The Costs of Counterparty Risk in Long Term Contracts
Code Guide - Section 5

Michael Duarte Gonçalves

July 4, 2025

Overview

This section extends the baseline analysis with counterparty risk (CPR) by introducing **public guarantees**. Public guarantees are modeled as a policy tool that allows sellers to contract at the *same equilibrium contract price and quantity* as would be achieved in a market with $\gamma = 0$ (i.e., no counterparty risk), but for any value of γ. This simulates a scenario in which the government or a public entity fully backs contracts, eliminating the impact of counterparty risk on sellers' willingness to contract.

Contents

1 Modeling Markets With Counterparty Risk (CPR) - Public Guarantees

1.1 Conceptual Approach

The key idea is to **fix the equilibrium contract price and quantity** at the values determined under $\gamma = 0$ for all values of γ. This means that, regardless of the actual counterparty risk in the market, sellers behave as if they face no such risk when it comes to contract settlement and supply decisions. The analysis then calculates welfare and profit outcomes under this hypothetical policy.

1.2 Numerical Methods

1.2.1 Extract $\gamma = 0$ Equilibrium

```
f_equilibrium_gamma_0_df <- wind_solar_proj_2022_long_baseline |>
  filter(gamma == 0) |>
  distinct(theta, f_equilibrium) |>
  mutate(xf_equilibrium = x * f_equilibrium)
```

Explanation: This code filters the baseline dataset to keep only rows where $\gamma = 0$ (no counterparty risk). It then selects unique pairs of demand scenario (`theta`) and equilibrium contract price (`f_equilibrium`), and computes the total contract value (`xf_equilibrium`), which is simply xf^*.

1.2.2 Replicate $\gamma = 0$ Data for All γ

```
public_guarantees <- crossing(
  gamma = wind_solar_proj_2022_long_baseline$gamma,
  gamma_0_data
) |>
  relocate(gamma, .after = theta)
```

Explanation: This code creates a new dataset where the $\gamma = 0$ equilibrium outcomes are assigned to every value of γ. This simulates the effect of public guarantees, which ensure that contract terms are unaffected by counterparty risk.

1.2.3 Integral Calculation for Counterparty Risk

The core of the risk adjustment lies in computing the expected loss from counterparty default. This is implemented through:

```
compute_integral_vec <- function(f_vec, alpha, beta) {
  vapply(f_vec, function(f) {
    integrate(function(p) (f - p) * dbeta(p, alpha, beta),
              lower = 0, upper = f)$value
  }, numeric(1))
}
```

Mathematical Foundation: This function computes the expected loss per unit contract quantity if the buyer defaults:

$$\text{Integral} = \int_0^{f^*} (f - p) \cdot \phi(p; \alpha, \beta) \, dp$$

where:

- f: Contract price (strike price)
- p: Realized market price (random variable)
- $\phi(p; \alpha, \beta)$: Beta probability density function (PDF)

1.2.4 Risk Adjustment in Public Guarantees

The integral is applied to compute risk costs while maintaining $\gamma = 0$ contract terms:

```
public_guarantees <- public_guarantees |>
  mutate(
    x_integral = x * compute_integral_vec(f_equilibrium, alpha, beta),
    q_i_mwh_lambda_gamma_x_integral = q_i_mwh * lambda * gamma * x_
       integral
  ) |>
  arrange(theta, gamma, xf_max_cpr)
```

Variable Explanations:

- `x_integral`: We create a new variable $x \cdot \int_0^{f^*} (f - p) \cdot \phi(p; \alpha, \beta)\, dp$
  - x is the scaling factor — $x = 60$
  - Multiplies integral result by quantity to get monetary loss
- `q_i_mwh_lambda_gamma_x_integral`: Total risk cost (EUR)
  - `q_i_mwh`: Generation capacity (MWh)
  - `lambda`: Per-unit social cost, $\lambda = 0.3$. This is a fixed parameter and is created in the *Readme* file from Section 1.
  - `gamma`: Share of opportunistic buyers γ
  - Scales expected loss by capacity and risk parameters
  - Mathematical expression: $q_i x \cdot \lambda \cdot \gamma \cdot \int_0^{f^*} (f - p) \cdot \phi(p; \alpha, \beta)\, dp$

