

# Problem Formulations Specification for Quantum Advantage Benchmarks

## Multi-Period Crop Rotation Optimization

OQI-UC002-DWave Project

December 2025

### Abstract

This document provides a comprehensive specification of all problem formulations used in the quantum advantage benchmarks. It enables direct comparison of results across different experiments by clearly defining the objective functions, constraints, parameters, and solution methods for each benchmark. All experiments share a common base formulation but differ in problem size, crop representation, and decomposition strategy.

## Contents

<b>1 Common Mathematical Framework</b>	<b>2</b>
1.1 Decision Variables . . . . .	2
1.2 Problem Data . . . . .	2
1.3 Objective Function Components . . . . .	2
1.3.1 Component 1: Base Agricultural Benefit (Linear) . . . . .	2
1.3.2 Component 2: Temporal Rotation Synergies (Quadratic) . . . . .	2
1.3.3 Component 3: Spatial Neighbor Interactions (Quadratic) . . . . .	3
1.3.4 Component 4: Diversity Bonus (Linear) . . . . .	3
1.3.5 Component 5: One-Hot Penalty (Quadratic) . . . . .	3
1.4 Constraints . . . . .	3
1.5 Common Parameters . . . . .	3
1.6 Rotation Matrix Construction . . . . .	3
<b>2 Formulation A: Native 6 Families</b>	<b>3</b>
2.1 Overview . . . . .	4
2.2 Crop Families . . . . .	4
2.3 Solution Methods . . . . .	4
2.3.1 Ground Truth: Gurobi . . . . .	4
2.3.2 Clique Decomposition (QPU) . . . . .	4
2.3.3 Spatial-Temporal Decomposition (QPU) . . . . .	4
2.4 Result Data Files . . . . .	5
<b>3 Formulation B: 27 Foods → 6 Families (Aggregated)</b>	<b>5</b>
3.1 Overview . . . . .	5
3.2 Food-to-Family Aggregation . . . . .	5
3.3 Three-Level Hierarchical Approach . . . . .	5
3.4 Solution Methods . . . . .	6
3.4.1 Ground Truth: Gurobi . . . . .	6
3.4.2 Hierarchical QPU . . . . .	6
3.5 Result Data Files . . . . .	6

<b>4 Comparison of Formulations</b>	<b>6</b>
4.1 Key Differences . . . . .	6
4.2 What Is Identical . . . . .	6
4.3 When Results Are Comparable . . . . .	7
<b>5 Benchmark Result Summary</b>	<b>7</b>
5.1 Formulation A Results (Native 6 Families) . . . . .	7
5.2 Formulation B Results (Aggregated 27→6) . . . . .	7
5.3 Significant Scenarios Results . . . . .	7
<b>6 Quantum Advantage Metrics</b>	<b>8</b>
6.1 Speedup Calculation . . . . .	8
6.2 Optimality Gap . . . . .	8
6.3 Where Quantum Advantage Occurs . . . . .	8
<b>7 Reproducibility</b>	<b>8</b>
7.1 Random Seeds . . . . .	8
7.2 Running Benchmarks . . . . .	9
7.3 D-Wave Configuration . . . . .	9
<b>8 Conclusion</b>	<b>9</b>

# 1 Common Mathematical Framework

All benchmarks in this project solve variants of the **Multi-Period Crop Rotation Optimization Problem**. This section defines the common mathematical framework.

