Merging Combinatorial Design and Optimization: the Oberwolfach Problem

Fabio Salassa\textsuperscript{a}, Gabriele Dragotto\textsuperscript{c,a,*}, Tommaso Traetta\textsuperscript{d}, Marco Buratti\textsuperscript{e}, Federico Della Croce\textsuperscript{a,b}

\textsuperscript{a}\{fabio.salassa, federico.dellacroce\}@polito.it
Dipartimento di Ingegneria Gestionale e della Produzione, Politecnico di Torino (Italy)

\textsuperscript{b}CNR, IEIIT, Torino, Italy

\textsuperscript{c}gabriele.dragotto@polymtl.ca
Canada Excellence Research Chair in Data Science for Real-time Decision-making, École Polytechnique de Montréal (Canada)

\textsuperscript{d}tommaso.traetta@unibs.it
DICATAM, Università degli Studi di Brescia (Italy)

\textsuperscript{e}marco.buratti@unipg.it
Dipartimento di Matematica e Informatica, Università degli Studi di Perugia (Italy)

Abstract

The Oberwolfach Problem $OP(F)$ – posed by Gerhard Ringel in 1967 – is a paradigmatic Combinatorial Design problem asking whether the complete graph $K_v$ decomposes into edge-disjoint copies of a 2-regular graph $F$ of order $v$. In this paper, we provide all the necessary equipment to generate solutions to $OP(F)$ for relatively small orders by using the so-called difference methods. From the theoretical standpoint, we present new insights on the combinatorial structures involved in the solution of the problem. Computationally, we provide a full recipe whose base ingredients are advanced optimization models and tailored algorithms. This algorithmic arsenal can solve the $OP(F)$ for all possible orders up to 60 with the modest computing resources of a personal computer. The new 20 orders, from 41 to 60, encompass 241200 instances of the Oberwolfach Problem, which is 22 times greater than those solved in previous contributions.

Keywords: Combinatorics, Graph Factorization, Integer Programming, Combinatorial Design, Oberwolfach Problem, Constraint Programming

\*Corresponding author.
1. Introduction

Gerhard Ringel proposed the Oberwolfach Problem (OP) for the first time in 1967 [18], while attending a conference at the Mathematical Research Institute of Oberwolfach, in Germany. In conferences held at the Institute, participants usually dine together in a room with circular tables of different sizes, and each participant has an assigned seat. Ringel asked whether there exists a seating arrangement for an odd number $v$ of people and $(v - 1)/2$ meals so that all pairs of participants are seated next to each other exactly once.

Formally, given a spanning 2-regular subgraph (a 2-factor) $F$ of $K_v$ (the complete graph of $v$ vertices), the Oberwolfach problem $OP(F)$ asks whether $K_v$ with $v$ odd decomposes into $(v - 1)/2$ edge-disjoint copies of $F$. We describe $F$ as $[\ldots, m_i c_i, \ldots]$ – a list of cycles with their multiplicity – where each $m_i$ is the multiplicity of $c_i$-length cycles in $F$. For the sake of simplicity, we omit any $c_i$ for which $m_i = 0$, and we remove the superscript when $m_i = 1$ for a given $i$.

In 1979, Huang et al. [21] extended the problem to the case where $v$ is even. Although $OP$ has drawn interest, and much progress has been made over the past few years (see, for instance, [5, 6, 8, 17, 19, 22, 27]), a complete solution has yet to be found. A survey of the most relevant results on this problem, updated to 2006, can be found in [13].

Solutions to $OP$ can often be found by focusing on those having symmetries with a particular action on the vertex set. By knowing the structure of these solutions, the problem of finding edge-disjoint 2-factors turns into finding a few well-structured 2-factors. The so-called difference methods – a family of algebraic tools – set the rules to construct such well-structured 2-factors. Difference methods were introduced for the first time by Anstice [3] to generalize solutions to Kirkman’s 15 schoolgirls problem, one of the paradigmatic problems in Combinatorial Design Theory. Arranging seats around tables is not new for Operations Research as well. García et al. [16], for instance, introduced a table placement problem aiming to maximize a measure of social benefit.

The baseline of this work is the contribution of Deza et al. [15]. There, authors solved $OP$ for $18 \leq v \leq 40$, modeling difference methods with undisclosed algorithms and tests which were carried out on a high-performance computing cluster [12].

In this paper, we provide all the necessary equipment to generate solutions to an Oberwolfach problem of a relatively small order; not only on the theoretical side but, most importantly and differently from Deza et al.
also on the computational side. Indeed we give a full recipe whose base ingredients are advanced optimization algorithms rather than an exhaustive search. These algorithms allowed us to rapidly obtain the desired solutions for all possible orders $v \leq 60$ with a personal computer which, we point out, is a much lower performance threshold compared to a high-performance computing cluster of Deza et al. [15] in 2008. We also recall that the number of generated solutions increases with the order of the problem, for instance, from 1756 partitions of $v = 40$ to 33552 partitions of $v = 60$.

Our approach blends combinatorial design theory with optimization and computation paradigms. We model difference methods as Constraint Programming (CP) problems, and leverage on state-of-the-art algorithms to find the combinatorial solutions. We were able to generate complete solutions for OP when $v \leq 60$. Our approach solves a generic instance within 5 seconds on a standard computer, compared to the need for a high-performance computing cluster [15]. The extensive computational testing allowed us to derive new theoretical results for OP, in particular, a new necessary condition was detected on the existence of 1-rotational solutions. Moreover, an Integer Programming (IP) model verifies the non-existence of solutions for $OP(23,5)$.

In a nutshell, the above optimization tools enabled us to solve large OP instances in limited CPU times and at the same time to derive theoretical results for general classes of instances. We believe such an approach could be generalized to a broader class of Combinatorial Design problems.

The paper proceeds as follows. In Section 2, since this work is at the intersection of two distinct domains, Combinatorial Design Theory and Combinatorial Optimization, we introduce a standard tool pertaining to the former (difference methods) by means of an illustrative example. Section 3 presents how to construct well-structured 2-factors for the solution of $OP(F)$. Section 4 shows how to translate into CP models the findings of section 3. Section 5 provides the outcome of the experimental testing. Section 6 concludes the paper with final remarks.

2. Difference methods and $OP(F)$: an illustrative example

Difference methods exploit the symmetries of a 2-factorization and tell us how to construct one well-structured 2-factor which yields, by translation, the complete set of 2-factors giving a solution to $OP(F)$. To explain it in the context of the Ringel’s informal formulation, we can construct, for instance, the first meal seating arrangement (the desired well-structured 2-factor) and derive/translate from it the remaining ones. In the following example, we
provide a well-structured 2-factor for $OP(3, 6)$, and show how the remaining meals can be derived starting from it.

Figure 1: A first meal arrangement for $OP(F = [3, 6])$.

Figure 1 depicts the seating arrangement of the first meal (see Section 3.1 Proposition 3.1) of $OP(F)$, where $F$ contains two cycles (dinner tables) of size 3 and 6, thus $F = [3, 6]$ and $v = 9$.

We label the vertices of $F$ with the elements of $Z_8 \cup \{\infty\}$ and for each edge incident with two vertices different from $\infty$, say $i$ and $j$, the two differences $i - j$ and $j - i$ (both mod $(v - 1)$) have to be calculated. For instance, if we consider the vertices labeled 2 and 1 in Figure 1, we obtain the differences 1 and $-1 \equiv 7 \pmod{8}$. The list $\Delta F$ of all possible differences between adjacent vertices of $F$, different from $\infty$, contains every element in $\{1, 2, \ldots, 7\}$ with multiplicity 2. Furthermore, $F + 4 = F$, where $F + 4$ is the graph obtained from $F$ by adding 4 to every vertex but $\infty$. In other words, we have found a vertex labeling of $F$ such that $\Delta F$ contains every element in $\{1, 2, \ldots, 7\}$ with multiplicity 2, and $F + 4 = F$. These are the two crucial conditions which guarantee that $F$ is the sought-after 2-factor (see Proposition 3.1) which will generate a solution to $OP(F)$. Indeed, the set $F = \{F, F + 1, F + 2, F + 3\}$ of all distinct translates of $F$ (see Figure 2) gives us a set of edge-disjoint copies of $F$ which decompose $K_9$, that is, a solution to $OP(F)$.

In the following section we provide conditions to find a well-structured 2-factor $F$ which guarantee that all its distinct translates yield a solution to $OP(F)$. In Section 4 these conditions are then reformulated as CP models to be tackled by a solver to generate solutions (i.e. first meal arrangements).
Figure 2: The remaining meals arrangements for $OP(F = [3, 6])$.

3. Constructions of well-structured 2-factors

A graph has a 2-factorization if and only if it is regular and of even degree, as Petersen [23] shows. However, given a particular 2-factor $F$, if we ask for a 2-factorization whose factors are all isomorphic to $F$, then the problem becomes much harder. Our focus is on constructing 2-factorizations of $K^*_v$, which is the complete graph $K_v$ of order $v$ when $v$ is odd, or $K_v - I$, i.e. the complete graph $K_v$ minus the 1-factor $I$, when $v$ is even. Given a
2-factor $F$ of order $v$, the Oberwolfach problem on $F$ $(OP(F))$ asks for a 2-factorization of $K^*_v$ into copies of $F$.

A solution to $OP(F)$ exists whenever the order of $F$ is less than 40 [15], except only when $F \in \{[2,3], [4,3], [4,5], [2,3,5]\}$. These are the only known cases in which the problem is not solvable. We point out that even though Piotrowski [25] self-cites for a computer-based proof of the non-existence of a solution to $OP([2,3,5])$, no published proof exists. $OP(F)$ has also been solved when $F$ is a uniform 2-factor (i.e., $F$ consists of cycles of the same length) [1, 2, 20], when $F$ is bipartite (i.e., $F$ contains only cycles of even lengths) [1, 5], when $F$ has exactly two cycles [27], or for an infinite family of prime orders [6]. In addition, [7, 8, 10] studied solutions having symmetries with a prescribed action on the vertices, and existence for sufficiently large $v$ can be found in [17]. Furthermore, a solution to the infinite variant of the Oberwolfach problem can be found in [14]. However, the problem is still open in general, and [13, Section VI.12] provides a detailed survey on this subject, updated to 2006.