1.2.5 Welfare W^0

```
W_0_pg <- public_guarantees |>
  filter(gamma == 0, profits_sp_no_cpr >= 0) |>
  group_by(theta) |>
  summarise(
    welfare_0_eur = sum(x_q_exp_p_total_costs - r, na.rm = TRUE),
    .groups = "drop"
  )
```

Explanation: Computes baseline welfare for $\gamma = 0$ scenario:

- `filter()`: Selects projects with no counterparty risk ($\gamma = 0$) and non-negative profits
- `sum(x_q_exp_p_total_costs - r)`: Calculates welfare as total revenue minus costs, $\sum_i xq_i \cdot \mathbb{E}(p) - C(k_i) - r_i$
- θ-specific baseline welfare (`welfare_0_eur`)

1.2.6 Welfare $W(\gamma)$

```
W_gamma_pg <- public_guarantees |>
  filter(xf_max_cpr <= xf_equilibrium & cumulative_capacity < theta) |>
  group_by(gamma, theta) |>
  summarise(
    x_q_exp_p_total_costs_R          = sum(x_q_exp_p_total_costs_R, na.
       rm = TRUE),
    q_i_mwh_lambda_gamma_x_integral = sum(q_i_mwh_lambda_gamma_x_
       integral, na.rm = TRUE),
    welfare_gamma_eur = x_q_exp_p_total_costs_R - q_i_mwh_lambda_gamma_
       x_integral,
```

```
    .groups = "drop"
  ) |>
  ungroup()
```

Explanation:

1. **Filtering Eligible Projects:**
   - `xf_max_cpr <= xf_equilibrium`: Only projects with a maximum contract price at or below the equilibrium price are included.
   - `cumulative_capacity < theta`: Ensures only projects within the required demand scenario (θ) are considered.

2. **Key Aggregated Quantities:**
   - `x_q_exp_p_total_costs_R`:

     $$\text{x\_q\_exp\_p\_total\_costs\_R} = \sum_{i \in S_{\gamma,\theta}} \left[x \cdot q_i \cdot \mathbb{E}[p] - C(k_i) - \cancel{R_i(f^*, \gamma)}\right]$$

     where $S_{\gamma,\theta}$ is the set of eligible projects for each (γ, θ). Remind that $R_i(f^*, \gamma)$ becomes zero when $\gamma = 0$.
   - `q_i_mwh_lambda_gamma_x_integral`:

     $$\sum_{i \in S_{\gamma,\theta}} q_i \cdot \lambda \cdot \gamma \cdot \left(x \int_0^{f^*} (f - p)\phi(p)\, dp\right)$$

3. **Welfare Calculation:**
   - The welfare for each (γ, θ) is then calculated as:

     $$\text{welfare\_gamma\_eur} = \text{x\_q\_exp\_p\_total\_costs\_R} - \text{q\_i\_mwh\_lambda\_gamma\_x\_integral}$$

     Or explicitly,

     $$W_\gamma^{PG}(\theta) = \sum_{i \in S_{\gamma,\theta}} \left[x \cdot q_i \cdot \mathbb{E}[p] - C(k_i) - \cancel{R_i(f^*, \gamma)}\right] - \sum_{i \in S_{\gamma,\theta}} q_i \cdot \lambda \cdot \gamma \cdot \left(x \int_0^{f^*} (f - p)\phi(p)\, dp\right)$$

     This reflects the total market welfare under public guarantees, accounting for both standard profits and the additional deadweight loss from counterparty risk.

