Constraint Optimization of MicroPlate Designs

Ramiz Gindullin[1], **María Andreína Francisco Rodríguez**[1]

[1]Uppsala University, Sweden
Ramiz.Gindullin@it.uu.se, Maria.Andreina.Francisco@it.uu.se

Abstract

Standardized microplates are used to conduct large-scale biomedical research. The design of microplate layouts plays an essential role in handling so-called *plate effects*, i.e., systematic variations across the geometry of a microplate. An *effective layout* allows us to detect and negate plate effects. The randomized placement of controls and compounds produces layouts of limited effectiveness, so specific approaches are needed. A previously developed system, PLAID, proposed a constraint satisfaction model to construct effective plate layouts. However, PLAID does not scale well with microplate dimensions. To improve on PLAID, we propose Constraint Optimization of MicroPlate Designs (COMPD), which allows for greater flexibility and higher quality of the layouts.

1 Introduction

Large-scale biomedical research often requires testing a very large number of drugs[1] and bio-materials (Blay et al. 2020). Microplates ('microwell plates' or 'plates') store the research materials in individual wells in a compartmentalized manner, enabling many parallel experiments alongside biomaterials in control groups (Wikimedia Commons 2003). Microplates are standardized, with wells arranged in a 2:3 rectangle with common sizes being 6, 12, 24, 48, 96, 384 or 1536 sample wells on a single plate. Liquid-handling systems ensure precise and fast transfer of drug compounds and samples to individual wells in microplates (Tegally et al. 2020; Torres-Acosta, Lye, and Dikicioglu 2022).

The main challenge of microplate usage is the presence of *plate (or positional) effects* (Zhang 2011), i.e. systematic variations across the geometry of a microplate (*within-plate effects*) and across different plates (*between-plate effects*). Common within-plate effects include (*i*) linear row effects, (*ii*) linear column effects, (*iii*) linear row and column effects and (*iv*) bowl-shaped spatial effects (Franchina and Sergiani 2020; Zhang 2011) (see Figure 1). Imperfections in the liquid-handling equipment and contamination of pipetting instruments usually lead to effects (*i*)–(*ii*). Environmental factors, e.g. variations in temperature, humidity, luminosity, etc., often lead to the plate effects (*iii*)–(*iv*).

[1]In this article, the terms 'drug compound', 'drug', and 'compound' are used interchangeably.

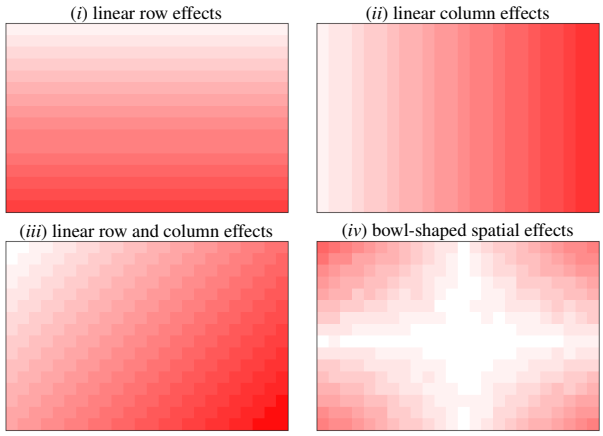


Figure 1: Visualized examples of plate effects (*i*)–(*iv*). The intensity of the color indicates the intensity of a plate effect.

Between-plate effects are often a result of their location in the lab, or a variance in the maintenance of microplates, e.g., improper cleaning and disinfection.

The quality of the acquired data is essential in large-scale biomedical research (Franchina and Sergiani 2020; Lu and Young 2020; Williams et al. 2023). Strong plate effects affect the experimental data quality and can render it unusable. One way to reliably detect and counteract plate effects is through the plate layout design (Birmingham et al. 2009; Zhang 2008), i.e., the allocation of individual drug compounds and control samples between wells of a plate. In some cases, there is a need to design layouts for several microplates simultaneously. There are three main approaches:

- Manual design of so-called border layouts, usually by placing controls on borders of microplates and distributing the drug samples in an easy-to-perform way for human hands (Birmingham et al. 2009; Brower, Fensterer, and Bush 2008; Connelly 2014; Williams et al. 2017). Border layouts have limited effectiveness (Mpindi et al. 2015; Zhang 2008) and human pipetting is infeasible for microplates with 384 wells or more.
- Randomized plate layouts (Caicedo et al. 2017; Roselle, Verch, and Shank-Retzlaff 2016). Pure randomization can produce large areas without control samples or place

a drug's replicates on adjacent wells, (*i*) making them both susceptible to the same plate effects, and (*ii*) they can also affect each other (Birmingham et al. 2009).

- Effective layouts (Francisco Rodríguez, Puigvert, and Spjuth 2023), i.e., layouts that distribute both controls and samples evenly across the plate in order to reduce unwanted bias and correct plate effects. The desired properties of an effective layout are: (*i*) an even distribution of controls, (*ii*) an even distribution of samples, (*iii*) an even distribution of empty wells, if there are any, (*iv*) avoiding edges and attempting to fill the middle of the plate first, (*v*) balancing controls across all plates, plate halves, and plate quadrants, and (*vi*) balancing controls whenever possible per row and column across all plates.

There are several plate editors available on the market: Brunn (Alvarsson et al. 2011), FlowJo (FlowJo 2017), Labfolder (LabForward 2025), PlateDesigner (Suprun and Suárez-Fariñas 2019), and PlateEditor (Delorme et al. 2021). Some of them can generate randomized layouts and none – effective layouts. Paper (Schiff et al. 2022) proposes to generate a number of randomized layouts, evaluate each, and select the best one. This procedure does not guarantee the selection of an effective plate layout.

PLAID (Francisco Rodríguez, Puigvert, and Spjuth 2023) (Plate Layouts using Artificial Intelligence Design) is a suite of tools designed to generate effective layouts by solving a *constraint satisfaction problem* (CSP) (Brailsford, Potts, and Smith 1999). PLAID was shown to generate layouts that produce robust results WRT to plate effects. Nonetheless, PLAID has a few limitations: (*i*) pre-calculated threshold amounts on when to spread the controls across the plate, (*ii*) the replicates are not placed as far as possible from each other, (*iii*) is unable to generate layouts for plate lines[2] with an odd number of rows or columns, (*iv*) only works with Gecode[3], and (*v*) its performance and memory usage scale poorly due to the extensive presence of reified constraints.

To overcome these limitations, we propose *Constraint Optimization of MicroPlate Designs* (COMPD), which solves a constraint optimization problem to generate effective layouts for microplates instead of solving a CSP. We argue that this reformulation adds flexibility to PLAID, which improves the quality of microplate layouts.

