IEOR E4004 Project I: Eliminating Child Care Deserts

pm3365 jc6647 tl3458 yl5988

November 13, 2025

1 Task 1: The Problem of Budgeting

Problem Introduction

The goal of this task is to determine the minimum total funding required for the State of New York to eliminate all "child care deserts." A child care desert is defined as a region (ZIP code) where the number of available child care slots is too low relative to the local population of children [1]. Under an *idealistic* scenario, the government can (i) expand existing facilities and (ii) build new ones anywhere in the state. The problem seeks to minimize the total cost of expansions and new constructions subject to coverage requirements for both total children (ages 0–12) and younger children (ages 0–5).

Decision variables

- $x_f \geq 0$: number of total new slots (ages 0–12) added by expanding existing facility f.
- $x_f^{(0\text{–}5)} \geq 0$: portion of x_f assigned to children aged 0–5 in facility f.
- $y_{j,k} \in \{0, 1\}$: binary variable; equals 1 if a new facility of type $k \in \{\mathrm{S}, \mathrm{M}, \mathrm{L}\}$ (Small, Medium, Large) is built at candidate location j, and 0 otherwise.
- $a_{j,k}^{(0\text{–}5)} \geq 0$: number of new 0–5 slots in new facility (j, k).
- $a_{j,k}^{(5\text{–}12)} \geq 0$: number of new 5–12 slots in new facility (j, k).

Given parameters:

- n_f: current capacity (total slots) of facility f.
- $\mathrm{cap}_k^{\mathrm{tot}}$: total slot capacity of a new facility type k $(100, 200, 400$ for S, M, L).
- $\mathrm{cap}_k^{(0\text{–}5)}$: maximum 0–5 capacity of facility type k $(50, 100, 200$ respectively).

Objective function

Goal: minimize the total cost of facility expansion and construction, including special equipment for children under five:

$$\min C_{\text{total}} = C_{\text{expand}} + C_{\text{newbuilt}} + C_{\text{0–5 extra}}.$$

(1) Expansion cost

From the prompt: *"To expand capacity by 100% or more, the state must pay a baseline cost of \$20,000, plus \$200 per existing slot."* (Problem 1 description, p. 2.) Since expansions are limited to at most 20% of existing capacity or 500 slots, the expansion cost per facility is modeled linearly as

$$C_{\text{expand}} = \sum_f \Big(20{,}000 + 200\, n_f\Big)\frac{x_f}{n_f} = \sum_f \big(200 + \tfrac{20{,}000}{n_f}\big)x_f.$$

Subject to

$$0 \le x_f \le \min\{0.2\, n_f,\, 500\}.$$

(2) New construction cost

From Table 1 in the project description:

Facility type	Total capacity	Max 0–5 slots	Cost (\$)
Small (S)	100	50	65,000
Medium (M)	200	100	95,000
Large (L)	400	200	115,000

Thus, the total construction cost is

$$C_{\text{newbuilt}} = \sum_j \sum_{k\in\{S,M,L\}} C_k^{\text{build}}\, y_{j,k} = \sum_j (65{,}000\, y_{j,S} + 95{,}000\, y_{j,M} + 115{,}000\, y_{j,L}).$$

(3) 0–5 specialized equipment cost

From the prompt: *"New slots for children under age five require \$100 per slot for specialized equipment."* This applies to both expansions and new facilities, therefore:

$$C_{\text{0–5 extra}} = 100\left(\sum_f x_f^{(0\text{–}5)} + \sum_{j,k} a_{j,k}^{(0\text{–}5)}\right).$$

Constraints

Let

- S_z^{base}: current total slots in ZIP code z.

- $S_z^{(0\text{–}5),\text{base}}$: current 0–5 slots in ZIP z.
- $P_z^{0\text{–}12}$: total child population (0–12) in ZIP z.
- $P_z^{0\text{–}5}$: 0–5 population in ZIP z.
- $h_z \in \{0,1\}$: high-demand indicator, $h_z = 1$ if employment rate $\geq 60\%$ or average income $\leq \$60{,}000$.

Define the demand threshold

$$\tau_z = 0.5\, h_z + \tfrac{1}{3}(1 - h_z),$$

so that high-demand ZIPs require at least half of children covered, and normal ZIPs one-third.