1.2.7 Extract Baseline $W(\gamma = 0)$ for Comparison

```
W_gamma_0_pg <- W_gamma_pg |>
  filter(gamma == 0) |>
  select(theta, welfare_gamma_0_eur = welfare_gamma_eur)
```

Explanation: This block filters the previously computed welfare dataset to retain only the rows where $\gamma = 0$, i.e., the baseline (no counterparty risk) scenario. It then renames the welfare column for clarity as `welfare_gamma_0_eur`. This baseline will be used as a reference for ratio and gap calculations.

1.2.8 Extract Equilibrium Quantities and Prices at $\gamma = 0$

```
eq_gamma_0_pg <- equilibrium_public_g |>
  filter(gamma == 0) |>
  select(
    theta,
    equilibrium_quantity_gamma_0 = equilibrium_quantity,
    equilibrium_price_gamma_0 = xf_equilibrium
  )
```

Explanation: This code selects the equilibrium quantity and price for each demand scenario (θ) at $\gamma = 0$. These values serve as the reference points for calculating ratios of equilibrium outcomes under different risk levels.

1.2.9 Combine All Metrics into Final Welfare Table

```
welfare_dataset_pg <- equilibrium_public_g |>
  left_join(W_gamma_pg,     by = c("gamma", "theta")) |>
  left_join(W_gamma_0_pg,   by = "theta") |>
  left_join(W_0_pg,         by = "theta") |>
  left_join(eq_gamma_0_pg,  by = "theta") |>
  mutate(
    equilibrium_price_eur      = xf_equilibrium,
    equilibrium_quantity_mw    = equilibrium_quantity,
    W_0 = if_else(gamma == 0, welfare_0_eur, NA_real_),
    welfare_gain_vs_baseline_meur = (welfare_gamma_eur - welfare_0_eur)
        / 1e6,
    welfare_ratio_percent         = (welfare_gamma_eur / welfare_gamma_
       0_eur) * 100,
    welfare_gap_million_eur       = (welfare_gamma_0_eur - welfare_
       gamma_eur) / 1e6,
    eq_quantity_ratio_percent = (equilibrium_quantity / equilibrium_
       quantity_gamma_0) * 100,
    eq_price_ratio_percent     = (xf_equilibrium / equilibrium_price_
       gamma_0) * 100
  ) |>
  select(
    gamma, theta,
    equilibrium_price_eur,
    eq_price_ratio_percent,
    equilibrium_quantity_mw,
    eq_quantity_ratio_percent,
    W_0,
    welfare_gamma_eur,
    welfare_ratio_percent,
    welfare_gap_million_eur,
    welfare_gain_vs_baseline_meur
  ) |>
  arrange(theta, gamma)
```

Explanation: This block merges all the previously computed reference and scenario-specific metrics into a single comprehensive dataset. It then calculates several comparative statistics:

- `welfare_gain_vs_baseline_meur`: Welfare gain (in million EUR) versus the baseline.
- `welfare_ratio_percent`: Welfare as a percentage of the $\gamma = 0$ baseline.

- welfare_gap_million_eur: Welfare loss (in million EUR) compared to the baseline.
- eq_quantity_ratio_percent, eq_price_ratio_percent: Equilibrium quantity and price as a percentage of the $\gamma = 0$ baseline.

The final table is arranged for easy comparison across all (γ, θ) combinations.

1.2.10 Prepare Baseline Welfare for Ratio Calculations

```
welfare_dataset_baseline <- welfare_dataset_baseline %>%
  select(gamma, theta, welfare_gamma_eur) |>
  rename(welfare_gamma_eur_baseline = welfare_gamma_eur)
```

Explanation: This code extracts and renames the baseline welfare values from the baseline dataset. The baseline dataset here refers to the results obtained from those computed in Section 4. This step ensures that welfare under public guarantees can be directly compared to the baseline (non-guarantee) for each (γ, θ) pair.