This paper starts with problem description in Section 2. Section 3 provides general principles behind COMPD. The COMPD model is presented in Section 4. In Section 5, we evaluate COMPD and compare it with PLAID. Finally, in Section 6 we present our concluding remarks.

2 Problem Description

In this section, we first present possible plate layouts and then discuss how materials are allocated between plates.

[2]A plate line is a rectangular subset of plate wells, the layout of which is copied 2 or more times on the same microplate. Plate lines are used to test the same drugs on different cell cultures.

[3]http://www.gecode.org (Accessed: 2025-06-01)

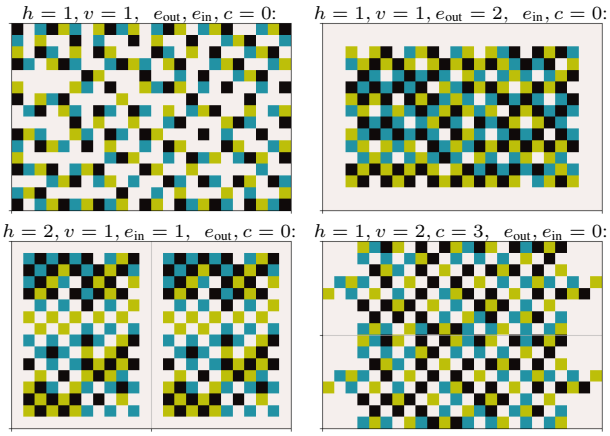


Figure 2: Various configurations of microplate layouts, where each color represents a different control or drug

2.1 Plate Dimensions

We are given P microplates. Each microplate has n rows and m columns. In most cases, the microplate layouts do not cover the whole plate (see Figure 2):

- by having an outer edge of empty wells of size e_{out},
- by splitting a microplate into h horizontal and v vertical 'lines' separated by inner edges of size e_{in},
- by forcing the corner wells of a line to be empty, in a square shape of size c.

Forcing outer and corner wells to be empty is useful for the research due to (*i*) these wells being influenced by environmental factors more than inner wells, and, in some cases, (*ii*) the equipment limitations, e.g., some microscopes might be unable to take good images in the corner wells.

When the microplate is split into plate lines, each line has identical layouts, which are then applied to test different cells or cultures identically, which is a common practice in biomedical research. Thus, we can simplify the model by taking only the dimensions of a line, n_l and m_l:

$$n_l = \frac{n - 2 \cdot e_{\text{out}}}{v} - 2 \cdot e_{\text{in}}, \quad m_l = \frac{m - 2 \cdot e_{\text{out}}}{h} - 2 \cdot e_{\text{in}} \quad (1)$$

The number of available wells on any given line, W_l, is equal to $W_l = n_l \times m_l - 4c^2$.

2.2 Distribution of Materials Between Microplates

Any layout is created to distribute certain amounts of various materials, i.e., controls and compounds in different concentrations, between each line. These amounts are adjusted to the number of lines, i.e,. divided by $h \cdot v$ without a remainder, thus we concentrate on only one line per microplate. If the total amount of materials is strictly less than $P \cdot W_l$ then we have empty wells to layout. While it is not necessary, we would like to ensure that empty wells do not create any large 'holes' in the microplate. Thus, we place them effectively.

For each material $j \in [1, M]$, we are provided with:

- $T_j \in \{$'empty', 'control', 'compound'$\}$, the type of the material ($(T_j =$ 'empty'$) \Leftrightarrow (j = 1)$),
- $Name_j$, the name of the material ('empty' when $j = 1$, names of controls and drugs to test, otherwise),
- $Concentration_j$, the concentration of the material j (0 for empty wells),
- $Q_j \geq 0$, the total quantity of material j, i.e. replicates. $\sum_{j=1}^{M} Q_j = P \cdot W_l$.

If $P = 1$, then the content of the single line is self-evident. Otherwise, we need to distribute materials between P microplates following one of 3 policies selected by the user:

1. random distribution of materials,
2. distributing materials evenly between plate lines,
3. placing all replicates of a drug on the same plate. Empty wells and controls are distributed proportionally to the remaining space of a plate line.

There are two approaches to make: (*i*) implement each strategy as a set of restrictions in a constraint model, or (*ii*) pre-calculate quantities $q_{i,j} \geq 0$, i.e., total quantity of material j on plate line i, such that:

$$\sum_{j=1}^{M} q_{i,j} = W_l, i \in [\overline{1, P}], \quad \sum_{i=1}^{P} q_{i,j} = Q_j, j \in [\overline{1, M}]. \quad (2)$$

The latter approach simplifies the model and focuses the search space. It also avoids using reified constraints during the calculation of the optimization criteria (see Sections 4.2 and 4.5). Thus, we chose to pre-calculate $q_{i,j}$ as input data.

3 Designing Effective Layouts

Before we present the details of our constraint model, we first discuss some general principles that guided its creation.

A uniform distribution of a material on a plate is the main factor behind an effective plate layout. One way of achieving uniform distribution while filling a plate line is to try to push replicates as far away as possible from each other. We can express this as maximization of a minimal distance between any two replicates, d_{min}[4]. In Figure 3, black wells have already fixed places on the grid. If we place a new material on the grid, representing a plate line, and choose a blue well, with $d_{\text{min}} = 1$. We can increase d_{min} further by, e.g., choosing the green well. By doing so, we increase the uniformity of well density and achieve $d_{\text{min}} = 2$.

The criterion can be strengthened further. Imagine we have 40 replicates on an 8×8 grid. The maximal value of d_{min} is 1, meaning that a solver can fill the first five rows of a grid with replicates even though it would not be an effective distribution. To ameliorate this, once the maximal d_{min} is achieved, we then maximize the sum of the distances from each replicate to its closest neighbor. This two-step approach performs well on small grids with only one compound, but, in preliminary experiments, it scaled poorly with the grid dimensions and the number of materials. Thus, we only try to maximize d_{min} for each individual drug and control because:

[4]To utilize the full potential of constraint programming, we used Manhattan distances.

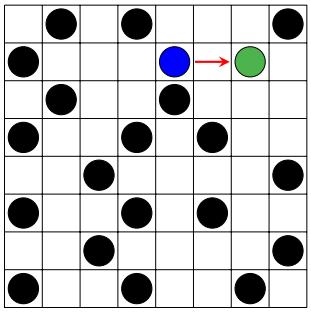


Figure 3: Placing an additional replicate less effectively (blue) and more effectively (green)

- in most situations, this will be sufficient to achieve a uniform distribution and
- spreading all materials on the grid would force the replicates to be placed more uniformly.