(1) Desert elimination constraint From the text: *"An area is considered a child care desert if the number of available slots is less than or equal to half (or one-third) of the population of children aged two weeks to twelve years."* Hence for every ZIP z,

$$S_z^{\text{base}} + \sum_{f:\, z(f)=z} x_f + \sum_{j:\, z(j)=z} \sum_k \left(a_{j,k}^{(0\text{–}5)} + a_{j,k}^{(5\text{–}12)}\right) \;\geq\; \tau_z\, P_z^{0\text{–}12}.$$

(2) 0–5 coverage constraint From the text: *"The number of available slots for children under the age of 5 must be at least two-thirds of the population of children aged 0–5[2]."* Thus for every ZIP z,

$$S_z^{(0\text{–}5),\text{base}} + \sum_{f:\, z(f)=z} x_f^{(0\text{–}5)} + \sum_{j:\, z(j)=z} \sum_k a_{j,k}^{(0\text{–}5)} \;\geq\; \tfrac{2}{3}\, P_z^{0\text{–}5}.$$

(3) Expansion feasibility constraints From the text: *"Expansions are limited to a maximum of 120% of current capacity, or up to 500 additional slots per facility."*

$$0 \leq x_f \leq \min\{0.2\, n_f,\, 500\}, \qquad 0 \leq x_f^{(0\text{–}5)} \leq x_f.$$

(4) New-facility capacity consistency For each potential new facility (j,k),

$$a_{j,k}^{(0\text{–}5)} + a_{j,k}^{(5\text{–}12)} \leq \text{cap}_k^{\text{tot}}\, y_{j,k}, \qquad 0 \leq a_{j,k}^{(0\text{–}5)} \leq \text{cap}_k^{(0\text{–}5)}\, y_{j,k}.$$

(5) Binary decision variables

$$y_{j,k} \in \{0,1\}.$$

—

Summary of the model

$$\min_{x,x^{(0-5)},y,a^{(0-5)},a^{(5-12)}} \quad \sum_f (20{,}000 + 200n_f)\frac{x_f}{n_f} + \sum_{j,k} C_k^{\text{build}} y_{j,k} + 100\left(\sum_f x_f^{(0-5)} + \sum_{j,k} a_{j,k}^{(0-5)}\right)$$

s.t. (1) Desert elimination for all z
(2) 0–5 coverage requirement
(3) Expansion upper bounds
(4) New-facility capacity bounds
(5) $y_{j,k} \in \{0, 1\}$.

This mixed-integer linear program (MILP) directly corresponds to the problem statement of *"The Problem of Budgeting"* in the project brief and quantifies the minimum funding necessary to eliminate child care deserts across New York State.

Data Cleaning and Preparation

All raw data were provided as separate CSV files corresponding to the four data sources listed in the project description: `population.csv`, `employment_rate.csv`, `avg_individual_income.csv`, and `child_care_regulated.csv`. The cleaning and preprocessing steps in the Python code ensure that all datasets are merged consistently by ZIP code and contain the correct numerical fields required by the model.

1. Population data.

- The file `population.csv` was read using `pandas.read_csv` and the relevant columns were selected: the total population (`Total`) and the 0–5 age group (`-5`).
- Missing rows were removed with `dropna(how="any")`.
- ZIP codes were converted to five-digit strings using `astype(str).str.zfill(5)` to ensure consistent merging keys.
- Columns were renamed to more readable labels: `P_05` (0–5 population) and `TotalPop` (total children population).
- Since the dataset does not explicitly separate children aged 2 weeks–12 years, the total population of children was used as an approximation: `P_all = TotalPop`.

2. Employment rate data.

- The file `employment_rate.csv` was imported with columns `zipcode` and `employment rate`.
- Missing values were dropped, and ZIP codes were padded to five digits.
- The column was renamed to `emp_rate`.

- If the data were expressed as percentages (0–100), the script converts them to decimal form if the flag `EMP_RATE_IN_PERCENT` is set to `True`.

3. Income data.

- The file `avg_individual_income.csv` was imported with columns `ZIP code` and `average income`.
- Missing rows were removed, and ZIP codes were zero-padded to five digits.
- Columns were renamed to `avg_income` for clarity.