3.1. 1-rotational solutions to $OP(F)$

Buratti and Rinaldi [8] construct 1-rotational solutions in the odd case, that is, 2-factorizations of $K_v$ with $v$ odd, with a well-behaved automorphism group. Let $v = 2n + 1$ be a positive integer, let $\mathbb{Z}_{2n}$ denote the group of integers modulo $2n$, and set $V = \mathbb{Z}_{2n} \cup \{\infty\}$. The list of differences of a subgraph $\Gamma$ of $K_V$ is the multiset $\Delta \Gamma$ of all possible differences between pairs of adjacent vertices of $\Gamma$, excluding the vertex $\infty$, namely:

$$\Delta \Gamma = \{x - y \mid [x, y] \in E(\Gamma \setminus \{\infty\})\}.$$  (1)

We use the notation $[x, y]$ to denote an edge between the nodes $x$ and $y$. Also, for every $g \in \mathbb{Z}_{2n}$, we denote by $\Gamma + g$ the graph with vertex set $V$ obtained from $\Gamma$ by replacing each vertex $x \neq \infty$ with $x + g$.

The following result, proven in [8], provides conditions which guarantee the existence of a solution to $OP(F)$.

**Proposition 3.1.** Let $F$ be a 2-regular graph satisfying the following properties:

1. $V(F) = \mathbb{Z}_{2n} \cup \{\infty\}$,
2. $\Delta F \supset \mathbb{Z}_{2n} \setminus \{0\}$,
3. $F + n = F$. 

6
Then \( \mathcal{F} = \{ F + g \mid g \in \mathbb{Z}_{2n} \} \) is a solution of \( \text{OP}(F) \).

A factorization \( \mathcal{F} \) of \( K_{2n+1} \) constructed as in Proposition 3.1 is called 1-rotational, since the permutation group \( G = \{ \tau_g \mid g \in \mathbb{Z}_{2n} \} \) of \( V \), where \( \tau_g \) fixes \( \infty \) and maps \( x \in \mathbb{Z}_{2n} \) to \( x + g \), is an automorphism group of \( \mathcal{F} \) whose action on \( V \setminus \{ \infty \} \) creates only one orbit.

In [11, Proposition 2.5] it is shown that if there exists a 2-regular graph \( F = [\ell_1, \ell_2, \ldots, \ell_s] \) of order \( 2n + 1 \) satisfying the assumptions of Proposition 3.1 then there exists a solution to \( \text{OP}([\ell_1 + 1, \ell_2, \ldots, \ell_s]) \) where \( \ell_1 \) is the length of the cycle of \( F \) through \( \infty \). The following result weakens this condition by showing that \( \ell_1 \) can be the length of any cycle of \( F \) provided that it contains a suitable difference.

**Proposition 3.2.** Let \( F = [\ell_1, \ell_2, \ldots, \ell_s] \) be a 2-regular graph of order \( 2n + 1 \) satisfying the assumptions of Proposition 3.1. If \( C \) is an \( \ell_1 \)-cycle of \( F \) such that \( \Delta C \) contains an element of \( \mathbb{Z}_{2n} \) of order 2 (mod 4), then there exists a solution to \( \text{OP}([\ell_1 + 1, \ell_2, \ldots, \ell_s]) \).

**Proof.** Let \( C = (c_1, c_2, \ldots, c_{\ell_1}) \) be the \( \ell_1 \)-cycle of \( F \) whose list of differences contains, by assumption, an element \( x \) of order \( u \equiv 2 \pmod{4} \). Without loss of generality, we can assume that \( x = c_1 - c_2 \).

Let \( \mathcal{G} = \{ 2x \cdot i + j \mid i \in \{0,1,\ldots,u/2-1\}, j \in \{0,1,\ldots,2n/u - 1\} \} \) and recall that, by the definition of order, \( u \) is the smallest positive integer such that \( xu \equiv 0 \pmod{2n} \); hence \( xu/2 \equiv n \pmod{2n} \). Therefore, it is not difficult to check that

\[
\text{both } \{ \mathcal{G}, \mathcal{G} + x \} \text{ and } \{ \mathcal{G}, \mathcal{G} + n \} \text{ are partitions of } \mathbb{Z}_{2n}. \tag{2}
\]

Considering that \( F \) satisfies the assumptions of Proposition 3.1, we have that \( \mathcal{F} = \{ F + g \mid g \in \mathbb{Z}_{2n} \} \) is a solution of \( \text{OP}(F) \). By taking into account \( \text{(2)} \) and recalling that \( F + n = F \), it follows that \( \mathcal{F} = \{ F + g \mid g \in \mathcal{G} \} \).

Now set \( C' = (c_1, \infty', c_2, \ldots, c_{\ell_1}) \), where \( \infty' \not\in \mathbb{Z}_{2n} \cup \{ \infty \} \), and let \( F' \) be the 2-regular graph of order \( 2n + 1 \) obtained from \( F \) by replacing \( C \) with \( C' \). Finally, set \( \mathcal{F}' = \{ F' + g \mid g \in \mathcal{G} \} \), where \( F' + g \) is the graph obtained from \( F' \) by replacing each vertex \( v \not\in \{ \infty, \infty' \} \) with \( v + g \), and set \( V = \mathbb{Z}_{2n} \cup \{ \infty, \infty' \} \).

In order to prove that \( \text{OP}(F') \) has a solution, we will show that \( \mathcal{F}' \) is a 2-factorization of \( K_{2n+2} = K_V - I \), where \( I = \{ [\infty, \infty'] \} \cup \{ [g, x + g] \mid g \in \mathcal{G} \} \). Note that \( \text{(2)} \) guarantees that \( I \) is a 1-factor of \( K_V \). Also, since \( F' \) contains all the edges of \( F \) except only for \( [c_1, c_2] \), and considering that

\[
\{ [c_2 + g, c_1 + g] \mid g \in \mathcal{G} \} = \{ [g, x + g] \mid g \in \mathcal{G} \} \subset I,
\]
it follows that every edge of $K - I$ of the form $[a, b]$ with $a \neq \infty' \neq b$ belongs to exactly one cycle of $F'$. Finally, again by (2) we have that 
\[ \{ [\infty', c_1 + g], [\infty', c_2 + g] \mid g \in G \} = \{ [\infty', \ell] \mid \ell \in \mathbb{Z}_{2n} \}, \]
therefore every edge of $K_{2n+2}^*$ of the form $[\infty', \ell]$ with $\ell \in \mathbb{Z}_{2n}$ belongs to exactly one cycle of $F'$. Hence, $F'$ is a 2-factorization of $K_{2n+2}^*$ and this completes the proof.

We now provide two necessary conditions for the existence of a 2-regular graph satisfying the assumptions of Proposition 3.1.

**Proposition 3.3.** If $F = [a_1\ell_1, a_2\ell_2, \ldots, a_s\ell_s]$ is a 2-regular graph of odd order $2n + 1$ satisfying the assumptions of Proposition 3.1, then
\[ \{|i \mid a_i\ell_i \text{ is odd}\} = 1. \]

**Proof.** In [8, Proposition 3.4], the authors show that
1. the cycle of $F$ passing through $\infty$ has odd length, and
2. if $C$ is any other cycle of $F$ such that $C + n = C$, then $C$ has even length.

Therefore, if $C$ is an odd length cycle of $F$ not passing through $\infty$, then $C \neq C + n \in F$. In other words, if $a_i$ denotes the number of cycles of $F$ of odd length $\ell_i$, then $a_i$ is even, unless $\ell_i$ is the length of the cycle through $\infty$ and the assertion follows.

**Remark 3.1.** Let $C$ be a cycle with vertices in $V = \mathbb{Z}_{2n} \cup \{\infty\}$ such that $C = C + n$. It is not difficult to check that $C$ has one of the following forms:

(a) $C = (\infty, c_1, \ldots, c_{m-1}, c_m, c_m + n, c_{m-1} + n, \ldots, c_1 + n)$,
(b) $C = (c_1, \ldots, c_{m-1}, c_m, c_m + n, c_{m-1} + n, \ldots, c_1 + n)$, and $\infty \notin V(C)$,
(c) $C = (c_1, c_2, \ldots, c_m, c_1 + n, c_2 + n, \ldots, c_m + n)$, and $\infty \notin V(C)$.

In the first two cases, the translation by $n$ acts on $C$ as a reflection, while in the latter case such an action is a rotation. In [8, Proposition 3.7], it is shown in particular that a 2-factor of $K_V$ satisfying the assumptions of Proposition 3.1 contains exactly one cycle on which the translation by $n$ acts as a reflection, which then coincides with the cycle through $\infty$. Therefore, any cycle $C$ of $F$ fixed by $n$ and not passing through $\infty$ has the same form as in (c).

The following result can be seen as a generalization of [9, Theorem 3.2].
Proposition 3.4. Let $F$ be a 2-regular graph of order $2n+1$ and let $r$ denote the number of cycles in $F$ of even length. If $F$ satisfies the assumptions of Proposition 3.1 and its cycle passing through $\infty$ has length 3, then either $n \equiv 0 \pmod{4}$ or $\frac{n-1}{2} + r$ is an even integer.

Proof. Let $F$ be a 2-regular graph of order $2n+1$ such that

1. $V(F) = \mathbb{Z}_{2n} \cup \{\infty\}$,
2. $\Delta F \supset \mathbb{Z}_{2n} \setminus \{0\}$,
3. $F + n = F$,

and let $C_\infty$ denote the cycle of $F$ through $\infty$. By assumption, $C_\infty$ has length 3, and by conditions 1 and 2 we have that $C_\infty + n = C_\infty$; hence $C_\infty = (\infty, g, g + n)$ for some $g \in \mathbb{Z}_{2n}$.