The criterion works well when the number of replicates is high enough, like in Figure 3. When the number of replicates is low, then this criterion would force all of them to be in the corners (see light-grey wells in Figure 4). In most cases, this behavior is not desirable. To combat this, we can create fake wells outside of the grid and use them during the calculation of d_{min}, which would force sparse replicates to be placed closer to the middle of the grid (see black wells in Figure 4). We apply fake wells only when the number of replicates of a drug is less than 4, and we do not use them for controls.

When replicates are few, we must also keep in mind that even with fake edge wells, this criterion alone can lead to designs that a human would deem less effective or less desirable. e.g., if we move a black well to the right side of the grid (see green arrow in Figure 4), both placements have the same $d_{\text{min}} = 3$, but the former is preferable. Thus, before optimizing WRT d_{min}, we first must assign replicates to the respective quadrants, giving priority to diagonal quadrants, i.e., quadrants I (red) and III (green).

The quadrant assignment also limits the search space for well coordinates, i.e., the solver has only to consider the dimensions of the quadrant and not the whole grid.

4 Constraint Optimization Model

In this section, we describe the COMPD model: coordinate variables, criteria sets, optional and symmetry-breaking constraints, and the optimization criterion.

4.1 Placing the Materials on Plate Lines

In Section 2.2 we pre-calculated all quantities $q_{i,j}$, $i \in [\overline{1, P}]$, $j \in [\overline{1, M}]$, i.e. the amount of material j assigned on plate line i with W_l wells, such that $\sum_{j=1}^{M} q_{i,j} = W_l$. We assign a material for each well $k \in [\overline{1, W_l}]$ on plate line i. For this, we use matrix I, where $I_{i,k} \in [\overline{1, M}]$, $i \in [\overline{1, P}$, $k \in [\overline{1, W_l}]$ means that we assign material $I_{i,k}$ on the well i on plate k, such that $\sum_{k=1}^{W_l} (I_{i,k} = j) = q_{i,j}$. We chose the sequential assignment due to the ease of implementation.

Next, we used constants $A_{i,k}, i \in \overline{[1,P]}, k \in \overline{[1,W_l]}$ to assign materials to each quadrant to (*i*) make sure that materials are spread more or less evenly across all quadrants, and (*ii*) fill quadrants in the following order: quadrant I, quadrant IV, quadrant II, and quadrant III (see Section 3). To avoid reified constraints, we precalculated the distribution of wells between quadrants in a deterministic manner; thus, on the same problem, it will not differ between the experiments.

The quadrant assignments must be balanced:

$$\begin{array}{l}\forall i \in \overline{[1,P]} : \sum_{k=1}^{W_l}(A_{i,k}=1) = \lfloor n_l/2 \rfloor \cdot \lceil m_l/2 \rceil,\\ \forall i \in \overline{[1,P]} : \sum_{k=1}^{W_l}(A_{i,k}=2) = \lceil n_l/2 \rceil \cdot \lceil m_l/2 \rceil,\\ \forall i \in \overline{[1,P]} : \sum_{k=1}^{W_l}(A_{i,k}=3) = \lfloor n_l/2 \rfloor \cdot \lfloor m_l/2 \rfloor,\\ \forall i \in \overline{[1,P]} : \sum_{k=1}^{W_l}(A_{i,k}=4) = \lceil n_l/2 \rceil \cdot \lfloor m_l/2 \rfloor.\end{array} \quad (3)$$

For each well k on plate i, we created a pair of integer variables, row $C_{i,k,1}$ and column $C_{i,k,2}$, to denote its position coordinates. We set the domains of coordinate variables:

$$\begin{array}{l}\forall i \in \overline{[1,P]}, \forall k \in \overline{[1,W_l]}, A_{i,k}=1:\\ \quad C_{i,k,1} \in \overline{[1, \lfloor n_l/2 \rfloor]},\ C_{i,k,2} \in \overline{[1, \lceil m_l/2 \rceil]},\\ \forall i \in \overline{[1,P]}, \forall k \in \overline{[1,W_l]}, A_{i,k}=2:\\ \quad C_{i,k,1} \in \overline{[\lfloor n_l/2 \rfloor + 1, n_l]},\ C_{i,k,2} \in \overline{[1, \lceil m_l/2 \rceil]},\\ \forall i \in \overline{[1,P]}, \forall k \in \overline{[1,W_l]}, A_{i,k}=3:\\ \quad C_{i,k,1} \in \overline{[1, \lfloor n_l/2 \rfloor]},\ C_{i,k,2} \in \overline{[\lceil m_l/2 \rceil + 1, m_l]},\\ \forall i \in \overline{[1,P]}, \forall k \in \overline{[1,W_l]}, A_{i,k}=4:\\ \quad C_{i,k,1} \in \overline{[\lfloor n_l/2 \rfloor + 1, n_l]},\ C_{i,k,2} \in \overline{[\lceil m_l/2 \rceil + 1, m_l]}.\end{array} \quad (4)$$

If corner wells are forced to be empty ($c > 0$), we imposed the following reified constraints:

$$\begin{array}{l}\forall i \in \overline{[1,P]}, \forall k \in \overline{[1,W_l]},\ A_{i,k}=1:\\ \quad (C_{i,k,1} \le c) \Rightarrow (C_{i,k,2} \ge c+1),\\ \forall i \in \overline{[1,P]}, \forall k \in \overline{[1,W_l]},\ A_{i,k}=2:\\ \quad (C_{i,k,1} \ge n_l - c + 1) \Rightarrow (C_{i,k,2} \ge c+1),\\ \forall i \in \overline{[1,P]}, \forall k \in \overline{[1,W_l]},\ A_{i,k}=3:\\ \quad (C_{i,k,1} \le c) \Rightarrow (C_{i,k,2} \le m_l - c),\\ \forall i \in \overline{[1,P]}, \forall k \in \overline{[1,W_l]},\ A_{i,k}=4:\\ \quad (C_{i,k,1} \ge n_l - c + 1) \Rightarrow (C_{i,k,2} \le m_l - c).\end{array} \quad (5)$$

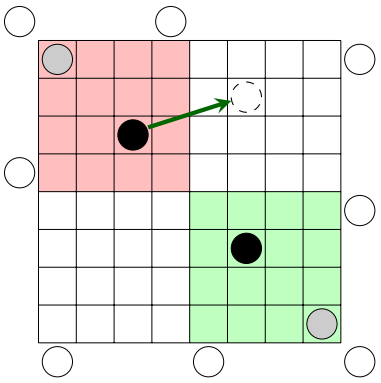


Figure 4: Various placements of two replicates on a plate effectively with and without fake edge wells

All wells of a plate must be placed on different coordinates, i.e., do not overlap. To enforce this, we used LEXALLDIFFERENT global constraint (Beldiceanu, Carlsson, and Rampon 2012, p. 1268):

$$\forall i \in \overline{[1,P]}: \text{LEXALLDIFFERENT}\left([\langle C_{i,k,1}, C_{i,k,2}\rangle]_{k=1}^{W_l}\right) \quad (6)$$

4.2 Criteria Sets

As discussed in Section 3, the goal of constructing a microplate layout is to maximize distances between materials of the same kind. More precisely, we want to spread out wells of the same drug without differentiating between concentrations. We also want to do the same with controls taken as a whole, without differentiating between various types of controls. Similarly, we also try to spread out empty wells.