## 1.1 Decision Variables

$$Y_{f,c,t} \in \{0, 1\} \quad \forall f \in \mathcal{F}, c \in \mathcal{C}, t \in \mathcal{T} \quad (1)$$

Where:

- $\mathcal{F}$  = set of farms (plots), with  $|\mathcal{F}| = F$
- $\mathcal{C}$  = set of crops/families, with  $|\mathcal{C}| = C$
- $\mathcal{T}$  = set of time periods, with  $|\mathcal{T}| = T = 3$

**Interpretation:**  $Y_{f,c,t} = 1$  means farm  $f$  grows crop  $c$  in period  $t$ .

**Total Variables:**  $N = F \times C \times T$

## 1.2 Problem Data

Symbol	Description
$A_f$	Land availability (hectares) at farm $f$
$A_{\text{total}}$	Total area: $\sum_{f \in \mathcal{F}} A_f$
$B_c$	Benefit score for crop $c$ (weighted combination of nutritional value, sustainability, affordability, environmental impact)
$R_{c_1,c_2}$	Rotation synergy matrix: benefit of growing $c_2$ after $c_1$
$\mathcal{N}(f)$	Set of $k$ -nearest spatial neighbors of farm $f$

## 1.3 Objective Function Components

The objective function is identical across all benchmarks:

$$\mathcal{O} = \mathcal{O}_{\text{benefit}} + \mathcal{O}_{\text{rotation}} + \mathcal{O}_{\text{spatial}} + \mathcal{O}_{\text{diversity}} - \mathcal{O}_{\text{penalty}} \quad (2)$$

### 1.3.1 Component 1: Base Agricultural Benefit (Linear)

$$\mathcal{O}_{\text{benefit}} = \sum_{f \in \mathcal{F}} \sum_{c \in \mathcal{C}} \sum_{t \in \mathcal{T}} \frac{B_c \cdot A_f}{A_{\text{total}}} \cdot Y_{f,c,t} \quad (3)$$

**Purpose:** Maximize total agricultural benefit weighted by land area.

### 1.3.2 Component 2: Temporal Rotation Synergies (Quadratic)

$$\mathcal{O}_{\text{rotation}} = \sum_{f \in \mathcal{F}} \sum_{t=2}^T \sum_{c_1 \in \mathcal{C}} \sum_{c_2 \in \mathcal{C}} \frac{\gamma_{\text{rot}} \cdot R_{c_1,c_2} \cdot A_f}{A_{\text{total}}} \cdot Y_{f,c_1,t-1} \cdot Y_{f,c_2,t} \quad (4)$$

**Purpose:** Reward beneficial crop sequences (e.g., legumes before grains) and penalize harmful sequences (e.g., same crop in consecutive periods).

### 1.3.3 Component 3: Spatial Neighbor Interactions (Quadratic)

$$\mathcal{O}_{\text{spatial}} = \sum_{(f_1, f_2) \in \mathcal{E}} \sum_{t \in \mathcal{T}} \sum_{c_1 \in \mathcal{C}} \sum_{c_2 \in \mathcal{C}} \frac{\gamma_{\text{spatial}} \cdot R_{c_1, c_2} \cdot 0.3}{A_{\text{total}}} \cdot Y_{f_1, c_1, t} \cdot Y_{f_2, c_2, t} \quad (5)$$

Where  $\mathcal{E}$  is the edge set of the spatial neighbor graph.

**Purpose:** Model pest management, pollination, and resource sharing between neighboring farms.

### 1.3.4 Component 4: Diversity Bonus (Linear)

$$\mathcal{O}_{\text{diversity}} = \sum_{f \in \mathcal{F}} \sum_{c \in \mathcal{C}} \beta_{\text{div}} \cdot \mathbf{1} \left[ \sum_{t \in \mathcal{T}} Y_{f, c, t} > 0 \right] \quad (6)$$

**Purpose:** Encourage crop diversity across the rotation cycle.

### 1.3.5 Component 5: One-Hot Penalty (Quadratic)

$$\mathcal{O}_{\text{penalty}} = \sum_{f \in \mathcal{F}} \sum_{t \in \mathcal{T}} \lambda_{\text{penalty}} \cdot \left( \sum_{c \in \mathcal{C}} Y_{f, c, t} - 1 \right)^2 \quad (7)$$

**Purpose:** Soft constraint enforcing exactly one crop per farm per period.

## 1.4 Constraints

$$\sum_{c \in \mathcal{C}} Y_{f, c, t} \leq 2 \quad \forall f \in \mathcal{F}, t \in \mathcal{T} \quad (8)$$

**Purpose:** Hard constraint allowing at most 2 crops per farm per period (the soft penalty in the objective pushes toward exactly 1).

