Measures of Balance in Combinatorial Optimization

Philippe Olivier[1,2], Andrea Lodi[1,2], and Gilles Pesant[1]

Polytechnique Montréal, Montreal, Canada[1]
Canada Excellence Research Chair[2]
{philippe.olivier, andrea.lodi, gilles.pesant}@polymtl.ca

Abstract. The concept of balance plays an important role in many combinatorial optimization problems. Yet there exist various ways of expressing balance, and it is not always obvious how best to achieve it. In this methodology-focused paper, we study three cases where its integration is deficient and analyze the causes of these inadequacies. We examine the characteristics and performance of the measures of balance used in these cases, and provide general guidelines regarding the choice of a measure.

1 Introduction

It is natural to think of *balance* as *things being as equal as possible*. Yet this notion of equality is hard to define. Suppose that we have candy bags of assorted sizes (5, 5, 6, 7, 9, 12, and 12 candies) which we want to distribute fairly to four children. We easily observe that a perfect distribution of 14 candies per children is not possible. What, then, constitutes a fair distribution?

We could consider as our criterion of fairness that the largest share of candies be as small as possible, which would give us handouts of 12, 12, 16, and 16 candies. Or we could instead consider an alternative criterion and ensure that the sum of candy discrepancies from the mean is minimal, giving us handouts of 12, 13, 14, and 17 candies. We can notice that the optimal solution for one criterion of fairness is not optimal for the other, and vice-versa. These options are both *fair*, yet neither is intrinsically better or worse than the other.

This simple example illustrates how the notion of balance becomes more ambiguous after scratching the surface. Besides, this notion deserves special attention, as fairness is of paramount importance in several practical situations. In many jurisdictions of the United States, for instance, algorithms have taken the role of decision-makers for delicate matters such as deciding whether a defendant awaiting trial should be released or not. One study found that African-Americans were one and a half times more likely than Caucasians to be wrongly classified as high risk by one of these algorithms [1]. Racial disparities being a sensitive issue, these algorithms need to ensure fairness in this process [2].

While this paper is concerned with balance in the context of combinatorial optimization, issues of fairness also arise in the related field of game theory, where

some criteria of fairness are of a different nature. *Envy-freeness* ensures that no player would want to trade their share for that of another, *Pareto efficiency* guarantees that no share can be improved without worsening some other share, and so on [3].

A tangential application combining balance types and fairness criteria is found in *social welfare functions*. These functions describe the collective welfare of a society based on the utilities, or satisfaction, of its individuals [4]. The utilitarian function measures collective welfare as the sum of all individual utilities, maximizing pure utility while disregarding any type of equality between individuals. In contrast, the egalitarian function considers the minimum of all individual utilities at the expense of a lower level of global welfare, putting everyone on an equal (albeit usually lower) footing [5]. The Nash social welfare function maximizes the product of all utilities, which provides a sort of middle ground between the two previous functions [6].

In this methodology-focused paper, we present three cases supporting the hypothesis that unfamiliarity with respect to the characteristics of measures of balance often leads to poor choices regarding modeling in combinatorial problems. We examine the characteristics of several measures, and provide general guidelines regarding the choice of an appropriate measure. We do this by keeping in mind two solving paradigms for combinatorial optimization, namely *constraint programming* (CP) and *integer programming* (IP), so as to gain, in addition, some understanding of the pros and cons of the analyzed measures with respect to the solution methods.

To start with, Section 1.1 briefly introduces CP and IP. Section 2 presents a few measures of balance and their characteristics, and outlines their use in the context of CP and IP. Three problematic cases are studied in Section 3. Finally, Section 4 discusses general guidelines for the application of the various measures of balance.

1.1 CP and IP Considerations

Constraint Programming is a programming paradigm for solving combinatorial problems. It is a form of declarative programming wherein an abstract model is constructed and then handled by a solver. This model is defined by the problem variables, their domains, and a set of constraints defining the relation between subsets of variables, restricting certain variable-value combinations. A solution to this model is a tuple of values consistent with the domains of the variables, and which satisfies the constraints.

There are several types of constraints: the `alldifferent` constraint, for example, ensures that a subset of variables are each assigned distinct values. Each constraint achieves its purpose through the action of filtering—the process of removing values inconsistent with the constraint from the domains of variables. While filtering reduces the domains of the variables, it is not enough to solve the problem by itself. Hence, a search tree is constructed, and is explored by branching on variables at each node, i.e., instantiating a variable with a specific value

in its domain. Proving optimality in CP amounts to dynamically adding constraints during search, such that for a solution to be feasible it must be strictly better than the best one found so far. The reader is refereed to [7] for a detailed methodological overview on CP.

The Integer (Linear) Programming paradigm is based on modeling by means of a linear objective function to be optimized over a set of linear constraints. The constraints are defined upon decision variables, a subset of which is restricted to take only discrete (integer) values. This latter requirement makes the problem hard from a complexity standpoint because the resulting feasible space is non-convex. For this reason, IP technology is based on the simple idea of relaxing the integrality requirements on the variables and iteratively solving the so-called continuous relaxation, which, on the contrary, is easy, i.e., it is solvable in polynomial time. The value of the optimal solution of the continuous relaxation provides a dual bound on the optimal solution of the original IP. Several algorithmic techniques, like preprocessing and cutting planes, enhance the quality of the continuous relaxation with respect to the convex hull of (mixed-)integer solutions and are fundamental building blocks of the IP technology. Conversely, the basic solution scheme requires to enforce integrality by means of a divide-and-conquer algorithm called branch and bound. Finally, feasible solutions for the overall problem are computed throughout the process by primal heuristics, thus providing a primal bound and making the overall algorithm converge by closing the gap between primal and dual bounds. The reader is referred to [8] for a detailed methodological overview on IP.

In this paper, we are concerned with the quality of the solutions rather than the difficulty of finding them, which is partly influenced by the method used to solve the problem. CP and IP are two approaches that are distinct in nature. As such, the different ways of computing balance have a varying influence on these methods. A CP solver does not assume any particular property of the solution space (such as convexity), and the various measures of balance generally have a similar impact on the performance of the model. IP, on the other hand, has a strong preference for linear modeling, as a linear model can generally be solved much more efficiently than a nonlinear one.[1] The difficulty in comparing CP and IP is further exacerbated by other factors, such as the type of problem being solved, or one's modeling choices. