i)=Y(i);
        wZ_f(2*(ind-1)+i)=Z;

i=2;
        walpha(2*(ind-1)+i)=alpha(i);
        wbeta(2*(ind-1)+i)=beta(i);
        wdelta(2*(ind-1)+i)=delta(i);
        woptimizer(2*(ind-1)+i)=optimizer(i);
        wa_fixed(2*(ind-1)+i)=a_f(2*(ind-1)+i);
        wb_fixed(2*(ind-1)+i)=b_f(2*(ind-1)+i);
        wd_fixed(2*(ind-1)+i)=d_f(2*(ind-1)+i);
        wa(2*(ind-1)+i)=a(i);
        wb(2*(ind-1)+i)=b(i);
        wd(2*(ind-1)+i)=d(i);
        wq(2*(ind-1)+i)=q(i);
        wx1(2*(ind-1)+i)=x(i,1);
        wx2(2*(ind-1)+i)=x(i,2);
        wprofit(2*(ind-1)+i)=profit(i);
        wf(2*(ind-1)+i)=f;
        wp1(2*(ind-1)+i)=p(1);
        wp2(2*(ind-1)+i)=p(2);
        wwelfare(2*(ind-1)+i)=welfare;
        wrep(2*(ind-1)+i)=rep;
        wst1(2*(ind-1)+i)=st1;
        wst2(2*(ind-1)+i)=st2;
        wtet(2*(ind-1)+i)=theta(i);
        wcoe(2*(ind-1)+i)=transCoef(i);
        wY_f(2*(ind-1)+i)=Y(i);
        wZ_f(2*(ind-1)+i)=Z;

ind=ind+1;
@OLE( 'TS1.xls', 'OUT_WNN!B2:E101') = wtet,wcoe,wY_f,wZ_f;
@OLE( 'TS1.xls', 'OUT_WNN!G2:AA101') =  walpha ,wbeta ,wdelta ,woptimizer ,wa_fixed
,wb_fixed,wd_fixed ,wa ,wb ,wd ,wq ,wx1 ,wx2 ,wprofit,wf,wp1,wp2,wwelfare,wrep,wst1,wst2;
);



ENDCALC
@WARN('LINGO Finished',1#GE#0);
END
```

10 The LINGO model used to find the equilibrium of the stochastic settlement mechanism with asymmetric firms

```
!The stochastic settlement model with asymmetric generators and non-negativity
constraints.;
MODEL:
DATA:
NumProblems= @OLE('SFSP.xls', 'GENERATORS!K16');
ENDDATA

!The similar parameters to the parameters defined in the original paper have a similar
definition.
The rest of parameters are defined as comments.;
SETS:

GENERATORS :  b , alpha , beta , delta , q , a , d, a_fixed, d_fixed, Opt, optimizer,
profit;
!
***a_fixed, b_fixed, d_fixed: The offered paramptters of the generators in the last run.
***opt: If the current decision of the optimizing generator is similar (with a precision)
to its decision in the last run, it is 1, otherwise it is zero.
***optimizer: in each round it is one for the optimizing generator and zero for the
others.
***
;
SCENARIOS : Y, theta, transCoef, p, C;
GEN_SCEN (GENERATORS, SCENARIOS): x, e, boundary ;
!
***e: The dual variable of the non-negativity consttraint q_i+x_{i,s}>=0.
***boundary: A binary variable to linearize the orthogonality constraint e_{i,s}(q_{i}+x_
{i,s})=0.
;
OPTIMIZERS  (GENERATORS);
!
The set of the optimizer generator in each step of the dynamic process.
;
FIXEDGENS(GENERATORS) | #NOT# @IN( OPTIMIZERS, &1) ;
!All non-optimizer generators;
ROWS /1..100/:alp,bet,del,op,a_f,b_f,d_f ,tet,coe,Y_f,Z_f,walpha ,wbeta ,wdelta
,woptimizer ,wa_fixed ,wb_fixed,wd_fixed ,wb ,wa ,wd ,wq ,wx1 ,wx2
,wprofit,wf,wp1,wp2,wwelfare,wrep,wst1,wst2,wtet,wcoe,wY_f,wZ_f;
!Degined for the purpose of collecting result of different runs of the model, and
outputting the results.;

ENDSETS

! Here is the data.
The data is read from an Excel file.
;
DATA:
GENERATORS, OPTIMIZERS= @OLE( 'SFSP.xls', 'GENERATORS','OPTIMIZERS');
SCENARIOS = @OLE('SFSP.xls','SCENARIOS');
theta, transCoef, Y = @OLE('SFSP.xls','SCENSDATA');
Z, MyBigM = @OLE('SFSP.xls' , 'Z' , 'MyBigM');
alp,bet,del,op,a_f,b_f,d_f = @OLE('SFSP.xls','GENERATORS!D16:J116');
tet,coe,Y_f,Z_f = @OLE('SFSP.xls','GENERATORS!N16:Q116');
precision =@OLE('SFSP.xls','GENERATORS!R18');
!A tolerance that determines the smallest value that we consider as a change in strategy.
In other words, if the change in a firm's strategy is less than this, we count that as a
```