1.2.11 Compute Welfare Ratios vs. Baseline

```
welfare_ratios_pg <- W_gamma_pg |>
  left_join(welfare_dataset_baseline, by = c("gamma", "theta")) |>
  mutate(
    welfare_ratio = (welfare_gamma_eur / welfare_gamma_eur_baseline) *
       100
  ) |>
  select(theta, gamma, everything()) |>
  arrange(gamma, theta)
```

Explanation: This block joins the public guarantee welfare results with the baseline results for each scenario, then computes the welfare ratio (as a percentage) of public guarantees relative to the baseline. This allows for a direct, scenario-by-scenario comparison of policy effectiveness.

1.2.12 Create Wide-Format Table for Comparison Across θ

```
welfare_ratios_wide_pg <- welfare_ratios_pg |>
  select(gamma, theta, welfare_ratio) |>
  pivot_wider(
    names_from = theta,
    values_from = welfare_ratio,
    names_prefix = "theta_"
  )
```

Explanation: This final step reshapes the welfare ratio data into a wide format, with each column corresponding to a different demand scenario (θ). This is useful for presenting results in tables or heatmaps, making cross-scenario comparisons straightforward.

1.2.13 Profit Calculation under Public Guarantees

```
profits_by_gamma_theta_pg <- public_guarantees |>
  filter(xf_max_cpr <= xf_equilibrium & cumulative_capacity < theta) |>
  group_by(gamma, theta) |>
```

```
  summarise(
    seller_profits_eur_pg = sum(profit_cpr_contracts, na.rm = TRUE),
    .groups = "drop"
  ) |>
  ungroup() |>
  arrange(theta, gamma, seller_profits_eur_pg)
```

Explanation: This code calculates total seller profits under public guarantees:

- `filter()`: Selects projects within equilibrium constraints
- `sum(profit_cpr_contracts)`: Aggregates profits from contracts
- Output: Total seller profits per (γ, θ) scenario

```
welfare_dataset_pg <- welfare_dataset_pg |>
  left_join(profits_by_gamma_theta_pg, by = c("gamma", "theta")) |>
  mutate(
    buyer_profits_eur_pg = welfare_gamma_eur - seller_profits_eur_pg,
    buyer_profit_share_percent = round((buyer_profits_eur_pg / welfare_
       gamma_eur) * 100, 2),
    seller_profit_share_percent = round((seller_profits_eur_pg /
       welfare_gamma_eur) * 100, 2)
  ) |>
  arrange(theta, gamma)
```

Explanation: This block integrates profit metrics into the welfare dataset:

- `left_join()`: Merges seller profits into welfare dataset
- `buyer_profits_eur_pg`: Computed as $W(\gamma) - \Pi_S$
- Profit shares calculated as percentages of total welfare:
  - Buyer share: $\frac{\Pi_B}{W(\gamma)} \times 100$
  - Seller share: $\frac{\Pi_S}{W(\gamma)} \times 100$

```
profits_by_gamma_theta_pg <- welfare_dataset_pg |>
  select(theta, gamma, starts_with("buyer"), starts_with("seller"))
```

Explanation: This final step creates a specialized profit analysis table:

- `select()`: Extracts profit-related columns
- Includes:
  - Buyer profits (EUR) and profit share (%)
  - Seller profits (EUR) and profit share (%)
- Maintains θ and γ as primary identifiers