## 1.5 Common Parameters

Parameter	Symbol	Default Value
Rotation weight	$\gamma_{\text{rot}}$	0.2
Spatial weight	$\gamma_{\text{spatial}}$	$\gamma_{\text{rot}} \times 0.5 = 0.1$
Diversity bonus	$\beta_{\text{div}}$	0.15
One-hot penalty	$\lambda_{\text{penalty}}$	3.0
Spatial neighbors	$k$	4
Frustration ratio	-	0.7 (70% negative synergies)
Negative strength	-	-0.8

## 1.6 Rotation Matrix Construction

The rotation synergy matrix  $R$  is constructed with seed 42 for reproducibility:

$$R_{c_1, c_2} = \begin{cases} -1.2 & \text{if } c_1 = c_2 \quad (\text{self-rotation penalty}) \\ \text{Uniform}(-0.96, -0.24) & \text{with prob. 0.7 (antagonistic)} \\ \text{Uniform}(0.02, 0.20) & \text{with prob. 0.3 (synergistic)} \end{cases} \quad (9)$$

This creates a *frustrated* system characteristic of NP-Hard optimization problems.

## 2 Formulation A: Native 6 Families

Used by: `statistical_comparison_test.py`

## 2.1 Overview

Property	Value
Crops/Families	$C = 6$ (native families)
Periods	$T = 3$
Farm sizes tested	$F \in \{5, 10, 15, 20, 25\}$
Variables	$N = F \times 6 \times 3 = \{90, 180, 270, 360, 450\}$
Aggregation	None (direct family assignment)

## 2.2 Crop Families

Index	Family Name
1	Legumes
2	Grains
3	Vegetables (Leafy)
4	Root Vegetables
5	Fruits
6	Proteins/Other

## 2.3 Solution Methods

### 2.3.1 Ground Truth: Gurobi

- Timeout: 300 seconds (5 minutes)
- MIP Gap: 10% tolerance
- MIP Focus: 1 (find feasible solutions quickly)
- Improve Start Time: 30 seconds

### 2.3.2 Clique Decomposition (QPU)

Decompose by farm, solving each farm independently:

- Subproblem size:  $6 \times 3 = 18$  variables per farm
- Sampler: DWaveCliqueSampler (native clique embedding)
- Reads: 100 per subproblem
- Iterations: 3 (boundary coordination)

### 2.3.3 Spatial-Temporal Decomposition (QPU)

Decompose by spatial clusters and time periods:

- Farms per cluster: 2-3 (based on problem size)
- Subproblem size:  $\leq 18$  variables
- Sampler: DWaveCliqueSampler
- Reads: 100 per subproblem
- Iterations: 3

## 2.4 Result Data Files

- Data/statistical\_comparison\_20251214\_192625.json
- Plots: Plots/plot\_\*.png

## 3 Formulation B: 27 Foods → 6 Families (Aggregated)

Used by: hierarchical\_statistical\_test.py, significant\_scenarios\_benchmark.py

### 3.1 Overview

Property	Value
Foods (original)	$C_{\text{orig}} = 27$ foods
Families (aggregated)	$C_{\text{agg}} = 6$ families
Periods	$T = 3$
Farm sizes tested	$F \in \{25, 50, 100\}$
Original variables	$N_{\text{orig}} = F \times 27 \times 3 = \{2025, 4050, 8100\}$
Aggregated variables	$N_{\text{agg}} = F \times 6 \times 3 = \{450, 900, 1800\}$

### 3.2 Food-to-Family Aggregation

Family	Constituent Foods (3-5 per family)
Legumes	Beans, Lentils, Chickpeas, Peas, Soybeans
Grains	Rice, Wheat, Maize, Barley, Oats
Vegetables	Tomatoes, Cabbage, Peppers, Spinach, Broccoli
Roots	Potatoes, Carrots, Cassava, Sweet Potatoes, Beets
Fruits	Bananas, Oranges, Mangoes, Apples, Grapes
Other	Nuts, Herbs, Spices, Seeds