```
no change in the strategy.;

maxRep=@OLE('SFSP.xls','GENERATORS!R19');
!If we do not find an equilibrium after "maxRep" steps, we stop searching for it.;

ENDDATA

SUBMODEL SFSP:
!This is the optimization model solved by a firm to maximize profit, assuming that the
strategy set of all other firms are fixed. ;
@FOR(GENERATORS: @FREE(a));
@FOR(SCENARIOS: @FREE(p));
@FOR(GEN_SCEN: @FREE(x));
@FREE(f);

[obj] MAX = @sum (GENERATORS(i): optimizer(i)* (  @sum (SCENARIOS(s): (theta(s)*p(s))) * q
(i)+ @sum (SCENARIOS(s): (theta(s)*(p(s)* transCoef(s)*x(i,1)-(alpha(i) * (q(i)+x(i,s))+
beta(i)/2 * (q(i)+transCoef(s)*x(i,1))^2 + delta(i)/2 *transCoef(s)^2*x(i,1)^2)))  ) ));
! The objective;

!The constraints include constraints of a generator on his offered supply function and KKT
consitions of the ISO's optimization problem;
@FOR (GENERATORS(i):
[Const_f]   -f + @sum ( SCENARIOS(s): (-e(i,s)+b(i)*theta(s)*x(i,s))) + a(i) + b(i)*q(i) =
0;
      x(i,2)=transCoef(2)*x(i,1);
);

@FOR(GEN_SCEN(i,s):
[Const_p]   -e(i,s) +theta(s)*(-p(s)+a(i)+b(i)*q(i)+(b(i)+d(i))*transCoef(s)*x(i,1))  = 0;
      q(i) + x(i,s) >= 0;
[Const_ebin]      e(i,s)  <= boundary(i,s)*MyBigM;
[Const_qxbin]     q(i)+x(i,s) <= (1-boundary(i,s))*MyBigM;
      @BIN(boundary(i,s));
);


@FOR(SCENARIOS(S):
[Const_p_demand]  theta(s) * (p(s)+ Z* C(s)-Y(s))  = 0;
[Const_C]   theta(s)*(-cQ+C(s)-@sum (GENERATORS(i): x(i,s))) = 0;
);

!Non-optimizing generators should offer their previous offered parameters;
@FOR (GENERATORS(k)|optimizer(k) #EQ# 0 :
      a(k)=a_fixed(k);
      d(k)=d_fixed(k);
);

f - @sum (SCENARIOS(s): (theta(s)*p(s)))  = 0;
cQ - @sum (GENERATORS(h): q(h))  = 0;

ENDSUBMODEL

!Calculations and procedure of the dynamic process to find an equilibrium for each of the
market settings.;
CALC:
@for(ROWS(k):

walpha(k)=0;
wbeta(k)=0;
wdelta(k)=0;
woptimizer(k)=0;
wa_fixed(k)=0;
wb_fixed(k)=0;
wd_fixed(k)=0;
wb(k)=0;
wa(k)=0;
```

```
wd(k)=0;
wq(k)=0;
wx1(k)=0;
wx2(k)=0;
wprofit(k)=0;
wf(k)=0;
wp1(k)=0;
wp2(k)=0;
wwelfare(k)=0;
wrep(k)=0;
wst1(k)=0;
wst2(k)=0;
wtet(k)=0;
wcoe(k)=0;
wY_f(k)=0;
wZ_f(k)=0;
);

!Reading different market settings (i.e. case studies or examples).;
ind=@OLE('SFSP.xls','GENERATORS!L16');
@WHILE (ind #LE# NumProblems:
eq=0;
rep=0;
alp1=alp(2*(ind-1)+1);
alp2=alp(2*(ind-1)+2);
bet1=bet(2*(ind-1)+1);
bet2=bet(2*(ind-1)+2);
del1=del(2*(ind-1)+1);
del2=del(2*(ind-1)+2);
op1=op(2*(ind-1)+1);
op2=op(2*(ind-1)+2);
a_f1=a_f(2*(ind-1)+1);
a_f2=a_f(2*(ind-1)+2);
b_f1=b_f(2*(ind-1)+1);
b_f2=b_f(2*(ind-1)+2);
d_f1=d_f(2*(ind-1)+1);
d_f2=d_f(2*(ind-1)+2);
tet1=tet(2*(ind-1)+1);
tet2=tet(2*(ind-1)+2);
coe1=coe(2*(ind-1)+1);
coe2=coe(2*(ind-1)+2);
Y_f1=Y_f(2*(ind-1)+1);
Y_f2=Y_f(2*(ind-1)+2);
Z_f1=Z_f(2*(ind-1)+1);