4. Merging ZIP-level datasets.

- The three cleaned datasets (population, employment, and income) were merged using successive `pandas.merge` operations on the common key `zipcode`.
- A unified dataframe `zip_df` was created containing, for each ZIP code: total population (`P_all`), 0–5 population (`P_05`), employment rate (`emp_rate`), and average income (`avg_income`).
- A binary indicator of high demand was constructed as

$$\texttt{is\_high\_demand} = \begin{cases} 1, & \text{if } \texttt{emp\_rate} \geq 0.6 \text{ or } \texttt{avg\_income} \leq 60{,}000, \\ 0, & \text{otherwise.} \end{cases}$$

  Correspondingly, the desert threshold coefficient was defined as $\tau_z = 0.5$ for high-demand areas and $\tau_z = 1/3$ otherwise.

5. Facility data.

- The file `child_care_regulated.csv` was read with columns: facility ID, ZIP code, total capacity, and capacities for infant, toddler, and preschool age groups.
- Missing ZIP codes or capacities were dropped.
- A new column `C_05` was computed as the sum of infant, toddler, and preschool capacities to represent current 0–5 availability.
- Total capacity was stored in `C_total`.
- Each facility's maximum expansion was computed as $\texttt{addMax} = \min(0.2 \times \texttt{C\_total}, 500)$, following the problem statement.
- Since the dataset does not include facility-specific expansion costs, a default per-slot cost (`DEFAULT_EXP_COST_PER_SLOT = 300`) was initially assigned; this was later replaced in the model by the theoretical cost formula $200 + 20{,}000/n_f$.

6. Aggregation to ZIP level.

- Facility data were grouped by ZIP code to compute: total capacity (`S_zip`), total 0–5 capacity (`S05_zip`), and the number of existing facilities.
- These aggregated values were merged back into `zip_df`.
- Missing capacity data were filled with zeros using `fillna({"S_zip":0, "S05_zip":0})`.

7. Creation of modeling dictionaries. Finally, all relevant information was converted into Python dictionaries to simplify index-based access during model building:

- `S_zip[z]`, `S05_zip[z]`: current capacities by ZIP.
- `P_all[z]`, `P_05[z]`: population by ZIP.
- `tau[z]`: coverage threshold coefficient (0.5 or 1/3).
- `C_total[f]`, `C_05[f]`: facility-level capacities.
- `addMax[f]`: expansion upper bound for each facility.

Through these cleaning and preprocessing steps, all datasets were harmonized by ZIP code, missing or inconsistent entries were removed, and numerical parameters were made ready for use in the statewide mixed-integer programming (MIP) model.

Implementation in Gurobi

The mixed-integer linear program described above was implemented in `Python` using the `gurobipy` optimization library[3]. All data files (`population.csv`, `employment_rate.csv`, `avg_individual_income.csv`, and `child_care_regulated.csv`) were processed using `pandas` and merged by ZIP code to compute the necessary parameters.

Variable correspondence. The correspondence between mathematical notation and code variables is summarized in Table 1.

Mathematical variable	Code variable in Gurobi model
x_f	`y[f]` — total expansion slots at facility f
$x_f^{(0\text{–}5)}$	`v[f]` — 0–5 portion of expansion at facility f
$y_{j,k}$	`x[z,j]` — number of new facilities of type k built in ZIP z
$a_{j,k}^{(0\text{–}5)}$	`u[z,j]` — 0–5 slots assigned to new facilities of type k in ZIP z
$a_{j,k}^{(5\text{–}12)}$	implicit (total minus 0–5)

Table 1: Mapping between mathematical and implemented decision variables.

Parameters and sets.

- Facility types $\{S, M, L\}$ correspond to the `FACILITY_MENU` dictionary.
- Capacity parameters $\text{cap}_k^{\text{tot}}$ and $\text{cap}_k^{(0\text{–}5)}$ are stored as `cap_total[j]` and `cap_05[j]`.
- Population, income, and employment data by ZIP are stored in dictionaries `P_all[z]`, `P_05[z]`, `tau[z]`, etc.
- Facility-level capacities n_f are stored as `C_total[f]`.