Let $C_1, C_2, \ldots, C_u$ be the list of the cycles in $F$ distinct from $C_\infty$, with $C_i = (c_{i,1}, c_{i,2}, \ldots, c_{i,\ell_i})$ for every $1 \leq i \leq u$. By condition 3, we can assume without loss of generality that $C_i + n = C_i$ when $1 \leq i \leq s$, and $C_i + n = C_{i+t}$ when $s+1 \leq i \leq s+t$, where $u = s + 2t$. Hence, for $1 \leq i \leq s$ we have that $\ell_i$ is even, and by Remark 3.1 we obtain that $c_{i,j+\ell_i/2} = c_{i,j} + n$ for every $1 \leq j \leq \ell_i/2$. Now set $m_i = \ell_i/2$ when $1 \leq i \leq s$, otherwise set $m_i = \ell_i$. Also, let $d_{i,j} = c_{i,j+1} - c_{i,j}$ (where the subscript $j$ is computed modulo $\ell_i$) and set $D_i = \{d_{i,j} \mid 1 \leq j \leq m_i\}$ for every $1 \leq i \leq s + t$. Considering that any translation preserves the differences, we have that

$$d \in \Delta C_i, \text{ if and only if } d \in \pm D_i.$$ 

for every $1 \leq i \leq s + t$. By recalling condition 2, and considering that $\Delta C_\infty = \{\pm n\}$ and $\sum_{i=1}^{s+t} m_i = n-1$, it follows that $\mathbb{Z}_{2n} \setminus \{0, n\} = \bigcup_{i=1}^{s+t} (\pm D_i)$.

Also, since $c_{i,1} + \sum_{j=1}^{m_i} d_{i,j} = c_{i,m_i+1}$, we have that $\sum_{j=1}^{m_i} d_{i,j} = n$ when $1 \leq i \leq s$, otherwise $\sum_{j=1}^{m_i} d_{i,j} = 0$; hence $\sum_{i=1}^{s+t} \sum_{j=1}^{m_i} d_{i,j} = sn$. Finally, considering that $\mathbb{Z}_{2n} \setminus \{0, n\}$ contains $2\left\lfloor \frac{n}{2}\right\rfloor$ odd integers and $-x \neq x$ for every $x \in \mathbb{Z}_{2n} \setminus \{0, n\}$, it follows that $\bigcup_{i=1}^{s+t} D_i$ contains exactly $\left\lfloor \frac{n}{2}\right\rfloor$ odd integers, therefore

$$sn \equiv \left\lfloor \frac{n}{2}\right\rfloor \pmod{2}.$$ 

If $n$ is even, then $n \equiv 0 \pmod{4}$. If $n$ is odd, then $\frac{n-1}{2} \equiv s \pmod{2}$. Denoting by $s'$ the number of even length cycles in $\{C_{s+1}, C_{s+2}, \ldots, C_{s+t}\}$ and recalling that $C_i \neq C_i + n \in F$ for $s + 1 \leq i \leq s + t$, it follows that the total number of even length cycles in $F$ is $r = s + 2s'$, hence $\frac{n-1}{2} \equiv r \pmod{2}$, that is, $\frac{n-1}{2} + r$ is even, and the assertion is proven. $\square$
Proposition 3.1 tells us how to construct 1-rotational solutions of order $2n + 1$. These can then be used, following Proposition 3.2, to construct solutions of order $2n + 2$. Finally, Propositions 3.3 and 3.4 give us necessary conditions for a 1-rotational solution to exist.

We use the above results to construct 1-rotational solutions to $OP(F)$ whenever $F$ is a 2-regular graph of order $4t + 1$, thus $n = 2t$ and $t \in \mathbb{N}$. Equation (4) defines $F$ as the graph containing $u_i$ cycles of odd length $l_i$, and $w_j$ cycles of even length $m_j$, for every $i \in \{1, 2, \ldots, h\}$ and $j \in \{1, 2, \ldots, k\}$. Recalling the necessary condition in Proposition 3.3, we have

$$F = [u_1 l_1, \ldots, u_h l_h; w_1 m_1, \ldots, w_k m_k] : |\{i \mid u_i \text{ is odd}\}| = 1.$$  (4)

The graph $F$ must also satisfy Equation (5), which implements Proposition 3.4.

$$\exists i : (l_i = 3 \land u_i \text{ is odd}) \Rightarrow 2t \equiv 0 \pmod{4} \lor \left(\frac{2t - 1}{2} + \sum_{i=1}^{k} w_i\right) \equiv 0 \pmod{2}.$$  (5)

The symmetries stated in Remark 3.1 reduce the labeling problem on $F$ to a simpler one on a new graph $F^*$, the asymmetric version of $F$, which can be seen as the union of 2 subsets, namely the set of paths ($\mathcal{P}$) and the set of cycles ($\mathcal{C}$). To better describe the structure of $F^*$, we assume without loss of generality that $u_1$ is odd, and the remaining $u_i$ are even. Hence we can write $u_1 = 2a_1 + 1$, $u_i = 2a_i$ for every $i \in \{2, 3, \ldots, h\}$, and $m_j = 2\mu_j$ for every $j \in \{1, 2, \ldots, k\}$. Thus Equation (6) describes the structure of the reduced graph $F^*$,

$$F^* = \mathcal{C} \cup \mathcal{P},$$

$$\mathcal{C} = [(l_1 + 1)/2, a_1 l_1, a_2 l_2, \ldots, a_h l_h],$$

$$\mathcal{P} = [[w_1 \mu_1, \ldots, w_k \mu_k]].$$

where $\mathcal{P}$ is the graph containing $w_j$ paths with $\mu_j$ edges, for every $j \in \{1, 2, \ldots, k\}$, and they are pairwise vertex-disjoint. $\mathcal{C}$ also contains a cycle with $(l_1 - 1)/2$ nodes, namely the one with infinity.

In a more general and intuitive way, the rationale for $F^*$ is as follows. On the one hand, we consider the cycles of odd length in $F$. If there is an even number $u_i \equiv 0 \pmod{2}$ of cycles of odd length $l_i$, these contribute in
$F^*$ with $u_i/2$ cycles of length $l_i$ in $C$. The only leftover cycle is of odd length $u_\infty$ (since the number of nodes is $4t+1$) is the one with $\infty$, which contributes with an open chain of length $(u_\infty - 1)/2$ in $P$. On the other hand, an even number $w_j \equiv 0 \pmod{2}$ of cycles of even length $m_j$ are represented in $F^*$ as $w_j/2$ cycles of length $m_j$ in $C$, similar to an even number of odd cycles. Any cycle of even length $m_j$ with single multiplicity $w_j = 1$ becomes a chain of length $m_j/2$ in $P$.

Note that the number of edges of $F^*$ is $2t - 1$. We seek to determine a vertex labeling of $F^*$ with the elements of $\mathbb{Z}_{4t}$ such that

1. $V(F^*)$ contains exactly one element in $\{x, x + 2t\}$, for every $x \in \mathbb{Z}_{4t}$,
2. $\Delta F^* = \mathbb{Z}_{4t} \setminus \{0, 2t\}$.

This labeling of the vertices of $F^*$ leads to a labeling of $F$ satisfying Proposition 3.1 and hence to a solution for $OP(F)$ (see Figure 3).

### 3.2. (Almost) 2-rotational solutions to $OP(F)$

Here we describe a method to construct solutions to $OP(F)$ in all cases where there is no 1-rotational solution and, in particular, when the necessary conditions of Propositions 3.3 and 3.4 do not hold.

Let $v = 2n + 1$ be a positive integer, and set $V = (\{0, 1\} \times \mathbb{Z}_n) \cup \{\infty\}$. For every subgraph $\Gamma$ of $K_V$ and for every $i, j \in \{0, 1\}$, let $\Delta_{ij}\Gamma$ be the list of $(i, j)$-differences of $\Gamma$ defined below:

$$
\Delta_{ij}\Gamma = \{x - y \mid [(i, x), (j, y)] \in E(\Gamma \setminus \{\infty\})\}.
$$

(9)

For every $g \in \mathbb{Z}_n$ we denote by $\Gamma + g$ the graph with vertex set $V$ obtained from $\Gamma$ by replacing each vertex $(i, x)$ with $(i, x + g)$.

The following result provides sufficient conditions for the existence of a solution to $OP(F)$.

**Proposition 3.5.** Let $F = [\ell_1, \ell_2, \ldots, \ell_s]$ be a 2-regular graph of order $2n+1$ satisfying the following conditions:

1. $V(F) = (\{0, 1\} \times \mathbb{Z}_n) \cup \{\infty\}$,
2. the vertices adjacent to $\infty$ are of the form $(0, x_0)$, $(1, x_1)$ for some $x_0, x_1 \in \mathbb{Z}_n$,
3. if $n$ is odd, then $\Delta_{00}F = \Delta_{11}F = \mathbb{Z}_n \setminus \{0\}$ and $\Delta_{01}F = \mathbb{Z}_n$,
4. if $n$ is even, then:
(a) $F$ contains the path $P = [(0,0), (0,n/2), (1,n/2), (1,0)]$ in one of its cycles,
(b) $\Delta_{ij}(F - P) = \mathbb{Z}_n \setminus \{0,n/2\}$ for every $(i,j) \in \{(0,0), (0,1), (1,1)\}$.

Then there exists a solution of $OP((\ell_1,\ell_2,\ldots,\ell_s))$. Furthermore, if $C$ is an $\ell_1$-cycle of $F$ such that $\Delta_{01}C$ contains an integer distinct from $n/2$, then there exists a solution to $OP([\ell_1+1,\ell_2,\ldots,\ell_s])$.

Proof. Let $F = \{F + g \mid g \in [1,n]\}$ when $n$ is odd, otherwise let $F = \{F + g, F^* + (n/2 + g) \mid 1 \leq g \leq n/2\}$, where $F^*$ is the 2-regular graph obtained by replacing the path $P$ (which is contained in $F$ by condition 1a) with $P^* = [(0,0), (1,n/2), (0,n/2), (1,0)]$. It is important to notice that in this case

$$F - P = F^* - P^*. \quad (10)$$

We claim that $F$ is a solution of $OP(F)$. By condition 1 and considering that the total number of edges (counted with their multiplicity) covered by $\mathcal{F}$ is $n|F| = n(2n + 1) = |E(K_{2n+1})|$, to prove the assertion it is enough to show that every edge of $K_V$, with $V = \{(0,1) \times \mathbb{Z}_n\} \cup \{\infty\}$, is contained in some 2-factor of $\mathcal{F}$.

Denoting with $(0,x_0)$ and $(1,x_1)$ the neighbours of $\infty$ in $F$ (condition 2), we have that $[\infty,(i,a)] \in F - x_i + a$ for every $(i,a) \in V \setminus \{\infty\}$. By recalling that (10) holds when $n$ is even, it follows that every edge of $K_V$ incident to $\infty$ belongs to some 2-factor of $\mathcal{F}$.

Now let $(i,a)$ and $(j,b)$ be two distinct vertices of $V \setminus \{\infty\}$ such that $a - b \neq n/2$ for $n$ even. By conditions 3 and 4, there exists an edge of $F$, say $[(i,a'),(j,b')]$ such that $a' - b' = a - b$. It follows that $[(i,a),(j,b)] = [(i,a'),(j,b')] + (b - b') \in F + (b - b')$. By taking into account (10) for $n$ even, we have that $[(i,a),(j,b)]$ belongs to some 2-factor of $\mathcal{F}$. It is not difficult to check that every edge of the form $[(i,a),(j,a + n/2)]$, with $1 \leq a \leq n/2$, is contained in $P + a$ or $P + (n/2 + a)$. Hence every edge of $K_V$ is contained in some 2-factor of $\mathcal{F}$ which is therefore a solution to $OP(F)$.

Now let $C = (c_0, c_1, \ldots, c_{\ell_1} - 1)$ be the $\ell_1$-cycle of $F$ such that $\Delta_{01}C$ contains an element distinct from $n/2$; in other words, $C$ contains an edge of the form $[(0,y_0),(1,y_1)]$ with $y_0 - y_1 \neq n/2$; hence, this edge does not belong to $P$. Without loss of generality, we can assume that $c_0 = (0,y_0)$ and $c_1 = (1,y_1)$.

Set $H$ and $H^*$ be the 2-regular graphs of order $2n + 2$ obtained from $F$ and $F^*$, respectively, by replacing the edge $[c_0,c_1]$ with the 2-path $[c_0,\infty',c_1]$, where $\infty' \notin V$. Also, $I = \{[\infty,\infty']\} \cup \{|c_0+g,c_1+g| \mid 1 \leq g \leq n\}$ is
clearly a 1-factor of $K_{2n+2} = K_{V \cup \{\infty\}}$. Finally, let $\mathcal{H} = \{H + g \mid g \in [1,n]\}$ when $n$ is odd, otherwise let $\mathcal{H} = \{H + g, H^* + (n/2 + g) \mid 1 \leq g \leq n/2\}$.

We claim that $\mathcal{H}$ is a solution to $OP([\ell_1 + 1, \ell_2, \ldots, \ell_s])$. Since $C$ is also a cycle of $F^*$ for $n$ even, both $H$ and $H^*$ are $2$-regular graphs of $K_{2n+2}$ isomorphic to $[\ell_1 + 1, \ell_2, \ldots, \ell_s]$. Also, considering that $F$ is a 2-factorization of $K_V$, every edge of $K_V \cup \{\infty\} - I$ not incident to $\infty'$ is contained in some 2-factor of $\mathcal{H}$. Since $H - P = H^* - P^*$ and $|\infty', (i, a)| = |\infty', c_i| + (a - y_i)$, it follows that every edge incident to $\infty'$ belongs to some 2-factor of $\mathcal{H}$, therefore $\mathcal{H}$ provides a solution to $OP([\ell_1 + 1, \ell_2, \ldots, \ell_s])$.

A factorization $F$ of $K_{2n+1} = K_V$, with $V = (\{0,1\} \times \mathbb{Z}_n) \cup \{\infty\}$, constructed as in Proposition 3.5 when $n$ is odd, is called $2$-rotational, since the permutation group $G = \{g \mid g \in \mathbb{Z}_n\}$ of $V$, where $\tau_g$ fixes $\infty$ and maps $(i, x) \in V \setminus \{\infty\}$ to $(i, x + g)$, is an automorphism group of $F$ whose action on $V \setminus \{\infty\}$ creates two orbits of size $n$.

The idea of constructing 2-factorizations of $K_{2n+1}$ with $n$ even, as described in Proposition 3.5, was first presented in [15].

4. Solving instances of the Oberwolfach Problem

4.1. Computing 1-rotational solutions

Recalling the content of Section 3.1, we propose a linear-time algorithm that implements Proposition 3.3 and the related Remark 3.1 and reduces $F$ to $F^*$. Afterward, $CP$ solves the problem over $F^*$, and therefore the labeling of $F$ is retrieved.

Algorithm 1 starts by reducing $F$ to $F^*$ with lines (5-8), where the only unpaired cycle $(u_i = 1 \mod 2$ as of Equation 4) of odd length $l_i$ reduces to a cycle of length $(l_i - 1)/2$ in $F^*$ (symmetry of case $a$). This latter cycle contains node $\infty$. In lines (9-12), a pair of 2 isomorphic $(u_i > 1)$ cycles of odd lengths $l_a$ and $l_b$ is reduced to a single cycle of length $l_i = l_a$ in $F^*$ (symmetry of case $c$). In the same way, in lines (15-17), a pair of 2 isomorphic $(w_i > 1)$ cycles of even length $m_a = m_b$ is reduced to a single cycle of length $m_i = m_a$ in $F^*$ (symmetry of case $c$). In lines (18-21), the remaining unpaired cycles $(w_i = 1)$ of even lengths $m_i$ are reduced to open chains of length $m_i/2$ in $F^*$ (symmetry of case $b$). We may derive $F$ from $F^*$ by an inverse constructive process.

Following the reduction, the $F^*$ labeling problem ($F^*$ LP) aims at finding a labeling for $F^*$ so that there is a solution to the complete $OP(F)$.

Problem 4.1 ($F^*$ Labeling). Let $F^* = (V, E)$ be a graph of order $|V(F^*)| = \gamma + 1$, and let $v_i \in V$ be an element in $G \cup \{\infty\}$ where $G = \mathbb{Z}_{2\gamma}$. Also, let
Algorithm 1: Reducing to $F^*$

1: **Input:** The original graph $F$
2: **Output:** The transformed version of the graph, $F^*$
3: infinite=false, and $V_{CP} = D_{CP}=0$
4: for $i$ in $u_i$ do ▷ Iterate through odd-length cycles
5: if $u_i \equiv 1 \pmod{2}$ &\& infinite=false then ▷ The cycle with infinite
6: Put a chain of length $(l_i - 1)/2$ in $F^*$ in position 0;
7: $u_i -; V_{CP} + = (l_i - 1)/2; D_{CP} = l_i/2 - 2;$
8: infinite=true;
9: else ▷ $u_i > 1$. Pair of odd-length cycles. Simplify one.
10: Put a cycle of length $l_i$ in $F^*$;
11: $u_i = u_i - 2; V_{CP} + = l_i; D_{CP} = l_i;$
12: end if
13: end for
14: for $i$ in $w_i$ do ▷ Iterate through even-length cycles.
15: if $w_i > 1$ then ▷ Pair of even-length cycles. Simplify one.
16: Put a cycle of length $m_i$ in $F^*$;
17: $w_i = w_i - 2; V_{CP} + = m_i; D_{CP} = m_i;$
18: else $w_i = 1$ ▷ Treat the cycle as a chain of half length.
19: Put a chain of length $m_i/2$ in $F^*$;
20: $w_i -; V_{CP} + = m_i/2; D_{CP} = m_i/2;$
21: end if
22: end for
23: return $F^*$

$F^* = O \cup C$ with $O$ set of open chains and $C$ set of closed cycles. For each node $i \in \tilde{V} = V \setminus \{\infty\}$ the $F^*$ Labeling Problem asks to assign a label $n_i \in G$ so that the following conditions hold:

1. Each element in the set $\tilde{V}$ receives a unique label from $G$.
   $n_\alpha \neq n_\beta$ for all $n_\alpha, n_\beta \in \tilde{V}$,
2. The set $\tilde{V}$ contains either label $n_\alpha$ or its $\gamma$-translated label.
   $n_\alpha \in \tilde{V} \lor n_\beta = n_\alpha + \gamma \pmod{2\gamma} \in \tilde{V}$ with $n_\alpha \in \mathbb{Z}_{2\gamma}$,
3. $\Delta F^*$ has cardinality $\lambda = 2\gamma - 2$ and contains all the elements in $G \setminus \{0, \gamma\}$
   with multiplicity 1. $|\Delta F^*| = 2(\gamma - 2) \land \Delta F^* = G \setminus \{0, \gamma\}$.

The corresponding CP model $[11][17]$ describes ($F^*$ LP).

We remark that alldifferent and card are typical CP operators on arrays of elements $[3]$. The first operator requires all array elements to exhibit different values. The second one, which takes an additional integer argument value $c$, imposes the cardinality of the integer value $c$ in the given array.
\begin{align*}
V &= \{n_i \mid n_i \in G\}, & (11) \\
\text{alldifferent}(V) & \quad \text{dom}(V) = [0, 2\gamma), \quad (12) \\
\text{card}(V|n_i) + \text{card}(V|(n_i + \gamma \mod 2\gamma)) &= 1 \quad \text{for all } n_i \in \mathbb{Z}_\gamma, \quad (13) \\
D &= dC \cup dO, \quad (14) \\
dC = \{(n_\alpha - n_\beta \mod 2\gamma)\} & \quad \text{for all } \alpha, \beta \in V \land [\alpha, \beta] \in C, \quad (15) \\
dO = \{\omega_i - \phi, \phi - \omega_i \mod 2\gamma\} & \quad \text{for all } a_i = [\omega_1, \ldots, \omega_i] \in O, \quad (16) \\
\phi &= \omega_1 + \gamma \mod 2\gamma, \\
\text{alldifferent}(D) & \quad \text{dom}(D) = (0, 2\gamma) \setminus \{\gamma\}. \quad (17)
\end{align*}

Equations (11)-(12) enforce Condition (1) in \((F^* \text{ LP})\), while (13) enforces Condition (2) for \(\gamma\)-translated labels. The difference-set is split into two subsets, as in Equation (14): \(dC\) in (15) for the edges over \(F^*\), and \(dO\) for open chains in (16). The virtual label \(\phi\) is reported in the latter subset. Finally, Constraint (17) enforces Condition (3) in \((F^* \text{ LP})\). Therefore, the problem of labeling \(F^*\) collapses to a feasible assignments of set \(V\), represented in (11).