### 3.3 Three-Level Hierarchical Approach

Level	Description
<b>Level 1</b>	<b>Aggregation + Spatial Decomposition:</b> - Aggregate 27 foods to 6 families (reduces variables by 4.5×) - Partition farms into spatial clusters (~5 farms per cluster) - Variables per cluster: $5 \times 6 \times 3 = 90$
<b>Level 2</b>	<b>QPU Solving with Boundary Coordination:</b> - Solve each cluster on D-Wave QPU (90 vars ≤ clique limit) - 3 iterations with boundary information exchange - Total QPU calls: $(F/5) \times 3$
<b>Level 3</b>	<b>Post-Processing: Family → Food Refinement:</b> - Refine family assignments to specific foods - Local optimization within each family (3-5 foods) - Diversity analysis and constraint verification

### 3.4 Solution Methods

#### 3.4.1 Ground Truth: Gurobi

- Operates on **aggregated** problem (6 families, not 27 foods)
- Timeout: 300 seconds
- MIP Gap: 10% tolerance
- Same objective function as QPU (fair comparison)

#### 3.4.2 Hierarchical QPU

- Farms per cluster: 5
- Cluster variables:  $5 \times 6 \times 3 = 90$  (within clique limit)
- Reads: 100 per cluster
- Iterations: 3 (boundary coordination)
- Total QPU access time: Tracked separately from wall time

### 3.5 Result Data Files

- `Data/hierarchical_results_20251212_124349.json`
- `Data/hierarchical_*_farms.json` (individual size results)
- `Data/significant/benchmark_results_20251214_*.json`

## 4 Comparison of Formulations

### 4.1 Key Differences

Aspect	Formulation A (Native)	Formulation B (Aggregated)
Crop representation	6 families directly	27 foods → 6 families
Problem scale	Small-medium (5-25 farms)	Medium-large (25-100 farms)
Variables per farm	18	18 (after aggregation)
Total variables	90-450	450-1800 (aggregated)
QPU decomposition	Per-farm or cluster	Spatial clusters
Post-processing	Optional	Required (family→food)

### 4.2 What Is Identical

Both formulations share:

1. **Objective function structure:** All 5 components with same coefficients
2. **Rotation matrix generation:** Same seed (42), same frustration ratio (0.7)
3. **Spatial neighbor graph:**  $k = 4$  nearest neighbors

4. **Parameter values:**  $\gamma_{\text{rot}} = 0.2$ ,  $\beta_{\text{div}} = 0.15$ ,  $\lambda = 3.0$
5. **Constraints:** Soft one-hot ( $\leq 2$  crops per farm-period)
6. **Gurobi configuration:** Same timeout, gap tolerance, focus settings

### 4.3 When Results Are Comparable

Results are **directly comparable** when:

- Using the **same formulation** (A or B)
- With the **same objective function** (verified in code)
- At **overlapping sizes** (e.g., 25 farms appears in both)

Results require **careful interpretation** when:

- Comparing across formulations (A vs B)
- Comparing original vs aggregated variable counts
- Evaluating post-processing impact on objective values

## 5 Benchmark Result Summary

### 5.1 Formulation A Results (Native 6 Families)

Farms	Vars	Gurobi Obj	Gurobi Time	QPU Obj	Gap	Speedup
5	90	4.078	300s	3.452	15.3%	15.1×
10	180	7.175	300s	6.157	14.2%	8.7×
15	270	11.526	300s	9.890	14.2%	6.0×
20	360	14.889	300s	13.209	11.3%	5.2×

**Note:** All Gurobi runs hit timeout (300s). Gap calculated against timeout solution, not proven optimal.

### 5.2 Formulation B Results (Aggregated 27→6)

Farms	Orig	Agg	Gurobi	Time	QPU	Gap	Speed
25	2025	450	12.32	300s	11.89	3.5%	7.5×
50	4050	900	24.71	300s	23.94	3.1%	5.0×
100	8100	1800	49.35	300s	46.82	5.1%	3.0×