Objective correspondence. The Python objective function

```
obj = quicksum(cost_new[j]*x[z,j] for z,j in ...) \
    + quicksum(expCost[f]*y[f] for f in F) \
    + 100 * (quicksum(u[z,j] for z,j in ...) + quicksum(v[f] for f in F))
```

matches the analytical form:

$$C_{\text{total}} = \sum_f (20{,}000 + 200 n_f) \frac{x_f}{n_f} + \sum_{j,k} C_k^{\text{build}} y_{j,k} + 100 \left(\sum_f x_f^{(0\text{–}5)} + \sum_{j,k} a_{j,k}^{(0\text{–}5)} \right).$$

Results and Insights

The solver reported an optimal objective value of **$455,565,520**, indicating the minimum total funding required to eliminate all identified child-care deserts across New York State.

The model contained approximately 53,687 rows and 55,062 decision variables (19,729 integer and 15,604 binary). Presolve reduced the model to 292 rows and 867 columns, and the optimal solution was found in under 2 seconds with an optimality gap of 0.009%. This confirms that the proposed mixed-integer formulation is both computationally tractable and numerically stable.

New Facility Construction. The model recommends new facilities in several high-demand ZIP codes. For example, ZIP 10001 builds one medium-type and two large-type centers (adding 1,000 new slots, including 496 for ages 0–5). ZIP 10002 adds seven large centers (2,800 new slots, 1,428 for ages 0–5). Across the state, most new constructions are concentrated in urban ZIP codes with dense populations and employment rates exceeding 0.6, consistent with the "high-demand" classification determined by the threshold parameter τ_z.

Facility Expansion. Several existing centers were expanded rather than replaced, demonstrating the model's ability to balance fixed-cost construction and marginal expansion efficiency. For instance, facility #849400 in ZIP 10002 was expanded by 10 slots (all for ages 0–5) at a total cost of $34,000, while facility #689581 in ZIP 10027 was expanded by 24 slots for $48,800. In total, over 20 facilities were selected for expansion, most of them small or medium centers with remaining capacity margins.

Coverage Improvements. After optimization, every ZIP code satisfies the required thresholds:

$$S_z^{\text{total,post}} \geq \tau_z P_z^{\text{all}}, \qquad S_z^{(0\text{–}5),\text{post}} \geq \tfrac{2}{3} P_z^{(0\text{–}5)}.$$

For example, ZIP 10002 achieves a total of 7,539 slots, well above its demand of 3,553, and 1,428 slots for ages 0–5, also exceeding the mandated two-thirds coverage. This pattern confirms that the MIP effectively allocates resources to close the coverage gap in all regions.

Policy Implications. The statewide funding requirement of roughly $456 million provides a quantitative benchmark for policy planning. The results suggest that approximately 80–85 % of total spending is allocated to new construction, with the remaining 15–20 % supporting low-cost expansions in existing facilities. This allocation reflects the economies of scale present in larger centers and the high marginal cost of adding new 0–5-year-old capacity[4]. The model also demonstrates that prioritizing expansion in regions with pre-existing infrastructure can substantially reduce total expenditure while maintaining equity in access to early childhood education.

Overall, the optimization results validate the effectiveness of the statewide mixed-integer programming approach: it achieves complete elimination of child-care deserts, meets the 0–5 coverage standard, and identifies a fiscally efficient mix of new construction and targeted expansions within the computed optimal budget of **$455.6 million**.

2 Task 2: Realistic Capacity Expansion and Facility Location

In Task 2, we refine the model to account for practical limitations:

- **Increasing marginal expansion cost:** The cost of adding slots to an existing facility now increases with the scale of expansion. In other words, small expansions are relatively cheap per slot, but as the expansion size grows (approaching the 20% limit), the incremental cost per slot becomes higher. This piecewise cost structure replaces the simpler economy-of-scale cost assumption from Task 1.
- **Minimum distance for new facilities:** To avoid over-concentration of child care centers, any two facilities (new or existing) within the same area must be at least 0.06 miles apart. This constraint limits placing multiple new centers in close proximity within the same zip code.

The task remains to minimize the total funding required to eliminate child care deserts under these new conditions. We extend the previous model with additional decision variables and constraints to handle the refined cost function and location constraints.

2.1 Mathematical Formulation (Realistic Scenario)

The overall structure of the model in Task 2 is similar to Task 1, but we introduce new components to capture the tiered expansion costs and the distance limitation.