@OLE('SFSP.xls','GENERATORS!D2:j2')=alp1,bet1,del1,op1,a_f1,b_f1,d_f1;
@OLE('SFSP.xls','GENERATORS!D3:j3')=alp2,bet2,del2,op2,a_f2,b_f2,d_f2;

@OLE('SFSP.xls','SCENARIOS!C2:E2')=tet1,coe1,Y_f1;
@OLE('SFSP.xls','SCENARIOS!C3:E3')=tet2,coe2,Y_f2;

@OLE('SFSP.xls','OtherParams!B2')=Z_f1;

@for( GENERATORS(i):
      Opt(i)=0;
);

! st1 and st2 records the status of the optimization problems i.e. whether it is found a
global optimal solution or a local optima. These are importnt to ensure that we actually
find a true equilibrium.;
st1=1000;
st2=1000;

@WHILE (eq #LE# 1 #AND# rep#LE#maxRep:
      st1=st2;
      alpha, beta, delta, optimizer, a_fixed, b, d_fixed = @OLE( 'SFSP.xls', 'GENSDATA');
```

```
      theta, transCoef, Y = @OLE('SFSP.xls','SCENSDATA');
      Z = @OLE('SFSP.xls' , 'Z');
      @SOLVE( SFSP);

      @for(GENERATORS(i)| optimizer(i) #EQ# 1 :
            @ifc( a(i) #GE# a_fixed(i)-precision  #AND#  a(i) #LE# a_fixed(i)+precision
#AND# d(i) #GE# d_fixed(i)-precision  #AND#  d(i) #LE# d_fixed(i)+precision:
                  Opt(i)=1;
            @else
                  Opt(i)=0;
             );

            a_fixed(i) = a(i);
            d_fixed(i) = d(i);
      );
      @for(GENERATORS(i):
            @ifc( optimizer(i) #EQ# 1:
                  optimizer(i)=0;
            @else
                  optimizer(i)=1;
             );

      );
      st2=@STATUS();
      eq = @sum(GENERATORS(i): Opt(i));
      @OLE( 'SFSP.xls', 'GENSDATA') = alpha, beta, delta, optimizer, a_fixed, b,d_fixed ;

      rep=rep+1;
      @for(GENERATORS(i):
            profit(i) =  f * q(i)+ @sum (SCENARIOS(s): (theta(s)*(p(s)* x(i,s)-(alpha(i) *
(q(i)+x(i,s))+ beta(i)/2 * (q(i)+x(i,s))^2 + delta(i)/2 *x(i,s)^2)))  ) ;
      );
      !Intermediate output;
      welfare = @sum(SCENARIOS(s): theta(s)*(Y(s)*C(s)-Z/2*C(s)^2-@sum(GENERATORS(i):
alpha(i)*(q(i)+x(i,s))+beta(i)/2*(q(i)+x(i,s))^2+delta(i)/2*x(i,s)^2)));
      @OLE( 'SFSP.xls', 'GENERATORS!L2:N3') = a, d, q;
      @OLE( 'SFSP.xls', 'GENERATORS!O2:O3') = @writefor(GEN_SCEN(i,s)|s #EQ# 1: x(i,s));
      @OLE( 'SFSP.xls', 'GENERATORS!P2:P3') = @writefor(GEN_SCEN(i,s)|s #EQ# 2: x(i,s));
      @OLE( 'SFSP.xls', 'GENERATORS!Q2:Q3') =profit;
      @OLE( 'SFSP.xls', 'GENERATORS!R2:R2') = f;
      @OLE( 'SFSP.xls', 'GENERATORS!S2:T2') = p;
      @OLE( 'SFSP.xls', 'GENERATORS!U2:U2') = welfare;
      @OLE( 'SFSP.xls', 'GENERATORS!V2:V2') = rep;
      @OLE( 'SFSP.xls', 'GENERATORS!W2:W2') = st1;
      @OLE( 'SFSP.xls', 'GENERATORS!X2:X2') = st2;
      @OLE( 'SFSP.xls', 'GENERATORS!Y2:Y2') = @write('WNN');
);