If there are more control wells (or replicates of a single drug) than $\lfloor W_l/2 \rfloor$, then it is more effective to spread each type of control (or each concentration of a drug, resp.) individually. Otherwise, the minimal distance between them will be 1, and solvers would likely place the wells sequentially.

Thus, for each plate i, based on the number of wells W_l, the pre-calculated quantities $q_{i,j}, i \in \overline{[1,P]}, j \in \overline{[1,M]}$, and on the assumptions described above, we construct a list of sets $\mathcal{S}_{i,s}, s \in \overline{[1, s_i^{\max}]}$, called *criteria sets*. The criteria set is a set of materials, the replicates of which try spreading out across the plate. The criteria sets of a plate line i, $i \in \overline{[1,P]}$, must not intersect and must cover all materials:

$$\forall s_1, s_2 \in \overline{[1, s_i^{\max}]}, s_1 \ne s_2 :\ \mathcal{S}_{i,s_1} \cap \mathcal{S}_{i,s_2} = \varnothing.$$

$$\bigcup_{s=1}^{s_i^{\max}} \mathcal{S}_{i,s} = \{j \mid \forall j \in \overline{[1,M]}\},\ \forall i \in \overline{[1,P]}.$$

We assign $\mathcal{S}_{i,1} = \{1\}$ because we always treat empty wells separately (see Section 2.2).

Example 1 *Assume that we have $M = 6$ materials:*

$$\begin{array}{l}T = \langle \text{'empty'}, \text{'control'}, \text{'control'}, \text{'compound'},\\ \qquad\qquad \text{'compound'}, \text{'compound'} \rangle,\\ Name = \langle \text{'empty'}, \text{'pos'}, \text{'neg'}, \text{'drug 1'}, \text{'drug 1'}, \text{'drug 1'} \rangle,\\ Concentration = \langle 0, 100, 100, 10, 100, 1000 \rangle.\end{array}$$

Also assume that we pre-calculated the following quantities $q_{i,j}$ for $P = 3$ plate lines, $W_l = 20$ wells each (we are not using any of the distribution policies described in Section 2.2 for simplicity):

$$q = \left\langle \begin{array}{cccccc} 3 & 4 & 4 & 3 & 3 & 3 \\ 2 & 6 & 6 & 2 & 2 & 2 \\ 4 & 2 & 2 & 4 & 4 & 4 \end{array} \right\rangle$$

For plate line 1, i.e. $i = 1$, the total number of wells assigned to controls is $q_{1,2} + q_{1,3} = 8$, which is less than half the number of wells on plate line 1, namely $W_l/2 = 10$, meaning that both 'pos' and 'neg' controls are in the same criteria set $\mathcal{S}_{1,2}$. The total number of wells assigned to the three concentrations of drug 1 is $q_{1,4} + q_{1,5} + q_{1,6} = 9$, which is again less than half the number of wells on plate

line 1*, so all concentrations of drug 1 are in the same criteria set* $\mathcal{S}_{1,3}$*. i.e. for plate line* 1 *we have* $s_1^{max} = 3$ *criteria sets:* $\langle\{1\},\{2,3\},\{4,5,6\}\rangle$.

For plate line 2*, i.e.* $i = 2$*, the total number of wells assigned to controls is* $q_{2,2} + q_{2,3} = 12$*, which is larger than half the number of wells on plate line* 2*, namely* $W_l/2 = 10$*, so in this case the 'pos' and 'neg' controls are in two different criteria sets* $\mathcal{S}_{2,2}$ *and* $\mathcal{S}_{2,3}$*. The total number of wells assigned to the three concentrations of drug* 1 *is* $q_{2,4} + q_{2,5} + q_{2,6} = 6 < 10$*, meaning that all concentrations of drug* 1 *are in the same criteria set* $\mathcal{S}_{2,4}$*. i.e. for plate line* 2 *we have* $s_2^{max} = 4$ *criteria sets:* $\langle\{1\},\{2\},\{3\},\{4,5,6\}\rangle$.

Finally, for plate line 3*, i.e.* $i = 3$*, the total number of wells assigned to controls is* $q_{3,2} + q_{3,3} = 4 < 10$*, so both controls are in the same criteria set* $\mathcal{S}_{3,2}$*. The total number of wells assigned to drug* 1 *is* $q_{3,4} + q_{3,5} + q_{3,6} = 12 > 10$*, meaning that the three concentrations of drug* 1 *are in three different criteria sets* $\mathcal{S}_{3,3}$*,* $\mathcal{S}_{3,4}$ *and* $\mathcal{S}_{3,5}$*. i.e. for plate line* 3 *we have* $s_3^{max} = 5$ *criteria sets:* $\langle\{1\},\{2,3\},\{4\},\{5\},\{6\}\rangle$.

For each criteria set $\mathcal{S}_{i,s}$ we assign two Boolean flags, $F^f_{i,s}, F^c_{i,s}$. Flag $F^f_{i,s}$ denotes whether or not we use fake edge wells in addition to the wells of the criteria set $\mathcal{S}_{i,s}$. Flag $F^c_{i,s}$ denotes whether or not we apply the optimization criteria for the given set $\mathcal{S}_{i,s}$ to account for drugs with a large enough number of replicates, meaning the plate effects become insignificant. Exclusion of such drugs from the criterion calculation (see Section 4.5) reduces the computational burden, while the placement of other materials would spread the overrepresented compound automatically.

Setting thresholds for flags $F^f_{i,s}$ and $F^c_{i,s}$ determines the performance of the constraint model and the quality of solutions it produces. Based on intuition and some preliminary observations, the thresholds used in COMPD are:

- $F^f_{i,s} \;\Leftrightarrow\; \left(\sum_{j\in\mathcal{S}_{i,s}} q_{i,j} \le 4 \wedge T_j \neq \text{'control'}\right)$;
- $W_l < 300$:
  $F^c_{i,s} \;\Leftrightarrow\; \sum_{j\in\mathcal{S}_{i,s}} q_{i,j} \le \min(90, 0.45\cdot W_l)$;
- $W_l \ge 300$:
  $F^c_{i,s} \;\Leftrightarrow\; \sum_{j\in\mathcal{S}_{i,s}} q_{i,j} \le \min(120, 0.35\cdot W_l)$.