1.2.14 Visualization of Results under Public Guarantees

To provide a clear and accessible summary of the main results under public guarantees, we generate a series of plots for contract prices, investment quantities, welfare, and profit distribution. These visualizations are created using the `plot_line_by_gamma` and `plot_profit_metric` functions, which are defined in Section 4 and reused here for consistency.

```
# Plots for Public Guarantees


# Contract Prices

# Equilibrium Price Plot
plot_line_by_gamma(
  data         = welfare_dataset_pg,
  x            = "gamma",
  y            = "equilibrium_price_eur",
  group_var    = "theta",
  color_var    = "theta",
  color_label = expression(theta),
  y_lab        = "Contract Price (EUR/MW)",
  x_lab        = expression(gamma),
  colors       = theme_palette_theta,
  filename     = "01_equilibrium_price_vs_gamma_theta_publicg.pdf",
  folder       = with_pg_fig_path,
  legend_position = c(0.08, 0.85)
)

# Equilibrium Quantity Plot
plot_line_by_gamma(
  data         = welfare_dataset_pg,
  x            = "gamma",
  y            = "equilibrium_quantity_mw",
  group_var    = "theta",
  color_var    = "theta",
  color_label = expression(theta),
  y_lab        = "Investment (MW)",
  x_lab        = expression(gamma),
  colors       = theme_palette_theta,
  filename     = "02_equilibrium_quantity_vs_gamma_theta_publicg.pdf",
  folder       = with_pg_fig_path,
  legend_position = c(0.08, 0.85)
)

# Welfare Plot (in million EUR)
plot_line_by_gamma(
  data              = welfare_dataset_pg,
  x                 = "gamma",
  y                 = "welfare_gamma_eur",
  group_var         = "theta",
  color_var         = "theta",
  color_label       = expression(theta),
  y_lab             = "Welfare (M-EUR)",
  x_lab             = expression(gamma),
  y_scale_million = TRUE,
  colors            = theme_palette_theta,
  filename          = "03_welfare_vs_gamma_theta_publicg.pdf",
  folder            = with_pg_fig_path,
```

```r
  legend_position = c(0.08, 0.85)
)

# --------------- Profit Metrics ---------------- #

# Seller Profit (M-EUR)
plot_profit_metric(
  data              = welfare_dataset_pg,
  x_var             = "gamma",
  x_label           = expression(gamma),
  y_var             = "seller_profits_eur_pg",
  group_var         = "theta",
  color_var         = "theta",
  color_label       = expression(theta),
  y_label           = "Seller Profit (M-EUR)",
  y_scale_million = TRUE,
  save_path         = with_pg_fig_path,
  file_name         = "04_seller_profit_vs_gamma_theta_publicg.pdf",
  legend_position = c(0.08, 0.85)
)

# Buyer Profit (M-EUR)
plot_profit_metric(
  data              = welfare_dataset_pg,
  x_var             = "gamma",
  y_var             = "buyer_profits_eur_pg",
  group_var         = "theta",
  color_var         = "theta",
  color_label       = expression(theta),
  y_label           = "Buyer Profit (M-EUR)",
  x_lab             = expression(gamma),
  y_scale_million = TRUE,
  save_path         = with_pg_fig_path,
  file_name         = "05_buyer_profit_vs_gamma_theta_publicg.pdf",
  legend_position = c(0.08, 0.70)
)

# Seller Profit Share (%)
plot_profit_metric(
  data                = welfare_dataset_pg,
  x_var               = "gamma",
  y_var               = "seller_profit_share_percent",
  group_var           = "theta",
  color_var           = "theta",
  color_label         = expression(theta),
  y_label             = "Seller Profit Share (%)",
  x_lab               = expression(gamma),
  y_scale_percent     = TRUE,
  save_path           = with_pg_fig_path,
  file_name           = "06_seller_profit_share_vs_gamma_theta_publicg.
    pdf",
  legend_position = c(0.08, 0.85)
)

# Buyer Profit Share (%)
plot_profit_metric(
  data                = welfare_dataset_pg,
  x_var               = "gamma",
```

```
  y_var                = "buyer_profit_share_percent",
  group_var            = "theta",
  color_var            = "theta",
  color_label          = expression(theta),
  y_label              = "Buyer Profit Share (%)",
  x_lab                = expression(gamma),
  y_scale_percent      = TRUE,
  save_path            = with_pg_fig_path,
  file_name            = "07_buyer_profit_share_vs_gamma_theta_publicg.pdf
     ",
  legend_position = c(0.08, 0.70)
)
```