**Example 4.1.** (referring to Problem 4.1) Consider an \(OP(F = [5, 2, 3, 2, 4, 6])\) of order \(4t + 1 = 25\) with \(t = 6\). \(F\) reduces to \(F^*\), and the simplified instance is \(OP(F^* = [3, 3, 4, 3, 3])\) where \(O = [3, 3]\) and \(3, 3\) is the cycle with \(\infty\). Therefore \(\gamma = 12\) and nodes in \(V(F^*)\) acquire their labels from \(\mathbb{Z}_{24} \cup \{\infty\}\). Figure 3 represents the reduced \(F^*\), with the virtual node \(12\). \(\Delta F^* = G \setminus \{0, 12\}\), hence the labeling is a valid solution for the \((F^* \text{ LP})\). Figure 4 represents the corresponding labeling for \(F\).

![Figure 3: \(F^*\) instance for \(F = [5, 2, 3, 2, 4, 6]\).](image)

A \(2 - \text{factor} \ F\) of order \(4t + 1\) which generates a \(1\)-rotational solution for \(OP(F)\) satisfies Proposition 3.1, 3.3, and 3.4. Correspondingly, a solution of order \(4t + 2\) can be derived from \(F\), according to Proposition 3.2 and
Figure 4: $F$ instance derived from $F^* = [3\infty, 3, 4, 3]$. 

its polynomial-time computation works as follows. Given $F = [l_1, l_2, ..., l_s]$ and $F' = [l_1 + 1, l_2, ..., l_s]$, a new node $\infty'$ joins the cycle $l_1$. The new node replaces a path $P = [c_1, c_2] \in l_1$ with a new path $P^* = [c_1, \infty', c_2]$ in the cycle $l_1$. Therefore, the difference-set of $F'$ omits values $c_1 - c_2$ and $c_2 - c_1$ (in modulo). For our computational tests, node $\infty'$ is arbitrarily inserted between two nodes $c_1, c_2$ so that $c_1 - c_2 = c_2 - c_1$ mod 2t. Solutions of order $4t + 2$ require the same computational effort as 1-rotational instances of order $4t + 1$. Moreover, we highlight how multiple instances of order $4t + 2$ originate from the same instance of order $4t + 1$ (see Example 4.2).

Example 4.2. Consider two instances of order $4t + 2$, such as OP$(19, 4, 3)$ and OP$(18, 5, 3)$ with $t = 6$. Both the instances originate from OP$(18, 4, 3)$, of order $4t + 1$.

4.2. Computing (almost) 2-rotational ($n$ odd)

This class of solutions derives from Proposition 3.5 (see also Section 3.2). Since that proposition distinguishes between odd and even values of $n$, we present the approach for odd values of $n$, and discard Condition (4) of Proposition 3.5. For even values of $n$, see the Appendix (6.1).

Given the 2-regular graph $F = (V, E)$ of order $2n + 1$, recalling that $n$ is odd, we write $n = 4t + 3$ where $t \in \mathbb{N}$. The set $V = (\{0, 1\} \times G) \cup \{\infty\}$ represents nodes labels, where $G$ is the additive group $\mathbb{Z}_{2t+1}$. Without loss of generality, $\infty$ lies in the longest cycle of $F$. There are 3 difference-sets, as Condition 3 of Proposition 3.5 states. Each difference $[\alpha, \beta] \in E(F)$ goes into a set depending on $\{0, 1\}$ labels of both $\alpha$ and $\beta$. We propose a two-step approach that initially determines the first labels and then the second ones. Once the first labels are determined, the problem resembles a 1-rotational problem where the decision variables are in a set of integers. On the other side, there are 3 difference-sets, as described by Equation (9).
The first-step solution provides information about the type of edge (e.g., $\alpha = (i,a), \beta = (i,b)$ | $a, b \in G$ is in the difference-set $\Delta F_{ii}$), and the second-step exploits such knowledge.

The Binary Labeling Problem (BLP) is the first-step subproblem, and asks to label each node $i \in V(F) \setminus \{\infty\}$ with a coordinate $c_i \in \{0,1\}$, namely the binary label. Once the solution of (BLP) is given, the Group Labeling Problem (GLP) seeks to assign a second coordinate $n_i \in G$, namely the group label, to each node, so that Condition 3 of Proposition 3.5 holds. Differences of the type $\Delta_{01} F$ are directed from $c_\alpha = c_x = 0$ to $c_\beta = c_y = 1$. Without loss of generality, the direction is arbitrarily fixed. Besides, $\Delta_{00} F$ and $\Delta_{11} F$ contain both the two undirected differences for each edge (e.g., both $\alpha - \beta$ and $\beta - \alpha$).

According to Condition 1 of Proposition 3.5, there are exactly $2t + 1$ nodes for each binary label. Condition 2 states that two nodes adjacent to $\infty$ have different binary labels. Condition 3 defines difference-sets’ cardinalities as $|\Delta_{00} F| = |\Delta_{11} F| = 2t$ while $|\Delta_{01} F| = 2t + 1$. (BLP) formalizes such conditions.

**Problem 4.2** (Binary Labeling Problem). Let $F = (V,E)$ be a 2-regular graph of order $|V| = 4t + 3$. (BLP) asks to assign to each node $i \in \tilde{V} = V \setminus \\{\infty\}$ a binary label $c_i \in \{0,1\}$ so that the following conditions hold:

1. The two nodes $\alpha, \beta \in \tilde{V}$ adjacent to $\infty$ have different binary labels.
   $\exists [\alpha,\infty] \land [\beta,\infty] : c_\alpha \neq c_\beta$.

2. There are exactly $2t + 1$ directed differences (edges) between nodes with different binary labels.
   $$|\Delta_{01} F| = \{\alpha - \beta \mid [(0,\alpha),(1,\beta)] \text{ for all } \alpha, \beta \in G\} = 2t + 1.$$

Equations (18-24) formulate (BLP) in CP.

\[
C = \{c_i \mid c_i \in \{0,1\}\} \quad \text{for all } i \in \tilde{V}, \quad (18)
\]
\[
dM = \{d_j \mid d_j \in \{0,1\}\} \quad \text{for all } j \in 0,1,\ldots, E(F \setminus \\{\infty\}), \quad (19)
\]
\[
d_j = \begin{cases} 1 & \text{if } c_\alpha = 0, c_\beta = 1 \\ 0 & \text{otherwise.} \end{cases} \quad \text{for all } \alpha, \beta \in \tilde{V} \land [\alpha,\beta], \quad (20)
\]
\[
\text{card}(dM|1) = 2t + 1, \quad (21)
\]
\[
\text{card}(C|1) = 2t + 1, \quad (22)
\]
\[
\text{card}(C|0) = 2t + 1, \quad (23)
\]
\[
c_\alpha = 1 \land c_\beta = 0 \quad \text{if } [\alpha,\infty] \land [\beta,\infty] \land \alpha \neq \beta. \quad (24)
\]
Example 4.3. (referring to Problem 4.2) Consider an OP(5, 6) of order 
$4t + 3 = 11$ with $t = 2$. An example of binary labels assigned according 
to the (BLP) is in Figure 5).

Each node $i \in \tilde{V}$ acquires a binary label $c_i$, hence the solution is the 
set $C$ in Equation 18. Each element $d_j \in dM$ is 1 if and only if the 
oriented edge $\lceil \alpha, \beta \rceil$ connects a node $\alpha : c_\alpha = 0$ with $\beta : c_\beta = 1$, and 0 
otherwise. Constraint (21) ensures that Condition (2) of (BLP) holds, while 
Constraints (22) and (23) bound the cardinality of binary-labeled nodes in $V$. (24) implements Condition (1) in (BLP) by hard-fixing labels of the two 
nodes adjacent to $\infty$.

Computational solutions for 2-rotational instances led us to understand the 
underlying structure of (BLP). Correspondingly, it was possible to devise a 
general polynomial-time algorithm to solve (BLP). The rationale is to search 
for known patterns and insert parts of solution (e.g.: label a subset of nodes) 
until the problem reduces to a basic pattern of the form $F[3]$, $F[5, 6]$, $F[^53]$, 
and $F[^35]$. Algorithm 2 in the Appendix (see 6.1) presents such a procedure.

Once (BLP) returns the binary labels, solving the $OP(F)$ is matter of a 
group labeling over the additive group $G$. Condition (3) of Proposition 3.5
holds on the difference-sets. In analogy with the labeling for 1-rotational methods, each group-label \( n_i \in G \mid i \in V(F) \) translates into two values in (two) difference-set, depending on its binary label. Difference-sets (9) fulfill Equations (25-27).

\[
\Delta_{00} F = \{n_\alpha - n_\beta \pmod{2^t + 1} : \text{for all } \alpha, \beta \in V(F) \land [\alpha, \beta] \land c_\alpha = c_\beta = 0 \}\tag{25}
\]

\[
\Delta_{11} F = \{n_\alpha - n_\beta \pmod{2^t + 1} : \text{for all } \alpha, \beta \in V(F) \land [\alpha, \beta] \land c_\alpha = c_\beta = 1 \}\tag{26}
\]

\[
\Delta_{01} F = \{n_\alpha - n_\beta \pmod{2^t + 1} : \text{for all } \alpha, \beta \in V(F) \land [\alpha, \beta] \land c_\alpha = 0, c_\beta = 1 \}\tag{27}
\]

Condition 2 of Proposition 3.5 infers domains on sets so that the desired 2-factor \( F \) is a 2-rotational solution for \( \text{OP}(F) \). Therefore, the Group Labeling Problem (GLP) formalizes Proposition 3.5.