4.3 Optional Constraints

We included the ability for the user to enforce additional restrictions on the plate layouts. They are not strictly necessary for the creation of effective plate layouts, but they can help avoid linear row or column plate effects.

Users can select whether or not they want to enforce the placement of replicates of a concentration of a drug on different rows, columns, or both whenever possible, i.e., when the plate line has enough rows or columns for the assigned number of wells on its quadrants.

If the number of replicates of a material is less than the number of rows, columns, or both, then we can further strengthen the optional constraints by placing the replicates on different rows, columns, or both across all plates.

If a user enforces the placement of replicates of materials on different rows, then for each material j, T_j = 'compound', and each plate $i \in \overline{[1,P]}$ we create flag $F^{rows}_{i,j}$:

$$F^{rows}_{i,j} \equiv \begin{pmatrix} |\langle k\rangle_{\forall k\in\overline{[1,W_l]}: I_{i,k}=j \wedge A_{i,k}\in\{1,3\}}| \le \lfloor n_l/2\rfloor \wedge \\ |\langle k\rangle_{\forall k\in\overline{[1,W_l]}: I_{i,k}=j \wedge A_{i,k}\in\{2,4\}}| \le \lceil n_l/2\rceil \end{pmatrix} \quad (7)$$

We also create flag F^{rp}_j, which denotes whether we can enforce placement of replicates of material j, T_j = 'compound', on different rows across all plates:

$$F^{rp}_j \equiv \begin{pmatrix} |\langle k\rangle_{\forall i\in\overline{[1,P]},\forall k\in\overline{[1,W_l]}: I_{i,k}=j \wedge A_{i,k}\in\{1,3\}}| \le \lfloor n_l/2\rfloor \wedge \\ |\langle k\rangle_{\forall i\in\overline{[1,P]},\forall k\in\overline{[1,W_l]}: I_{i,k}=j \wedge A_{i,k}\in\{2,4\}}| \le \lceil n_l/2\rceil \end{pmatrix} \quad (8)$$

If a user enforces the placement of replicates of materials on different columns, then for each material j, T_j = 'compound', and each plate $i \in \overline{[1,P]}$, we create flag $F^{cols}_{i,j}$:

$$F^{cols}_{i,j} \equiv \begin{pmatrix} |\langle k\rangle_{\forall k\in\overline{[1,W_l]}: I_{i,k}=j \wedge A_{i,k}\in\{1,2\}}| \le \lceil m_l/2\rceil \wedge \\ |\langle k\rangle_{\forall k\in\overline{[1,W_l]}: I_{i,k}=j \wedge A_{i,k}\in\{3,4\}}| \le \lfloor m_l/2\rfloor \end{pmatrix} \quad (9)$$

We also create flag F^{cp}_j, which denotes whether we can enforce placement of replicates of material j, T_j = 'compound', on different columns across all plates:

$$F^{cp}_j \equiv \begin{pmatrix} |\langle k\rangle_{\forall i\in\overline{[1,P]},\forall k\in\overline{[1,W_l]}: I_{i,k}=j \wedge A_{i,k}\in\{1,2\}}| \le \lceil m_l/2\rceil \wedge \\ |\langle k\rangle_{\forall i\in\overline{[1,P]},\forall k\in\overline{[1,W_l]}: I_{i,k}=j \wedge A_{i,k}\in\{3,4\}}| \le \lfloor m_l/2\rfloor \end{pmatrix} \quad (10)$$

When $T_j \neq$ 'compound', flags $F^{cols}_{i,j}$, $F^{cols}_{i,j}$, $i \in [1,P]$, F^{rp}_j and F^{cp}_j are always set to `false`. If the user does not wish to enable either of these options, then corresponding flags are set to `false` for all $i \in \overline{[1,P]}, j \in \overline{[1,M]}$.

Thus, we stated the following optional constraints:

$$\begin{aligned}
&\forall j \in \overline{[1,M]},\ F^{rp}_j = \texttt{true}: \\
&\textsc{AllDifferent}\Big(\langle C_{i,k,1}\rangle_{\forall i\in\overline{[1,P]},\forall k\in\overline{[1,W_l]}: I_{i,k}=j}\Big), \\
&\forall j \in \overline{[1,M]},\ F^{cp}_j = \texttt{true}: \\
&\textsc{AllDifferent}\Big(\langle C_{i,k,2}\rangle_{\forall i\in\overline{[1,P]},\forall k\in\overline{[1,W_l]}: I_{i,k}=j}\Big), \\
&\forall i \in \overline{[1,P]},\ \forall j \in \overline{[1,M]},\ F^{rows}_{i,j} = \texttt{true}: \\
&\qquad \textsc{AllDifferent}\Big(\langle C_{i,k,1}\rangle_{\forall k\in\overline{[1,W_l]}: I_{i,k}=j}\Big), \\
&\forall i \in \overline{[1,P]},\ \forall j \in \overline{[1,M]},\ F^{cols}_{i,j} = \texttt{true}: \\
&\qquad \textsc{AllDifferent}\Big(\langle C_{i,k,2}\rangle_{\forall k\in\overline{[1,W_l]}: I_{i,k}=j}\Big).
\end{aligned} \quad (11)$$

4.4 Symmetry-Breaking Constraints

To avoid searching through equivalent solutions, we enforced lexicographic order by LexChain (Beldiceanu, Carlsson, and Rampon 2012, p. 1276) on coordinate variables of the same material on the same quadrant. We grouped materials on plate lines by criteria sets, unless $F^{rows}_{i,j} \vee F^{cols}_{i,j} = \texttt{true}$, to ensure the strictness of the search:

$$\begin{aligned}
&\forall i \in \overline{[1,P]},\ \forall s \in \overline{[1, s_i^{\max}]},\ \forall a \in \overline{[1,4]}: \\
&\textsc{LexChain}\Big(\big\langle\langle C_{i,k,1}, C_{i,k,2}\rangle\big\rangle_{\substack{\forall k\in\overline{[1,W_l]}: I_{i,k}\in\mathcal{S}_{i,s} \\ \wedge I_{i,k}\in\mathcal{S}_{i,s}\wedge A_{i,k}=a \\ \wedge\neg F^{rows}_{i,I_{i,k}}\wedge\neg F^{cols}_{i,I_{i,k}}}}\Big)
\end{aligned} \quad (12)$$