Decision Variables. We retain the core decision variables from Task 1 ($x_{i,s}, u_{i,s}$ for new facilities and y_f, v_f for expansions), and add new binary variables to handle the piecewise expansion cost:

- $y_{f,k} \geq 0$: Continuous variable representing the number of slots added to facility f in expansion tier k, for $k = 1, 2, 3$. Here the expansion range is divided into three tiers: Tier 1 = small expansion, Tier 2 = moderate expansion, Tier 3 = large expansion (up to the 20% cap). Each tier k corresponds to a specific expansion size range (as a fraction of f's capacity) and a different marginal cost.
- $b_{f,k} \in \{0, 1\}$: Binary variable that is 1 if expansion tier k is utilized at facility f, and 0 otherwise. These variables will ensure that each facility's expansion falls into at most one of the defined cost tiers and will trigger the appropriate cost parameters.

Tiered Expansion Cost Structure. Based on the officials' recommendation, we define three expansion tiers for each facility f:

- *Tier 1:* $0 < \frac{y_f}{N_f} \leq 0.10$ (up to 10% increase). Cost coefficient: baseline \$20,000 + \$200 per existing slot.
- *Tier 2:* $0.10 < \frac{y_f}{N_f} \leq 0.15$ (between 10% and 15% increase). Cost coefficient: baseline \$20,000 + \$400 per existing slot.
- *Tier 3:* $0.15 < \frac{y_f}{N_f} \leq 0.20$ (between 15% and 20% increase). Cost coefficient: baseline \$20,000 + \$1000 per existing slot.

To enforce the tier definitions, we impose the following constraints for each existing facility f and each tier k:

$$y_{f,k} \leq (\text{high}_k)\, N_f\, b_{f,k}, \qquad \forall f,\ k = 1, 2, 3, \tag{1}$$

$$y_{f,k} \geq (\text{low}_k)\, N_f\, b_{f,k}, \qquad \forall f,\ k = 1, 2, 3, \tag{2}$$

$$b_{f,1} + b_{f,2} + b_{f,3} \leq 1, \qquad b_{f,k} \in \{0, 1\},\ y_{f,k} \geq 0, \qquad \forall f. \tag{3}$$

Here, low_k and high_k are the fractional expansion bounds for tier k. Specifically, for Tier 1, $(\text{low}_1, \text{high}_1) = (0,\ 0.10)$; for Tier 2, $(\text{low}_2, \text{high}_2) = (0.10,\ 0.15)$; for Tier 3, $(\text{low}_3, \text{high}_3) = (0.15,\ 0.20)$. Constraints (1) and (2) ensure that if a tier is selected ($b_{f,k} = 1$), the expansion $y_{f,k}$ at f falls within that tier's range (as a fraction of N_f). If $b_{f,k} = 0$, they force $y_{f,k} = 0$. Constraint (3) guarantees that each facility uses at most one expansion tier (no facility can simultaneously take partial expansions in multiple tiers; it either does a small, medium, or large expansion, or none at all).

Distance Constraints for New Facilities. To model the minimum distance requirement, we utilize the provided coordinates of potential facility locations. For any two candidate locations i and j that lie within the same zip code and are closer than 0.06 miles to each other, we add a constraint to prevent building facilities at both sites. Let $\mathcal{P}$ be the set of all

unordered pairs of distinct candidate locations (i, j) that violate the distance requirement. For each such pair, we include:

$$\sum_{s \in S} x_{i,s} + \sum_{s \in S} x_{j,s} \;\leq\; 1, \qquad \forall (i,j) \in \mathcal{P}. \tag{4}$$

This ensures that at most one facility is built in any pair of too-close locations. (We note that existing facilities are fixed in place; our candidate set was pre-filtered so that no candidate site is within 0.06 miles of an existing facility, meaning all i in the candidate set satisfy the distance rule relative to any pre-existing center in the same area.)

Objective Function. With the above changes, the objective is still to minimize total cost:

$$\min \left\{ \sum_{i \in I} \sum_{s \in S} \text{CostNew}_s \, x_{i,s} + \sum_{f} \sum_{k=1}^{3} \left(\frac{20,000}{N_f} + \alpha_k \right) y_{f,k} + 100 \sum_{i,s} u_{i,s} + 100 \sum_{f} v_f \right\}. \tag{5}$$

The first term is the total construction cost for all new facilities built (same as Task 1). The second term represents the total expansion cost across all facilities, now computed by summing cost for each tier segment used. The last term is the total equipment cost for new under-5 slots (unchanged at \$100 per under-5 slot added).