!Final output;
welfare = @sum(SCENARIOS(s): theta(s)*(Y(s)*C(s)-Z/2*C(s)^2-@sum(GENERATORS(i): alpha(i)*
(q(i)+x(i,s))+beta(i)/2*(q(i)+x(i,s))^2+delta(i)/2*x(i,s)^2)));
@OLE( 'SFSP.xls', 'GENERATORS!L2:N3') = a, d, q;
@OLE( 'SFSP.xls', 'GENERATORS!O2:O3') = @writefor(GEN_SCEN(i,s)|s #EQ# 1: x(i,s));
@OLE( 'SFSP.xls', 'GENERATORS!P2:P3') = @writefor(GEN_SCEN(i,s)|s #EQ# 2: x(i,s));
@OLE( 'SFSP.xls', 'GENERATORS!Q2:Q3') =profit;
@OLE( 'SFSP.xls', 'GENERATORS!R2:R2') = f;
@OLE( 'SFSP.xls', 'GENERATORS!S2:T2') = p;
@OLE( 'SFSP.xls', 'GENERATORS!U2:U2') = welfare;
@OLE( 'SFSP.xls', 'GENERATORS!V2:V2') = rep;
@OLE( 'SFSP.xls', 'GENERATORS!W2:W2') = st1;
@OLE( 'SFSP.xls', 'GENERATORS!X2:X2') = st2;
@OLE( 'SFSP.xls', 'GENERATORS!Y2:Y2') = @write('WNN');
i=1;
      walpha(2*(ind-1)+i)=alpha(i);
      wbeta(2*(ind-1)+i)=beta(i);
      wdelta(2*(ind-1)+i)=delta(i);
      woptimizer(2*(ind-1)+i)=optimizer(i);
```

```
        wa_fixed(2*(ind-1)+i)=a_f(2*(ind-1)+i);
        wb_fixed(2*(ind-1)+i)=b_f(2*(ind-1)+i);
        wd_fixed(2*(ind-1)+i)=d_f(2*(ind-1)+i);
        wa(2*(ind-1)+i)=a(i);
        wb(2*(ind-1)+i)=b(i);
        wd(2*(ind-1)+i)=d(i);
        wq(2*(ind-1)+i)=q(i);
        wx1(2*(ind-1)+i)=x(i,1);
        wx2(2*(ind-1)+i)=x(i,2);
        wprofit(2*(ind-1)+i)=profit(i);
        wf(2*(ind-1)+i)=f;
        wp1(2*(ind-1)+i)=p(1);
        wp2(2*(ind-1)+i)=p(2);
        wwelfare(2*(ind-1)+i)=welfare;
        wrep(2*(ind-1)+i)=rep;
        wst1(2*(ind-1)+i)=st1;
        wst2(2*(ind-1)+i)=st2;
        wtet(2*(ind-1)+i)=theta(i);
        wcoe(2*(ind-1)+i)=transCoef(i);
        wY_f(2*(ind-1)+i)=Y(i);
        wZ_f(2*(ind-1)+i)=Z;

i=2;
        walpha(2*(ind-1)+i)=alpha(i);
        wbeta(2*(ind-1)+i)=beta(i);
        wdelta(2*(ind-1)+i)=delta(i);
        woptimizer(2*(ind-1)+i)=optimizer(i);
        wa_fixed(2*(ind-1)+i)=a_f(2*(ind-1)+i);
        wb_fixed(2*(ind-1)+i)=b_f(2*(ind-1)+i);
        wd_fixed(2*(ind-1)+i)=d_f(2*(ind-1)+i);
        wa(2*(ind-1)+i)=a(i);
        wb(2*(ind-1)+i)=b(i);
        wd(2*(ind-1)+i)=d(i);
        wq(2*(ind-1)+i)=q(i);
        wx1(2*(ind-1)+i)=x(i,1);
        wx2(2*(ind-1)+i)=x(i,2);
        wprofit(2*(ind-1)+i)=profit(i);
        wf(2*(ind-1)+i)=f;
        wp1(2*(ind-1)+i)=p(1);
        wp2(2*(ind-1)+i)=p(2);
        wwelfare(2*(ind-1)+i)=welfare;
        wrep(2*(ind-1)+i)=rep;
        wst1(2*(ind-1)+i)=st1;
        wst2(2*(ind-1)+i)=st2;
        wtet(2*(ind-1)+i)=theta(i);
        wcoe(2*(ind-1)+i)=transCoef(i);
        wY_f(2*(ind-1)+i)=Y(i);
        wZ_f(2*(ind-1)+i)=Z;

ind=ind+1;
@OLE( 'SFSP.xls', 'OUT_WNN!B2:E101') = wtet,wcoe,wY_f,wZ_f;
@OLE( 'SFSP.xls', 'OUT_WNN!G2:AA101') =  walpha ,wbeta ,wdelta ,woptimizer ,wa_fixed
,wb_fixed,wd_fixed ,wa ,wb ,wd ,wq ,wx1 ,wx2 ,wprofit,wf,wp1,wp2,wwelfare,wrep,wst1,wst2;
);

ENDCALC
@WARN('LINGO Finished',1#GE#0);
END
```