**Problem 4.3 (Group Labeling Problem).** Let \( F = (V, E) \) be a 2-regular graph of order \( |V| = 4t + 3, V(F) = \{(0,1) \times G\} \cup \{\infty\} \) is the set of nodes over \( F \), where \( G \) is the additive group \( \mathbb{Z}_{2^t + 1} \). For each node \( i \in \tilde{V} = V \setminus \{\infty\} \) with its binary label \( c_i \in \{0,1\} \), the (GLP) asks to assign a group label \( n_i \in G \) so that the following conditions hold:

1. Undirected difference-sets are so that \( \Delta_{00} F = \Delta_{11} F = G \setminus \{0\} \),

2. The directed difference-set is so that \( \Delta_{01} F = G \).

Equations (28-37) formulate the (GLP) with \( CP \).

- \( V = \{A \cup B\} \), \hspace{1cm} \tag{28}
- \( A = \{a_i \mid a_i \in G, c_i = 0\} \), \hspace{1cm} \tag{29}
- \text{alldifferent}(A) \hspace{1cm} \text{dom}(A) = [0, 2^t + 1), \hspace{1cm} \tag{30}
- \text{alldifferent}(B) \hspace{1cm} \text{dom}(B) = [0, 2^t + 1), \hspace{1cm} \tag{31}
- \( dA = \{(a_\alpha - a_\beta \pmod{2^t + 1})\} \) \hspace{1cm} \text{for all } \alpha, \beta \in A \land [\alpha, \beta], \hspace{1cm} \tag{32}
- \( dB = \{(b_\alpha - b_\beta \pmod{2^t + 1})\} \) \hspace{1cm} \text{for all } \alpha, \beta \in B \land [\alpha, \beta], \hspace{1cm} \tag{33}
- \( dAB = \{(a_\alpha - b_\beta \pmod{2^t + 1})\} \) \hspace{1cm} \text{for all } \alpha \in A, \beta \in B \land [\alpha, \beta], \hspace{1cm} \tag{34}
- \text{alldifferent}(dA) \hspace{1cm} \text{dom}(dA) = [0, 2^t + 1), \hspace{1cm} \tag{35}
- \text{alldifferent}(dB) \hspace{1cm} \text{dom}(dB) = [0, 2^t + 1), \hspace{1cm} \tag{36}
- \text{alldifferent}(dAB) \hspace{1cm} \text{dom}(dAB) = [0, 2^t + 1), \hspace{1cm} \tag{37}
- \text{card}(dA) = \text{card}(dB) = 2t \hspace{1cm} \text{card}(dAB) = 2t + 1. \hspace{1cm} \tag{38}
Equation (28) represents the set $V$ as the union of $A$ and $B$, respectively the subset of nodes with binary label $c_i = 0$, and $c_i = 1$. The solution is a feasible assignment for $V$. Constraints (30)-(31) force on both $A$ and $B$ domains over $G$. Difference-sets in (32-34) rewrite sets in Equations (25-26). Finally, constraints and domains in (35-37) enforce Conditions 1 and 2 of (GLP). In particular, Constraint (38) ensures that difference-sets have the required cardinalities. The (GLP) solution generates a complete solution for $OP(F)$, with roto-translation similar to the ones explained for 1-rotational methods. Proposition 3.5 describes how $F$ generates the other $2-regular$ copies.

Example 4.4. (referring to Problem 4.3) Consider an $OP(F = [5, 6])$ of order $4t + 3 = 11$ with $t = 2$. Assuming binary labels are assigned, an example of group labels from (GLP) is represented in Figure 6.

![Figure 6](image)

According to Proposition 3.5 a solution of order $4t + 3$ generates a solution of order $4(t + 1)$. The process requires polynomial-time, and is as follows. Starting from the $4(t + 1)$ instance, a 2-path $P = [a, i^*, b]$ is selected from the cycle $C_\infty$ (the cycle containing the $\infty$ node), and replaces the edge
The resulting graph is the 2-regular $F^*$, of order $|V(F^*)| = 4t + 3$. The 2-rotational approach solves the $4t + 3$ instance on $F^*$. Afterwards, the pruned node $i^*$ goes back to $F^*$, so that $F^*$ becomes $F$. Without loss of generality, $i^*$ lies between nodes with different binary labels $i^*$ inside the cycle $C_\infty$, so that $P^* = [\alpha, i^*, \beta]$, $c_\alpha \neq c_\beta$. Node $i^*$ is relabeled as $\infty_2$ while the original $\infty$ becomes $\infty_1$. Therefore, the difference-set $\Delta_{01}F$ on $F$ loses the difference $n_\alpha - n_\beta$ (or $n_\beta - n_\alpha$ if and only if $c_\beta = 0$, $c_\alpha = 1$).

## 5. Experimental Results

We implemented the proposed algorithms and formulations with Java 1.8 (see Section 6.1 for code), IBM ILOG CPLEX and CP Optimizer 12.7. Tests ran on a Intel(R) Core i5-3550 @ 3.30GHz with 4GB of RAM, a computer. Deza et al. [15] solve instances of much smaller size (order $\leq 40$), with undisclosed algorithms running on a dedicated cluster machine [12]. The new 20 orders we solve here, from 41 to 60, are 241200, that is 22 times greater than those (10897) solved in [15]. Moreover, they generalize $r$-rotational rules also with $r \not\in \{1, 2\}$, while our contribution deals only with $r \in \{1, 2\}$.

Table 1 reports computational results for instances with $n \in [40, 60]$, and complete solutions are available online (see Section 6.1). Timelimits for (GLP) and ($F^*$ LP) are $5 \cdot (1 + |V(F)|/50)$ seconds, and $|V(F^*)|/20$ seconds, respectively, while Algorithm 2 solves (BLP) in negligible time. The 2-rotational approach (see 4.2) with odd $n$ values (see Proposition 3.5) solves instances of order $4t + 3$. Solutions of order $4t$ directly derive from the $4t + 3$ ones. The 1-rotational approach (see 4.1) solves instances of order $4t + 1$, and hence $4t + 2$. While solving orders $4t + 1$, we discovered that certain instances do not have a 1-rotational solution, and, consequently, we derived Proposition 3.4.

The formal proof stemmed after this empirical evidence. The 2-rotational approach with even $n$ values (see Proposition 3.4) solves instances not fulfilling requirements in Proposition 3.4. We generated all the order-dependent partitions of integers $i \in [40, 60]$ with at least three or more cycles (tables), since Traetta [27] provides complete solutions to the two-table $OP$. Each order (1st column of Table 1) is tackled by means of 2-rotational and/or 1-rotational rules (3rd column). The time (4th column) represents the total time required for the class $OP$, so that all the instances (5th column) have a solution (6th column). Correspondingly, the average time per instance (7th column) is the arithmetic mean. The proposed methodologies solved all the instances. Finally, from our tests, 1-rotational methods appear to be faster than 2-rotational methods in terms of CPU times, according to Table 1. Also, we
can report that single solutions for $OP$ with $n < 120$ could be generated in less than 60 seconds.

| #  | Type        | Method          | Time (s) | Partitions | Solved | Avg. Time (s.ms) |
|----|-------------|-----------------|----------|------------|--------|------------------|
| 40 | 4t          | (Derived from 39) | 911      | 1756       | 1756   | 0.519            |
| 41 | 4t+1        | 1-rotational     | 807      | 2056       | 2056   | 0.393            |
|    |             | A 2-rotational   | 90       | 1433       | 1433   | 0.063            |
|    |             |                 | 717      | 623        | 623    | 1.151            |
| 42 | 4t+2        | (Derived from 41) | 90       | 2418       | 2418   | 0.037            |
| 43 | 4t+3        | A 2-rotational   | 2462     | 2822       | 2822   | 0.872            |
| 44 | 4t          | (Derived from 43) | 2462     | 3302       | 3302   | 0.746            |
| 45 | 4t+1        | 1-rotational     | 3268     | 3851       | 3851   | 0.849            |
|    |             | A 2-rotational   | 1406     | 2547       | 2547   | 0.552            |
|    |             |                 | 1862     | 1304       | 1304   | 1.428            |
| 46 | 4t+2        | (Derived from 45) | 1406     | 4488       | 4488   | 0.313            |
| 47 | 4t+3        | A 2-rotational   | 6348     | 5215       | 5215   | 0.217            |
| 48 | 4t          | (Derived from 47) | 6348     | 6072       | 6072   | 0.045            |
| 49 | 4t+1        | 1-rotational     | 5587     | 7033       | 7033   | 0.794            |
|    |             | A 2-rotational   | 460      | 4417       | 4417   | 0.104            |
|    |             |                 | 5127     | 2616       | 2616   | 0.960            |
| 50 | 4t+2        | (Derived from 49) | 460      | 8158       | 8158   | 0.056            |
| 51 | 4t+3        | A 2-rotational   | 16705    | 9441       | 9441   | 0.769            |
| 52 | 4t          | (Derived from 51) | 16705    | 10920      | 10920  | 0.530            |
| 53 | 4t+1        | 1-rotational     | 18998    | 12600      | 12600  | 0.508            |
|    |             | A 2-rotational   | 4246     | 7513       | 7513   | 0.665            |
|    |             |                 | 14752    | 5087       | 5087   | 0.900            |
| 54 | 4t+2        | (Derived from 53) | 4246     | 14552      | 14552  | 0.292            |
| 55 | 4t+3        | A 2-rotational   | 57043    | 16753      | 16753  | 0.405            |
| 56 | 4t          | (Derived from 55) | 57043    | 19296      | 19296  | 0.956            |
| 57 | 4t+1        | 1-rotational     | 42700    | 22183      | 22183  | 0.925            |
|    |             | A 2-rotational   | 2519     | 12557      | 12557  | 0.201            |
|    |             |                 | 40181    | 9626       | 9626   | 1.174            |
| 58 | 4t+2        | (Derived from 57) | 2519     | 25491      | 25491  | 0.099            |
| 59 | 4t+3        | A 2-rotational   | 105258   | 29241      | 29241  | 0.600            |
| 60 | 4t          | (Derived from 59) | 105258   | 33552      | 33552  | 0.337            |