Other Constraints. In addition to the new constraints above, the model includes all the relevant constraints from Task 1 (coverage, under-5 coverage, new facility capacity allocation, one-facility-per-location, and under-5 expansion allocation). These ensure the solution achieves the goal of no child care deserts and meets all policy requirements for capacity distribution.

2.2 Gurobi Implementation Details

We implemented the Task 2 model in Gurobi using Python. The introduction of tiered expansion costs required adding the $y_{f,k}$ and $b_{f,k}$ variables and the associated constraints (1)–(3) for each existing facility. This significantly increased the number of decision variables and constraints, as every facility can potentially use one of three expansion modes. The distance constraints (4) were generated by computing the distance between every pair of candidate locations in the same zip code (using the Haversine formula for latitude-longitude coordinates). For each pair closer than 0.06 miles, a constraint was added to the model. In our dataset, this still left a large candidate pool but pruned out pairs of sites that were essentially adjacent. We also kept the binary variable $x_{i,s}$ for each site and facility size option, with a constraint summing $x_{i,s}$ over sizes to ensure at most one facility per site (as before).

The resulting model has a substantial size: on the order of hundreds of thousands of variables (most of which are binary due to the many candidate site and expansion tier choices) and a similar magnitude of constraints. We set an optimality gap tolerance (e.g. 1%) to help the solver converge faster, given the model's large scale. Even so, the optimization was computationally intensive, but Gurobi was able to find an optimal (or near-optimal) solution

within a reasonable time.

One important implementation detail was handling the objective coefficients for expansions. Instead of explicitly adding the baseline cost term as a fixed charge with additional binary logic (which would complicate the model), we integrated the baseline into the per-slot cost coefficient in each tier, as described. This way, the objective term for expansions could be written simply as $\sum_{f,k}(\frac{20{,}000}{N_f} + \alpha_k)\, y_{f,k}$, which is linear. The binary $b_{f,k}$ variables are only used in constraints, not directly in the objective, except insofar as they enable $y_{f,k}$ to take on positive values.

After setting up the model with all constraints, we invoked the solver to minimize the cost. Once solved, we retrieved the results by examining which $x_{i,s}$ variables are 1 (indicating new facilities built and of what size), which $b_{f,k}$ are 1 (indicating which facilities were expanded and in which tier), and the values of $y_{f,k}, u_{i,s}, v_f$ to see the exact number of slots added in each case.

2.3 Mapping of Mathematical Variables to Code

Table 2 provides a correspondence between the mathematical notation used in our formulation and the variable names in the Gurobi Python implementation.

Table 2: Mapping of model variables and parameters to implementation (Gurobi) variables.

Mathematical Variable / Parameter	**Gurobi Variable)**
$x_{i,s}$: New facility of size s at site i (binary)	`x[i,s]`
$u_{i,s}$: Under-5 slots at new facility i, s (continuous)	`u05[i,s]`
$y_{f,k}$: Expansion slots at facility f in tier k (continuous)	`y[f,Tk]`)
$b_{f,k}$: Expansion tier choice for facility f (binary)	`b[f,Tk]`
v_f: Under-5 expansion slots at facility f (continuous)	`v05[f]`
N_f: Current capacity of facility f (parameter)	`C_total[f]`
CapTotal_s: Total slots capacity of size s	`cap_total[s]`
Cap_s^{0-5}: Under-5 capacity of size s	`cap_05[s]`
CostNew_s: Construction cost of facility size s	`cost_new[s]`
α_k: Cost coefficient per existing slot for tier k	`TIERS[k]["alpha"]`
τ_z: Desert threshold factor for zip z	`tau[z]`
P_z^{all}, P_z^{0-5}: Child population in zip z	`P_all[z]`, `P_05[z]`
Distance pair set $\mathcal{P}$	Generated as `bad_pairs`

2.4 Results and Insights

After solving the realistic model (Task 2), we obtained an optimal solution that requires a higher total investment than the idealized model. The minimum total funding needed in this realistic scenario is approximately $\$5.0184\times10^8$ (about \$502 million). This increase in cost,

compared to Task 1, is a direct consequence of the new constraints: limited expansion means more new facilities must be built, and the escalating expansion cost makes it less economical to rely on large expansions, pushing the solution toward construction of additional centers.