If corresponding optional constraints are applied, we enforce LEXCHAIN only on the wells of material j of plate i:

$$\forall i \in [\overline{1,P}], \forall j \in [\overline{1,M}], \forall a \in [\overline{1,4}], F^{rows}_{i,j} \vee F^{cols}_{i,j} : \text{LEXCHAIN}\Big(\big\langle \langle C_{i,k,1}, C_{i,k,2} \rangle \big\rangle_{\forall k \in [\overline{1,W_l}]: I_{i,k}=j \wedge A_{i,k}=a}\Big) \quad (13)$$

4.5 Optimization Criterion

First, we assigned $f = 8$ fake edge wells (see Section 3):

$$\mathcal{F} = \Big\langle \langle 0,0 \rangle, \langle n_l+1, 1 \rangle, \langle 1, m_l+1 \rangle, \langle n_l+1, m_l+1 \rangle, \langle 0, \lfloor m_l/2 \rfloor \rangle, \langle \lfloor n_l/2 \rfloor+1, m_l \rangle, \langle n_l+1, \lfloor m_l/2 \rfloor+1 \rangle, \langle \lfloor n_l/2 \rfloor, 0 \rangle \Big\rangle \quad (14)$$

Next, minimal distances of each criteria set $\mathcal{S}_{i,s}$ with at least 2 wells and $F^c_{i,s} = \texttt{true}$ are collected in list $\mathcal{D}$:

$$\mathcal{D} = \Big\langle \min\Big(\big\langle |C_{i,k_1,1} - C_{i,k_2,1}| + |C_{i,k_1,2} - C_{i,k_2,2}| \big\rangle_{\substack{\forall k_1,k_2 \in [\overline{1,W_l}]: \\ k_1 > k_2 \wedge \\ I_{i,k_1}, I_{i,k_2} \in \mathcal{S}_{i,s}}} + \big\langle |C_{i,k_1,1} - \mathcal{F}_{k_2,1}| + |C_{i,k_1,2} - \mathcal{F}_{k_2,2}| \big\rangle_{\substack{\forall k_1 \in [\overline{1,W_l}], \\ \forall k_2 \in [\overline{1,f}]: \\ F^f_{i,s} \wedge I_{i,k_1} \in \mathcal{S}_{i,s}}} \Big) \Big\rangle_{\forall i \in [\overline{1,P}], \forall s \in [\overline{1,s^{\max}_i}]: F^c_{i,s} \wedge \sum_{j \in [\overline{1,M}]: j \in \mathcal{S}_{i,s}} q_{i,j} \geq 2} \quad (15)$$

Finally, we wanted to maximize $\langle \min(\mathcal{D}), \sum \mathcal{D} \rangle$ lexicographically. Since MiniZinc[5] does not support this functionality, we used a substitute criterion:

$$\text{maximize } \Big(\mathcal{M} \cdot \min(\mathcal{D}) + \sum \mathcal{D} \Big), \quad (16)$$

where $\mathcal{M}$ is a sufficiently large natural number. We tested a large range of values, and stopped with $\mathcal{M} = 1000$, as it ensures effective search without affecting the performance.

5 Evaluation

To evaluate COMPD layouts, we replicated dose-response and screening experiments from (Francisco Rodríguez, Puigvert, and Spjuth 2023, Sections 2, 3, and 7). *Dose-response experiments* attempt to evaluate the effect of a substance in a specific assay at increasing concentrations (Vandenberg 2022). *Screening experiments* attempt to identify hits from a large number of samples for further analysis (Mpindi et al. 2015; Zhang 2008). We refer to (Francisco Rodríguez, Puigvert, and Spjuth 2023) to provide comprehensive descriptions and explanations of the experiments.

[5]https://www.minizinc.org/ (Accessed: 2025-06-01)

Experimental setting We wrote COMPD in MiniZinc. COMPD can be used with any CP, SAT or LCG solver. For this evaluation, we used CP-SAT 9.12.4544.

COMPD layouts were generated on a 2024 MacBook Air with an 8-core Apple M3 and 16 GB. When generating layouts for dose-response experiments with COMPD, we used 8 threads and the 180-seconds timeout. All layouts for screening experiments were generated within 25 seconds with a proof of optimality using 8 threads.

We took PLAID experiments[6] and updated them[7] to (*i*) ensure compatibility with the latest versions of Python and its packages, and (*ii*) exclude border layouts and include COMPD layouts in the results. We adjusted the results for COMPD layouts using LOESS regression, the same as for PLAID and random layouts. We then compiled the results[8].

Discussion Dose-response experiments demonstrated that COMPD produces layouts of the same quality as PLAID across all metrics regardless of concentrations, replicates, or plate effects (see Figure 5 and detailed results).

During screening experiments, COMPD layouts demonstrated a consistent improvement compared to PLAID across all quality metrics (see Table 1), i.e., COMPD's placement of controls (see Figure 6) results in statistically significant higher precision with a smaller variance. See Figures 2 and 18–26 in the supplementary materials for detailed results.

We did not conduct a comprehensive performance comparison between PLAID and COMPD due to the difficulty of comparisons between an optimization model, which can be interrupted at any moment, and a satisfaction model, which terminates after finding a solution. Additionally, both approaches produce layouts of varying quality.

PLAID can take a significant amount of time during the generation of layouts for microplates with 384 wells without a guarantee of a solution due to a restrictive search strategy; this strategy is randomized, thus sometimes it leads to a dead-end. e.g. during an example[9] PLAID spent 8468 seconds, utilizing 8 cores, before terminating without a solution. COMPD generated a layout with proof of optimality within five minutes using CP-SAT. Additionally, during this example, PLAID utilized up to 75 Gb of RAM, while COMPD only took 3 Gb. In some examples, PLAID significantly outperformed COMPD. e.g. during a dose-response experiment with 12 compounds, 8 concentrations, and 3 replicates, PLAID constructs a layout in 6 seconds, while COMPD continues optimizing after 180 seconds.