Table 1: Computational results for the OP with $n \in [40, 60]$, with more than 3 cycles per instance.
5.1. IP proves the absence of solution for $OP(2^3, 5)$

The Handbook of Combinatorial Design from [13] states that it is well known that $OP(2^3, 5)$ has no solution, referring to a conference proceeding of Rosa [26]. In a different work, Alspach et al. [2] cite an unpublished paper by Piotrowski [25]. In the latter, the author self-cites an unpublished paper [24] from 1979, where he describes a proof made with the aid of a computer. We provide a simple proof of non-existence for $OP(2^3, 5)$ with an IP formulation. The $OP(2^3, 5)$ is the problem of arranging $n=11$ people in 2 tables of 3 and 1 table of 5 for $M=5$ meals. Each person has a label in $\mathbb{Z}_{11}$. The IP formulation enumerates every feasible combination of labels for tables of 3 (triplets) and tables of 5 (5-sets). Afterwards, IP seeks to select for $M=5$ meals, one 5-set and two triplets so that each node is seated next to every other node exactly once during all the meals. There are $\binom{11}{3}=165$ different triplets, with at least one distinct label, and $\binom{11}{5} \cdot 12 = 5544$ 5-sets with different adjacencies. The IP formulation in (39)-(44) models $OP(2^3, 5)$.

\[
\begin{align*}
\text{min}_{F,T} & \quad 0 \quad (39) \\
\text{subject to} & \\
\sum_{i \in I} F_{id} &= 1 \quad \text{for all } d \in D, \quad (40) \\
\sum_{j \in J} T_{jd} &= 2 \quad \text{for all } d \in D, \quad (41) \\
\sum_{i \in I} F_{id} \cdot fl_{il} + \sum_{j \in J} T_{jd} \cdot tl_{jl} &= 1 \quad \text{for all } d \in D, \text{for all } l \in L, \quad (42) \\
\sum_{d \in D} \left( \sum_{i \in I} F_{id} \cdot fa_{i\alpha\beta} + \sum_{j \in J} T_{jd} \cdot ta_{j\alpha\beta} \right) &= 1 \quad \text{for all } \alpha, \beta \in L \land \alpha \neq \beta, \quad (43) \\
F_{id}, T_{jd} &\in \{0, 1\} \quad \text{for all } i \in I, j \in J, d \in D. \quad (44)
\end{align*}
\]

The binary variables $F_{id} : i \in I = [1, 5544]$ and $T_{jd} : j \in J = [1, 165]$ with $d \in D = [1, D = 5]$ respectively represent all the different 5-sets and triplets over the 5 meals, respectively. $F_{id}$ and $T_{jd}$ take value of 1 if and only if the corresponding element - the $i^{th}$ 5-set or $j^{th}$ triplet - is selected for the $d^{th}$ day. Coefficients $fl_{il}$ and $tl_{jl}$ are respectively equal to 1 if the label $l \in L = [1, 11]$ is present in the $i^{th}$ 5-set or $i^{th}$ triplet. Coefficients $fa_{i\alpha\beta}$ and $Ta_{j\alpha\beta}$ are equal to 1 if two different labels $\alpha, \beta \in L$ are adjacent in the $i^{th}$ 5-set or $i^{th}$ triplet. The model has no objective function, as of in Equation
Equations (40) and (41) enforce the selection of one 5-set and two
triplets per each \( d \in D \). Constraint (42) enforces that each label appears
only once for each \( d \in D \). Constraint (43) enforces that two labels \( \alpha, \beta \in L \)
are adjacent only one time over all the meals. The continuous relaxation
of (39)–(44) finds no solution in less than a second: hence \( OP(2^3, 5) \) has no
solution. Correspondingly, the following proposition holds.

**Proposition 5.1.** Let \( OP(2^3, 5) \) be the Oberwolfach Problem with 2 cycles
of length 3 and a cycle of length 5. There is no solution to \( OP(2^3, 5) \).

6. **Concluding remarks**

\( CP \) – particularly its propagation and inference algorithms – exploits
difference methods for the \( OP \) problem by means of well-posed formul-
ations. 1 and 2-rotational methods reduce the complete \( OP \) to the problem
of labeling a single 2-factor and that problem is efficiently solved in \( CP \).
In particular, Constraint propagation exploits the relations of mutual exclu-
sion between labels to smartly guide the search procedure. Computational
results prove the effectiveness of the approach, which provided complete so-
lutions for the \( OP \) for \( n \in [40, 60] \). Moreover, theoretical results such as
*Proposition 3.4* and the proof of absence of a solution for \( OP(2^3, 5) \) con-
stitute a further outcome of this work. The complementarity of Combinatorial
Design and Combinatorial Optimization and their positive interaction is, in
our view, the main stake pointed out here. The contribution is bidirectional:
computational evidence helps to deduce theoretical results, and the latter
provides models for the former. We believe that approaches similar to the
one presented may help to solve other problems in Combinatorial Design
Theory.

6.1. **Solutions and code**

We complement the paper with the software written to implement the
presented methodologies. To make its use accessible, we provide a GUI in-
terface.

The software is available on GitHub at the following repository:

[https://github.com/ALCO-PoliTO/TheOberSolver](https://github.com/ALCO-PoliTO/TheOberSolver)

Full solutions are available at:

[https://github.com/ALCO-PoliTO/TheOberSolver/tree/master/OberResults](https://github.com/ALCO-PoliTO/TheOberSolver/tree/master/OberResults)

The *IP DinnerFor11* formulation of \( OP(2^3, 5) \) is available at

[https://github.com/ALCO-PoliTO/DinnerFor11](https://github.com/ALCO-PoliTO/DinnerFor11)
Acknowledgements

This work has been partially supported by “Ministero dell’Istruzione, dell’ Università e della Ricerca” Award TESUN-83486178370409 finanziamento dipartimenti di eccellenza CAP. 1694 TIT. 232 ART. 6”.
Appendix

**BLP Algorithm**

The graph \( F = [n_1l_1, n_2l_2, ..., n_al_b] \) is described with \( n_i \) the number of cycles of length \( l_i \), \( T = \sum_i n_i \) the number of cycles, and \( t_M = \max_b l \) the longest cycle in \( F \). The input is an unlabeled \( F \) and the output is the BLP solution for \( F \), namely \( F_l \). The Algorithm iteratively adds to the incumbent a partial labeling for a known pattern, and terminates when all the nodes have been labeled and transferred from \( F \) to \( F_l \). Lines (3-6) iterate through \( T \) cycles, and reduce each cycle \( l_i \geq 7 \) with a cycle of length at most of 6, by labeling patterns of 4 nodes at a time. Cycles with exactly 4 nodes - as of in lines (7-9), are labeled instantly. Lines (12-18) search for more complex patterns (e.g: \( F = [3, 5] \)). Lines (19-21) label basic patterns in \( F \). The order reported in line (14) is binding, and labeled patterns have different orientations depending on the incumbent labeled nodes. The worst-case time complexity of Algorithm 2 is \( O(T \cdot t_M) \) with an implementation without Lists.

**Algorithm 2: (BLP) Algorithm**

1: **Input:** The graph \( F = [n_1l_1, n_2l_2, ..., n_al_b] = [t_1, t_2, ..., t_T] \)
2: **Output:** The labeled (colored) graph \( F_l \)
3: for \( t_i \) in \( F \) do
4: while \( t_i \geq 7 \) do
5: \( F_l \leftarrow \) last labels of \( t_i \) are [1100]; \( t_i \leftarrow (t_i - 4) \)  
\( \triangleright \) Color last four elements
6: end while
7: if \( t_i = 4 \) then
8: \( F_l \leftarrow \) labels of \( t_i \) are [1100]; \( t_i \leftarrow (t_i - 4) \)  
\( \triangleright \) Color last four elements
9: end if
10: end for
11: \( \text{found} = \text{true} \)
12: while \( \text{found} \) do  
\( \triangleright \) Search for known patterns
13: \( \text{found} = \text{false} \)  
\( \triangleright \) The order of search is strictly as specified
14: if \( F \) contains patterns like \( F[3, 5], F[4, 5], F[3, 3], F[3, 6], F[2, 3, 6] \) then
15: \( \text{found} = \text{true} \)
16: \( F_l \leftarrow \) labels of \( t \) are [Pattern]; \( t_i \leftarrow (t_i - \text{Pattern.length}) \)
17: end if
18: end while
19: if the remaining problem is of the form of \( F[3, 5], F[5, 6], F[3, 3], F[3, 5] \) then
20: \( F_l \leftarrow \) labels of \( t \) are [Pattern]; \( t_i \leftarrow (t_i - \text{Pattern.length}) \)
21: end if  
\( \triangleright \) Check for basic solutions. The order of search is as specified
22: return \( F_l \)
(Almost) 2-rotational with $n$ even

The approach to this class of instances is similar to the one presented for odd $n$. The 2-rotational method with even $n$ solves instance not fulfilling Proposition 3.3. The methodology is analogous to the one for odd $n$, but Condition from Proposition 3.5 is discarded, while Condition 4 holds. If $n$ is even, $F$ has order of $4t + 1$ with $n = 2t$. The set of vertices is $V = (\{(0,1)\} \times G) \cup \{\infty\}$, with $G$ the additive group $\mathbb{Z}_{2t}$. (BLP) and (GLP) slightly vary, according to the Proof of Proposition 3.5. In particular, according to Condition 4 of Proposition 3.5, a cycle of $F$ contains a path of the form $P = [\langle 0,0 \rangle, \langle 0,n/2 \rangle, \langle 1,n/2 \rangle, \langle 1,0 \rangle]$. The modified(BLP) takes into account the path $P$ so that the first two nodes of $P$ take the label 0, and the last two the label 1. We call critical paths all the candidates paths in $F$. Difference-sets, represented Equations (25-27), are considered over the graph $F - P$, with modulo $4t$, and their cardinality is lowered to $2t - 2$. For ease of notation, the introduced new sub-problems are the Even Binary Labeling Problem (eBLP) and the Even Group Labeling Problem (eGLP).