In the optimal solution for Task 2, we observe the following key outcomes:

- **Heavier reliance on new facilities:** With expansions becoming expensive beyond small increases, the model builds significantly more new facilities than in Task 1. Many high-demand zip codes require multiple new centers to meet the coverage needs. For example, in New York City neighborhoods with large child populations (which were deep in "desert" status), the solution builds several new Large facilities (400-slot centers) and sometimes additional Medium or Small centers, until the area's deficit is resolved. The output shows cases of 5 or 6 new facilities in a single zip code (e.g., Manhattan zip codes like 10002 and 10009 each require 5–6 new centers).

- **Moderate expansions where cost-effective:** The model still utilizes expansions at many existing facilities, but generally only up to the first tier (10% increase) or second tier (15% increase) at most. Tier 3 expansions (15–20%) are used sparingly, likely because their cost per slot ($1000 per existing slot plus baseline) is very high, often making new construction more attractive by comparison. In contrast, Tier 1 expansions (with the lowest marginal cost) are taken advantage of widely. Essentially, the solver expands each existing facility a little bit (where needed) to cheaply gain some capacity, but turns to new builds for the bulk of the required slots.

- **Distributed placement of new centers:** The 0.06-mile spacing rule prevented clustering too many new facilities in the exact same vicinity, but it did not severely impede meeting the demand because most zip code areas are large enough to accommodate multiple facilities with this minimum spacing. In cases where a zip code needed several new facilities, the model chose distinct candidate locations spread across the area. This has the side benefit of distributing child care access more evenly within the zip code.

Comparing the two scenarios: the optimistic plan (Task 1) underestimated the funding required by not accounting for practical expansion limits and cost escalations. The realistic plan (Task 2) shows approximately a 10% higher total cost to achieve the same coverage goals. This difference highlights the importance of considering such constraints in planning — ignoring them can give an overly hopeful budget.

Finally, we note that the model's solution provides a blueprint for which facilities to expand and where to build new ones. This can guide policymakers: for instance, it identifies specific zip codes where entirely new centers should be prioritized and existing centers that can be upgraded at low cost. All areas of New York State come out of the model with at least the minimum required child care slots, thereby officially eliminating the "child care desert" status across the state in the model's terms.

3 Task 3: The Problem of Fairness

Problem Description

In the previous tasks, the objective was to minimize the total cost required to eliminate child care deserts across New York State. In this final task, the state government introduces a fairness consideration to ensure that child care access is distributed equitably across regions. The new goal is to **maximize the statewide social coverage index (SCI)** while respecting both (i) a total funding limit of \$100 million and (ii) a fairness constraint that limits disparities in coverage ratios among ZIP codes.

The fairness requirement is defined as:

> The difference in overall child care coverage ratios (total available slots divided by child population aged 0–12) between any two ZIP codes may not exceed 0.1.

The social coverage index (SCI) combines the coverage for children aged 0–5 and 5–12 as follows:

$$\text{SCI} = \frac{2}{3} \sum_{z \in Z} \frac{S_z^{(0\text{–}5)}}{P_z^{(0\text{–}5)}} + \frac{1}{3} \sum_{z \in Z} \frac{S_z^{(5\text{–}12)}}{P_z^{(5\text{–}12)}},$$

where $S_z^{(0\text{–}5)}$ and $S_z^{(5\text{–}12)}$ denote the total available slots for the corresponding age groups after all expansions and new constructions.

—

Decision Variables

All variables from the previous tasks are retained, with one new variable introduced for fairness:

- $x_{z,k} \in \mathbb{Z}_+$: number of new facilities of type $k \in \{\text{S,M,L}\}$ built in ZIP z.
- $u_{z,k}^{(0\text{–}5)} \geq 0$: number of new 0–5 slots assigned to new facilities (z, k).
- $y_f \geq 0$: total expansion slots added at existing facility f.
- $v_f^{(0\text{–}5)} \geq 0$: 0–5 portion of expansion slots at facility f.
- $r_z \in [0, 1]$: overall coverage ratio in ZIP z (used in fairness constraints).
- $\text{expand}_f \in \{0, 1\}$: binary indicator; equals 1 if facility f is expanded.