1.2.15 Generated Figures

Table 1: Summary of Figures Generated for Public Guarantees Analysis

# Plot	Description
# 01	Equilibrium contract price as a function of γ and θ
# 02	Equilibrium investment (MW) as a function of γ and θ
# 03	Total welfare (in million €) as a function of γ and θ
# 04	Seller profit (in million €) as a function of γ and θ
# 05	Buyer profit (in million €) as a function of γ and θ
# 06	Seller profit share (%) as a function of γ and θ
# 07	Buyer profit share (%) as a function of γ and θ

1.2.16 Plotting Functions

The plots are generated using the following reusable functions, originally defined in Section 4:

- `plot_line_by_gamma`: For line plots of contract price, investment, and welfare versus γ.
- `plot_profit_metric`: For line plots of seller and buyer profits and their shares versus γ.

These functions take as input the processed results dataset (`welfare_dataset_pg`) and output publication-quality PDF figures to the specified folder.

1.2.17 Exporting Public Guarantees Results

The final step of the analysis is to systematically save all key datasets related to the public guarantees scenarios into a single, well-organized Excel workbook. This ensures that all results are easily accessible for further analysis, sharing, or reporting.
What this code does:

1. **Dataset Collection:** A list called `datasets_guarantees` is created, containing all major datasets from the public guarantees analysis. These include project-level data, equilibrium prices and quantities, welfare metrics, welfare ratios, and profit distributions, each labeled with a descriptive sheet name.
2. **Filtering:** Each dataset is filtered by the risk parameter γ using a function `filter_by_gamma` (assumed to be defined elsewhere). This step can reduce data size or focus the output on specific risk levels.

3. **Workbook Preparation:** The output Excel filename and path are constructed.
4. **Workbook Creation:** A new Excel workbook is initialized.
5. **Sheet Population:** For each dataset, a new worksheet is added to the workbook, named according to the dataset's label. The first row and column are frozen for easier navigation, and the data is written to the sheet.
6. **Saving:** The workbook is saved to the specified path, overwriting any existing file with the same name.

Purpose: This process organizes and exports all relevant results from the public guarantees analysis into a single, structured Excel file. The resulting file is suitable for further quantitative analysis, easy sharing with collaborators, or inclusion in reports and presentations.

1.2.18 Created Tables

Table 2: Variables in `Wind Solar Projects (Public G.)` Sheet

Variable	Description
projectname	Project name
capacity	Installed capacity (MW)
avgcapacityfactor	Average capacity factor
type	Technology type (Wind/Solar)
hours	Annual full-load hours
power_kw	Installed power (kW)
q_i_kwh	Annual generation (kWh)
q_i_mwh	Annual generation (MWh)
v_q_i_mwh	Consumer valuation (MWh)
c_inv	Investment cost per MW
c_om	O&M cost per MW
total_cost	$C(k_i)$
avg_cost_euro_kwh	Average cost (EUR/kWh)
avg_cost_euro_mwh	Average cost (EUR/MWh)
r_0	r_0
r	r_i
profits_sp_no_cpr	Π_S^0
theta	Demand scenario θ (MW)
gamma	Share of opportunistic buyers γ
f_c_cpr	f_c
f_spot_cpr	f_{spot}
f_upper	Second root f_{upper}, if any
f_upper_message	Says if `f_upper`$> \mathbb{E}(p)$ or `f_upper`$\leq \mathbb{E}(p)$
f_max_cpr	f_{max}
f_equilibrium	f^*
R_f_equilibrium_cpr	$R_i(f^*, \gamma)$
xf_c_cpr	xf_c
xf_spot_cpr	xf_{spot}
xf_upper	Second root xf_{upper}, if any
xf_max_cpr	xf_{max}
xf_equilibrium	xf^*
equilibrium_quantity	q^*
x_q_exp_p_total_costs	$xq_i \cdot \mathbb{E}(p) - C(k_i)$
x_q_exp_p_total_costs_R	$xq_i \cdot \mathbb{E}(p) - C(k_i) - R_i(f^*, \gamma)$
cumulative_production	Cumulative energy production (MWh)
cumulative_capacity	Cumulative capacity (MW)
profit_cpr_contracts	Profits under CPR contracts, using f^*
x_integral	Expected loss: $\int_0^{f^*} (f - p) \cdot \phi(p; \alpha, \beta)\, dp$
q_i_mwh_lambda_gamma_x_integral	$q_i x \cdot \lambda \cdot \gamma \int_0^{f^*} (f - p) \cdot \phi(p; \alpha, \beta)\, dp$