**Problem 6.1** (Even Binary Labeling Problem). Let $F = (V,E)$ be a 2-regular graph of order $|V| = 4t + 1$. The (eBLP) asks to assign to each node $i \in \bar{V} = V \backslash \{\infty\}$ a binary label $c_i \in \{0,1\}$ so that the following conditions hold:

1. $[\alpha, \infty] \wedge [\beta, \infty] \implies c_\alpha \neq c_\beta$,

2. There is at least one critical path $P$ so that $P = [x,y,z,k] \mid x,y,z,k \in \bar{V} \wedge c_x = c_y = 0, c_z = c_k = 1$,

3. $|\Delta(F - P)_{01}| = \{|\alpha - \beta : [(0,\alpha),(1,\beta)] \forall \alpha,\beta \in G\}| = 2t - 2$.

The CP model (18-24) is modified to fit the additional Condition (2) for the (eBLP). Constraints (21-23) are modified to enforce the new cardinality $(2t - 1)$ for both $dM$ and $C$. Moreover, the following additional Constraints hold:

$$A = \{A_i \mid A_i \in \{0,1\}\} \quad \text{for all } i \in PA,$$

$$A_i = \begin{cases} 1 & \text{if } c_x = c_y = 0 \land c_z = c_k = 1 \\ 0 & \text{otherwise.} \end{cases}$$

$$PA_i = [x,y,z,k],$$

$$\text{card}(A_i|1) \geq 1.$$
if \( c_x = c_y = 0 \) and \( c_z = c_k = 1 \), and hence \( A_i \) is a critical path. At least one critical path exists as of (47). Once (eBLP) is solved, the (eGLP) labels are determined considering a single critical path \( A_i \). If no solution is found for the latter sub-problem, a new critical path \( A_j \neq A_i \) induces a different (eGLP). In terms of (eGLP), Condition 4a induces four specific group-labels on the critical path \( A_i \).

**Problem 6.2 (Even Group Labeling Problem).** Let \( F = (V, E) \) be a 2-regular graph of order \( |V(F)| = 4t + 1 \). \( V(F) = \{0, 1\} \times G \cup \{\infty\} \) is the set of nodes over \( F \), where \( G \) is the additive group \( \mathbb{Z}_{2t} \). For each node \( i \in \bar{V} = V \setminus \{\infty\} \) - given the binary label \( c_i \in \{0, 1\} \) of \( V \) and a critical path \( P = [x,y,z,k] = [(0,0),(0,t),(1,t),(1,0)] \).

The (eGLP) \( CP \) formulation is similar to the one in (28-37), and the critical-path \( P = A_i \) constitutes an additional input.

\[
\begin{aligned}
V &= \{A \cup B\}, \quad (48) \\
A &= \{a_i \mid a_i \in G, c_i = 0\} \quad \text{dom}(A) = [0, 2t], \quad (49) \\
alldifferent(A) &\quad \text{dom}(B) = [0, 2t], \quad (50) \\
dA &= \{(a_\alpha - a_\beta \mod (2t))\} \quad \text{for all } \alpha, \beta \in A, [\alpha, \beta] \notin P, \quad (52) \\
dB &= \{(b_\alpha - b_\beta \mod (2t))\} \quad \text{for all } \alpha, \beta \in B, [\alpha, \beta] \notin P, \quad (53) \\
dAB &= \{(a_\alpha - b_\beta \mod (2t))\} \quad \text{for all } \alpha \in A, \beta \in B, [\alpha, \beta] \notin P, \quad (54) \\
alldifferent(dA) &\quad \text{dom}(dA) = (0, 2t)\setminus\{t\}, \quad (55) \\
alldifferent(dB) &\quad \text{dom}(dB) = (0, 2t)\setminus\{t\}, \quad (56) \\
alldifferent(dAB) &\quad \text{dom}(dAB) = (0, 2t)\setminus\{t\}, \quad (57) \\
n_x = n_k = 0, n_y = n_z = t \quad P = [x,y,z,k], \quad (58) \\
\text{card}(dA) = \text{card}(dB) = \text{card}(dAB) \quad \text{card}(dAB) = 2t - 2. \quad (59)
\end{aligned}
\]

Constraints (49, 54 and 59) have different cardinalities and modulo arguments in. The value \( n/2 = t \) is not in difference-sets of Constraints (52-54). Moreover, edges inside \( P \) do not figure in difference-sets. Condition 2 of constraints...
(eGLP) assigns labels for nodes inside the critical path $P$, as in Constraint [58].

References

[1] B. Alspach and R. Häggkvist. Some observations on the oberwolfach problem. *Journal of Graph Theory*, 1985. ISSN 10970118. doi: 10.1002/jgt.3190090114.

[2] B. Alspach, P. J. Schellenberg, D. R. Stinson, and D. Wagner. The oberwolfach problem and factors of uniform odd length cycles. *Journal of Combinatorial Theory, Series A*, 52(1):20–43, 1989.

[3] R. R. Anstice. On a problem in combinations. *Cambridge and Dublin Math. J.*, 7:279–292, 1852.

[4] N. Beldiceanu, M. Carlsson, and J.-X. Rampon. Global constraint catalog, 2010.

[5] D. Bryant and P. Danziger. On bipartite 2-factorizations of $K_n - I$ and the Oberwolfach problem. *Journal of Graph Theory*, 2011. ISSN 03649024. doi: 10.1002/jgt.20538.

[6] D. Bryant and V. Scharaschkin. Complete solutions to the Oberwolfach problem for an infinite set of orders. *Journal of Combinatorial Theory. Series B*, 2009. ISSN 00958956. doi: 10.1016/j.jctb.2009.03.003.

[7] M. Buratti and G. Rinaldi. On sharply vertex transitive 2-factorizations of the complete graph. *Journal of Combinatorial Theory. Series A*, 2005. ISSN 00973165. doi: 10.1016/j.jcta.2004.11.014.

[8] M. Buratti and G. Rinaldi. 1-Rotational $k$-factorizations of the complete graph and new solutions to the Oberwolfach problem. *Journal of Combinatorial Designs*, 2008. ISSN 10638539. doi: 10.1002/jcd.20163.

[9] M. Buratti and G. Rinaldi. A non-existence result on cyclic cycle-decompositions of the cocktail party graph. *Discrete Mathematics*, 2009. ISSN 0012365X. doi: doi:10.1016/j.disc.2008.05.042.

[10] M. Buratti and T. Traetta. 2-Starters, Graceful Labelings, and a Doubling Construction for the Oberwolfach Problem. *Journal of Combinatorial Designs*, 2012. ISSN 10638539. doi: 10.1002/jcd.21296.
[11] M. Buratti and T. Traetta. The structure of 2-pyramidal 2-factorizations. *Graphs and Combinatorics*, 31(3):523–535, May 2015. ISSN 1435-5914. doi: 10.1007/s00373-014-1408-2. URL https://doi.org/10.1007/s00373-014-1408-2

[12] C. Canada. Sharcnet, 2019. URL http://www.sharcnet.ca

[13] C. J. Colbourn and J. H. Dinitz. The CRC handbook of combinatorial designs. *CRC Press series on discrete mathematics and its applications.*, 1996.

[14] S. Costa. A complete solution to the infinite oberwolfach problem. *Journal of Combinatorial Designs*, 28(5):366–383, 2020. doi: 10.1002/jcd.21702. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/jcd.21702

[15] A. Deza, F. Franek, W. Hua, M. Meszka, and A. Rosa. Solutions to the Oberwolfach problem for orders 18 to 40. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 2010. ISSN 08353026.

[16] S. García, V. Cacchiani, L. Vanhaverbeke, and M. Bischoff. The table placement problem: a research challenge at the ewi 2007. *TOP*, 22(1):208–226, Apr 2014. ISSN 1863-8279. doi: 10.1007/s11750-012-0249-5. URL https://doi.org/10.1007/s11750-012-0249-5

[17] S. Glock, F. Joos, J. Kim, D. Kühn, and D. Osthus. Resolution of the oberwolfach problem. *arXiv preprint arXiv:1806.04644*, 2018.

[18] R. K. Guy. Unsolved combinatorial problems. In *Combinatorial Mathematics and its Applications*, pages 121–127. Academic Press, 1971.

[19] A. Hilton and M. Johnson. Some results on the oberwolfach problem. *Journal of the London Mathematical Society*, 64(3):513–522, 2001.

[20] D. G. Hoffman and P. J. Schellenberg. The existence of $C_k$-factorizations of $K_{2n}−F$. *Discrete Mathematics*, 1991. ISSN 0012365X. doi: 10.1016/0012-365X(91)90440-D.

[21] C. Huang, A. Kotzig, and A. Rosa. Variation of the Oberwolfach Problem. *Discrete Mathematics*, 27(3):261–277, 1979. ISSN 0012-365X.

[22] M. Ollis and A. D. Sterr. From graceful labellings of paths to cyclic solutions of the oberwolfach problem. *Discrete Mathematics*, 309(14):4877–4882, 2009.
[23] J. Petersen. Die theorie der regulären graphen. *Acta Math*, 15:193–220, 1891.

[24] W. L. Piotrowski. Untersuchungen über das oberwolfacher problem. unpublished, 1979.

[25] W. L. Piotrowski. The solution of the bipartite analogue of the Oberwolfach problem. *Discrete Mathematics*, 1991. ISSN 0012365X. doi: 10.1016/0012-365X(91)90449-C.

[26] A. Rosa. Two-factorizations of the complete graph. *Rend. Sem. Mat. Messina II*, 9:201–210, 2003.

[27] T. Traetta. A complete solution to the two-table Oberwolfach problems. *Journal of Combinatorial Theory, Series A*, 120(5): 984–997, 2013. ISSN 0097-3165. doi: https://doi.org/10.1016/j.jcta.2013.01.003. URL http://www.sciencedirect.com/science/article/pii/S0097316513000149