**Note:** Gurobi operates on aggregated problem (6 families) for fair comparison.

### 5.3 Significant Scenarios Results

Scenario	Farms	Vars	Method	Gap	Speedup
rotation_micro_25	5	90	clique	14.4%	5.5×
rotation_small_50	10	180	clique	19.2%	3.3×
rotation_medium_100	20	360	clique	0.15%	2.0×
rotation_large_25farms	25	2025	hier.	—	—
rotation_xlarge_50farms	50	4050	hier.	—	—
rotation_xxlarge_100farms	100	8100	hier.	—	—

**Note:** Hierarchical QPU had errors in this run; see individual JSON files for details.

## 6 Quantum Advantage Metrics

### 6.1 Speedup Calculation

$$\text{Speedup} = \frac{T_{\text{Gurobi}}}{T_{\text{QPU}}} \quad (10)$$

Where  $T$  represents wall-clock time including all preprocessing and post-processing.

### 6.2 Optimality Gap

$$\text{Gap} = \frac{|\mathcal{O}_{\text{Gurobi}} - \mathcal{O}_{\text{QPU}}|}{|\mathcal{O}_{\text{Gurobi}}|} \times 100\% \quad (11)$$

**Interpretation:**

- Positive gap: QPU objective lower than Gurobi (QPU “worse”)
- Zero gap: Equal solutions
- Note: Gurobi may not have found optimal due to timeout

### 6.3 Where Quantum Advantage Occurs

Based on benchmark results:

1. **Time Advantage:** QPU is faster for all tested sizes (5-100 farms)
  - Small problems (5-10 farms): 5-15× speedup
  - Medium problems (15-25 farms): 3-7× speedup
  - Large problems (50-100 farms): 2-5× speedup
2. **Solution Quality:** Within 15% of Gurobi timeout solution
  - Gurobi hits timeout without proving optimality
  - Gap is vs. best-found solution, not proven optimal
  - At larger sizes, gap decreases (both struggle similarly)
3. **Scaling Behavior:** QPU time scales sub-linearly
  - Gurobi: 300s timeout at all sizes
  - QPU: 20s (5 farms) to 100s (100 farms)
  - Advantage increases with problem size

## 7 Reproducibility

### 7.1 Random Seeds

- Rotation matrix: seed 42 (NumPy)
- Farm positioning: deterministic grid layout
- D-Wave sampling: stochastic (100 reads for variance)

## 7.2 Running Benchmarks

```
# Formulation A (5-25 farms, 6 families)
python Scripts/statistical_comparison_test.py

# Formulation B (25-100 farms, 27->6 families)
python Scripts/hierarchical_statistical_test.py

# Unified benchmark (all sizes)
python Scripts/significant_scenarios_benchmark.py
```

## 7.3 D-Wave Configuration

- Sampler: DWaveCliqueSampler
- Reads: 100 (default)
- Annealing time: 20  $\mu$ s (default)
- Chain strength: Auto (for problems > clique size)

## 8 Conclusion

This document has specified the complete mathematical formulations used across all quantum advantage benchmarks:

1. **Common Framework:** All benchmarks share the same 5-component MIQP objective function, differing only in crop representation and problem scale.
2. **Formulation A:** Direct 6-family representation for small-medium problems (90-450 variables), solved with Clique Decomposition or Spatial-Temporal QPU methods.
3. **Formulation B:** Aggregated 27 $\rightarrow$ 6 representation for medium-large problems (450-1800 variables), solved with Hierarchical QPU decomposition.
4. **Comparability:** Results within the same formulation are directly comparable. Cross-formulation comparisons require acknowledging the aggregation and post-processing differences.

For comprehensive plots comparing quantum vs. classical performance, see the companion visualization document and the `Plots/` directory.