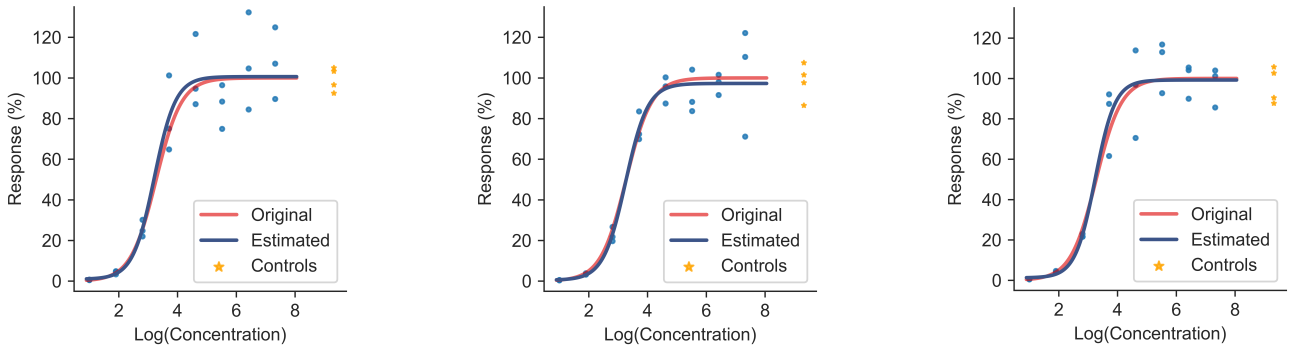
We observed that enabling the optional constraints doubles COMPD's performance. At the same time, when we enabled symmetry-breaking constraints, a small example was solved within 5 seconds. When disabled, it took 35 seconds to find an optimal solution without a proof, and we terminated the search for the proof after a few minutes.

[6]https://github.com/pharmbio/plaid/tree/multiplate-poc/simulations (Accessed: 2025-07-01)

[7]https://github.com/astra-uu-se/COMPD/tree/main/evaluation_aaai26 (Accessed: 2025-11-10)

[8].../evaluation_aaai26/detailed experimental results.pdf

[9].../evaluation_aaai26/dzn-files/memory_test.dzn

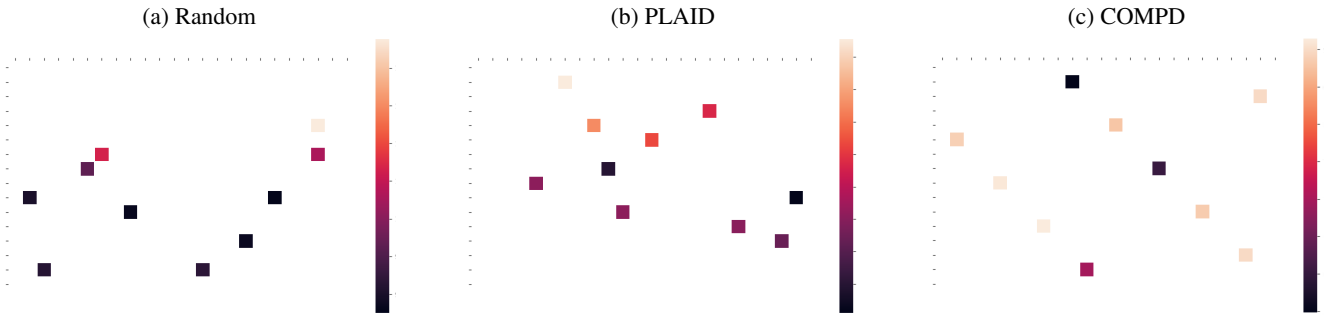


(a) $R^2 = 0.91$, $IC_{50} = 61.92$ (b) $R^2 = 0.95$, $IC_{50} = 55.48$ (c) $R^2 = 0.95$, $IC_{50} = 57.66$

Figure 5: An example of dose response curves for simulated experiments using 8 doses with 3 replicates (blue dots), 4 negative controls (yellow stars), and strong bowl-shaped plate effects on layouts constructed (a) randomly, (b) with PLAID and (c) with COMPD. The expected $IC_{50} = 53.43$. On the caption below each plot we show the obtained R^2 and IC_{50} values.

	Hit-Rate	1%	5%	10%	20%	30%	40%
ROC -AUC	Random	0.93 ± (0.004)	0.93 ± (0.007)	0.93 ± (0.004)	0.93 ± (0.003)	0.93 ± (0.002)	0.93 ± (0.003)
	PLAID	0.96 ± (0.002)	0.96 ± (0.004)	0.96 ± (0.002)	0.96 ± (0.002)	0.96 ± (0.002)	0.97 ± (0.002)
	COMPD	**0.97 ± (0.002)**	**0.98 ± (0.002)**	**0.98 ± (0.002)**	**0.98 ± (0.001)**	**0.98 ± (0.001)**	**0.98 ± (0.001)**
PR -AUC	Random	0.68 ± (0.032)	0.77 ± (0.017)	0.82 ± (0.017)	0.87 ± (0.005)	0.91 ± (0.003)	0.93 ± (0.004)
	PLAID	0.78 ± (0.009)	0.84 ± (0.008)	0.88 ± (0.005)	0.92 ± (0.004)	0.94 ± (0.002)	0.96 ± (0.002)
	COMPD	**0.84 ± (0.007)**	**0.9 ± (0.007)**	**0.92 ± (0.005)**	**0.95 ± (0.001)**	**0.97 ± (0.002)**	**0.97 ± (0.001)**

Table 1: Mean and standard deviation of the ROC-AUC and the PR-AUC obtained for 10 screening experiments with varying hit rates, using layouts with 10 positive and 10 negative controls in the presence of very strong bowl-shaped plate effects. Each screening contains 40 plates. Hits are randomly located on the plates.



(a) Random (b) PLAID (c) COMPD

Figure 6: Examples of placing 10 negative controls (a) randomly, (b) with PLAID and (c) with COMPD

6 Conclusion

Designing an effective layout, i.e., a layout that allows the detection and minimization of plate effects, is an important step for any biomedical research. To assist, we propose COMPD, a constraint model that, unlike its predecessor, PLAID, formulates the microplate design problem as an optimization problem instead of a CSP, allowing the solver to balance material allocation dynamically rather than imposing fixed threshold-based constraints. Related research appeared in CP (Pesant and Régin 2005) and operations research (Prokopyev, Kong, and Martinez-Torres 2009). COMPD integrates these ideas into a unified CP framework, overcoming limitations of PLAID listed in Introduction, and with demonstrated efficiency and scalability: e.g., COMPD outperformed PLAID during screening experiments. While the improvement is not as dramatic as between PLAID and randomized layouts, it allows for further cost reduction in large-scale biomedical research.

Future research includes the development of COMPD-specific search strategies to increase the performance, and further modifying and fine-tuning the optimization criteria, to ensure better placement of replicates of individual drugs.

References

Alvarsson, J.; Andersson, C.; Spjuth, O.; Larsson, R.; and Wikberg, J. E. 2011. Brunn: An open source laboratory information system for microplates with a graphical plate layout design process. *BMC bioinformatics*, 12: 1–8.

Beldiceanu, N.; Carlsson, M.; and Rampon, J.-X. 2012. Global constraint catalog, (revision a).