Table 3: Variables in `Equilibrium P. & Q.` Sheet

Variable	Description
gamma	Share of opportunistic buyers γ
theta	Demand scenario (MW) - θ
xf_equilibrium	Equilibrium contract price xf^* (EUR/MWh)
equilibrium_quantity	Total contracted quantity at equilibrium (MW), q^*

Table 4: Variables in `Welfare Public G.` Sheet

Variable	Description
gamma	Share of opportunistic buyers γ
theta	Demand scenario (MW) - θ
equilibrium_price_eur	Equilibrium contract price xf^* (EUR/MWh)
eq_price_ratio_percent	Price as % of $\gamma = 0$ baseline
equilibrium_quantity_mw	Equilibrium contracted quantity (MW), q^*
eq_quantity_ratio_percent	Quantity as % of $\gamma = 0$ baseline
W_0	Welfare W^0
welfare_gamma_eur	Welfare $W(\gamma)$ for given θ (EUR)
welfare_ratio_percent	Welfare as % of $W(\gamma = 0)$ baseline
welfare_gap_million_eur	Welfare gap vs. baseline (million EUR)
welfare_gain_vs_baseline_meur	Welfare gain vs. baseline (million EUR)
seller_profits_eur_pg	Seller profits under public guarantees (EUR)
buyer_profits_eur_pg	Buyer profits under public guarantees (EUR)
buyer_profit_share_percent	Buyer share of welfare (%)
seller_profit_share_percent	Seller share of welfare (%)

Table 5: Variables in `Welfare Baseline` Sheet

Variable	Description
gamma	Share of opportunistic buyers γ
theta	Demand scenario (MW) - θ
welfare_gamma_eur_baseline	Baseline welfare (EUR), no public guarantees (*c.f.* Section 4)

Table 6: Variables in `Welfare Ratios Pub.G.` Sheet

Variable	Description
theta	Demand scenario (MW) - θ
gamma	Share of opportunistic buyers γ
x_q_exp_p_total_costs_R	$xq_i \cdot \mathbb{E}(p) - C(k_i) - R_i(f^*, \gamma)$
q_i_mwh_lambda_gamma_x_integral	$q_i x \cdot \lambda \cdot \gamma \int_0^{f^*} (f - p) \cdot \phi(p; \alpha, \beta)\, dp$
welfare_gamma_eur	Welfare under public guarantees (million EUR)
welfare_gamma_eur_baseline	Baseline welfare - Section 4 results (million EUR)
welfare_ratio	$W(\gamma)^{PG} / W(\gamma)^{Baseline}$ in percent

Table 7: Variables in `Welfare Ratios Pub.G. (Wide)` Sheet

Variable	Description
gamma	Share of opportunistic buyers γ
theta_2500	Welfare ratio ($\frac{W^{PG}}{W^{Baseline}}$ — %) for $\theta = 2500$
theta_3500	Welfare ratio ($\frac{W^{PG}}{W^{Baseline}}$ — %) for $\theta = 3500$
theta_4500	Welfare ratio ($\frac{W^{PG}}{W^{Baseline}}$ — %) for $\theta = 4500$

Table 8: Variables in `Profits (P.G.)` Sheet

Variable	Description
theta	Demand scenario (MW) - θ
gamma	Share of opportunistic buyers γ
buyer_profits_eur_pg	Buyer profits under P.G. (million EUR)
buyer_profit_share_percent	Buyer profits as % of welfare
seller_profits_eur_pg	Seller profits (million EUR)
seller_profit_share_percent	Seller profits as % of welfare