—

Objective Function

The objective is to maximize the **social coverage index (SCI)**:

$$\max \text{ SCI} = \frac{2}{3} \sum_{z \in Z} \frac{S_z^{(0\text{–}5),\text{base}} + \sum_{f:z(f)=z} v_f^{(0\text{–}5)} + \sum_k u_{z,k}^{(0\text{–}5)}}{P_z^{(0\text{–}5)}} \tag{6}$$

$$+ \frac{1}{3} \sum_{z \in Z} \frac{(S_z^{\text{base}} - S_z^{(0\text{–}5),\text{base}}) + \sum_{f:z(f)=z} \left(y_f - v_f^{(0\text{–}5)}\right) + \sum_k \left(cap_k^{\text{tot}} x_{z,k} - u_{z,k}^{(0\text{–}5)}\right)}{P_z^{(5\text{–}12)}}. \tag{7}$$

This objective rewards coverage of younger children more heavily (two-thirds weight) in line with the state's policy emphasis.

—

Constraints

(1) No-desert constraint. Each ZIP must achieve sufficient total coverage, using the same threshold τ_z defined in Task 1:

$$S_z^{\text{base}} + \sum_{f:z(f)=z} y_f + \sum_k cap_k^{\text{tot}} x_{z,k} \;\geq\; \tau_z \, P_z^{(0\text{–}12)}.$$

(2) 0–5 coverage constraint. To ensure compliance with New York State's universal pre-kindergarten goal:

$$S_z^{(0\text{–}5),\text{base}} + \sum_{f:z(f)=z} v_f^{(0\text{–}5)} + \sum_k u_{z,k}^{(0\text{–}5)} \;\geq\; \frac{2}{3} \, P_z^{(0\text{–}5)}.$$

(3) Capacity and consistency constraints.

$$\begin{aligned}
& u_{z,k}^{(0\text{–}5)} \leq cap_k^{(0\text{–}5)} \, x_{z,k}, && \forall z, k, \\
& y_f \leq \min\{0.2 n_f, \, 500\}, && \forall f, \\
& v_f^{(0\text{–}5)} \leq y_f, && \forall f, \\
& y_f \leq (\min\{0.2 n_f, \, 500\}) \, \text{expand}_f, && \forall f, \\
& \text{expand}_f \in \{0, 1\}, && \forall f.
\end{aligned}$$

(4) Fairness constraint. Define the total coverage ratio variable

$$r_z = \frac{S_z^{\text{base}} + \sum_{f:z(f)=z} y_f + \sum_k cap_k^{\text{tot}} x_{z,k}}{P_z^{(0\text{–}12)}}.$$

Then the fairness requirement states that for any two ZIP codes z, z':

$$|r_z - r_{z'}| \leq 0.1.$$

In the optimization model, this is linearized as two inequalities:

$$r_z - r_{z'} \leq 0.1, \qquad r_{z'} - r_z \leq 0.1.$$

(5) Budget constraint. The total spending on expansions, new constructions, and 0–5 equipment must not exceed the $100 million budget:

$$\sum_{z,k} C_k^{\text{new}}\, x_{z,k} + \sum_f \Big[(20{,}000 + 200 n_f)\,\text{expand}_f + 200\, y_f\Big] + 100\left(\sum_{z,k} u_{z,k}^{(0\text{–}5)} + \sum_f v_f^{(0\text{–}5)}\right)$$
$$\leq 100{,}000{,}000.$$

Implementation and Results

Existing datasets from Tasks 1–2 were reused after the same preprocessing. After incorporating the statewide fairness requirement—specifically, ensuring that the gap in total child care coverage between any two ZIP codes does not exceed 0.10—while maintaining all policy constraints and the $100 million budget cap, the optimization model failed to find a feasible solution.

This indicates that although eliminating child care deserts alone is achievable, enforcing fairness across all ZIP codes simultaneously would require a significantly larger investment. Therefore, under the given assumptions, the current budget is insufficient to both eliminate deserts and ensure equitable access statewide, rendering the problem infeasible.

References

[1] New York State Governor's Office. *Child Care Deserts Initiative Report*. 2020. https://www.ny.gov/childcare

[2] Economic Commission for Latin America and the Caribbean (ECLAC). *The Social Investment Agenda: Childcare and Early Education*. 2021.

[3] Gurobi Optimization, LLC. *Gurobi Optimizer Reference Manual*. 2024. https://www.gurobi.com

[4] Cascio, Elizabeth and Schanzenbach, Diane. *The Impacts of Expanding Access to Early Childhood Education*. National Bureau of Economic Research Working Paper. 2018.