Birmingham, A.; Selfors, L. M.; Forster, T.; Wrobel, D.; Kennedy, C. J.; Shanks, E.; Santoyo-Lopez, J.; Dunican, D. J.; Long, A.; Kelleher, D.; et al. 2009. Statistical methods for analysis of high-throughput RNA interference screens. *Nature methods*, 6(8): 569–575.

Blay, V.; Tolani, B.; Ho, S. P.; and Arkin, M. R. 2020. High-throughput screening: today's biochemical and cell-based approaches. *Drug Discovery Today*, 25(10): 1807–1821.

Brailsford, S. C.; Potts, C. N.; and Smith, B. M. 1999. Constraint satisfaction problems: Algorithms and applications. *European journal of operational research*, 119(3): 557–581.

Brower, S. L.; Fensterer, J. E.; and Bush, J. E. 2008. The ChemoFx® assay: an ex vivo chemosensitivity and resistance assay for predicting patient response to cancer chemotherapy. *Apoptosis and Cancer: Methods and Protocols*, 57–78.

Caicedo, J. C.; Cooper, S.; Heigwer, F.; Warchal, S.; Qiu, P.; Molnar, C.; Vasilevich, A. S.; Barry, J. D.; Bansal, H. S.; Kraus, O.; et al. 2017. Data-analysis strategies for image-based cell profiling. *Nature methods*, 14(9): 849–863.

Connelly, B. 2014. Plotting Microtiter Plate Maps. http://pfigshare-u-files.s3.amazonaws.com/1486138/PlottingMicrotiterPlateMaps.pdf. Accessed: 2025-07-01.

Delorme, V.; Woo, M.; Falcão, V. C. d. A.; and Wood, C. 2021. PlateEditor: A web-based application for the management of multi-well plate layouts and associated data. *Plos one*, 16(5): e0252488.

FlowJo, L. 2017. FlowJo. https://www.flowjo.com/. Accessed: 2025-07-01.

Franchina, L.; and Sergiani, F. 2020. High quality dataset for machine learning in the business intelligence domain. In *Intelligent Systems and Applications: Proceedings of the 2019 Intelligent Systems Conference (IntelliSys) Volume 1*, 391–401. Springer.

Francisco Rodríguez, M. A.; Puigvert, J. C.; and Spjuth, O. 2023. Designing microplate layouts using artificial intelligence. *Artificial Intelligence in the Life Sciences*, 3: 100073.

LabForward, G. 2025. Well Plate Templates. https://www.labfolder.com/well-plate-templates. Accessed: 2025-07-01.

Lu, Y.; and Young, S. 2020. A survey of public datasets for computer vision tasks in precision agriculture. *Computers and Electronics in Agriculture*, 178: 105760.

Mpindi, J.-P.; Swapnil, P.; Dmitrii, B.; Jani, S.; Saeed, K.; Wennerberg, K.; Aittokallio, T.; Östling, P.; and Kallioniemi, O. 2015. Impact of normalization methods on high-throughput screening data with high hit rates and drug testing with dose–response data. *Bioinformatics*, 31(23): 3815–3821.

Pesant, G.; and Régin, J.-C. 2005. Spread: A balancing constraint based on statistics. In *International Conference on Principles and Practice of Constraint Programming*, 460–474. Springer.

Prokopyev, O. A.; Kong, N.; and Martinez-Torres, D. L. 2009. The equitable dispersion problem. *European Journal of Operational Research*, 197(1): 59–67.

Roselle, C.; Verch, T.; and Shank-Retzlaff, M. 2016. Mitigation of microtiter plate positioning effects using a block randomization scheme. *Analytical and bioanalytical chemistry*, 408: 3969–3979.

Schiff, L.; Migliori, B.; Chen, Y.; Carter, D.; Bonilla, C.; Hall, J.; Fan, M.; Tam, E.; Ahadi, S.; Fischbacher, B.; et al. 2022. Integrating deep learning and unbiased automated high-content screening to identify complex disease signatures in human fibroblasts. *Nature communications*, 13(1): 1590.

Suprun, M.; and Suárez-Fariñas, M. 2019. PlateDesigner: a web-based application for the design of microplate experiments. *Bioinformatics*, 35(9): 1605–1607.

Tegally, H.; San, J. E.; Giandhari, J.; and de Oliveira, T. 2020. Unlocking the efficiency of genomics laboratories with robotic liquid-handling. *BMC genomics*, 21: 1–15.

Torres-Acosta, M. A.; Lye, G. J.; and Dikicioglu, D. 2022. Automated liquid-handling operations for robust, resilient, and efficient bio-based laboratory practices. *Biochemical Engineering Journal*, 188: 108713.

Vandenberg, L. N. 2022. Low dose effects and nonmonotonic dose responses for endocrine disruptors. In *Endocrine disruption and human health*, 141–163. Elsevier.

Wikimedia Commons. 2003. Microplate. https://en.wikipedia.org/wiki/Microplate. Accessed: 2025-07-01.

Williams, A.; Shrine, N.; Naghra-van Gijzel, H.; Betts, J.; Chen, J.; Hessel, E.; John, C.; Packer, R.; Reeve, N.; Yeo, A.; Abner, E.; Åsvold, B.; Auvinen, J.; Bartz, T.; Bradford, Y.; Brumpton, B.; Campbell, A.; Cho, M.; Chu, S.; Crosslin, D.; Feng, Q.; Esko, T.; Gharib, S.; Hayward, C.; Hebbring, S.; Hveem, K.; Järvelin, M.; Jarvik, G.; Landis, S.; Larson, E.; Liu, J.; Loos, R.; Luo, Y.; Moscati, A.; Mullerova, H.; Namjou, B.; Porteous, D.; Quint, J.; null, n.; Ritchie, M.; Sliz, E.; Stanaway, I.; Thomas, L.; Wilson, J.; Hall, I.; Wain, L.; Michalovich, D.; and Tobin, M. 2023. Genome-wide association study of susceptibility to hospitalised respiratory infections [version 2; peer review: 1 approved, 2 approved with reservations]. *Wellcome Open Research*, 6(290).

Williams, S. P.; Gould, C. M.; Nowell, C. J.; Karnezis, T.; Achen, M. G.; Simpson, K. J.; and Stacker, S. A. 2017. Systematic high-content genome-wide RNAi screens of endothelial cell migration and morphology. *Scientific Data*, 4(1): 1–11.

Zhang, X. D. 2008. Novel analytic criteria and effective plate designs for quality control in genome-scale RNAi screens. *Journal of biomolecular screening*, 13(5): 363–377.

Zhang, X. D. 2011. *Optimal high-throughput screening: practical experimental design and data analysis for genome-scale RNAi research*. Cambridge University Press.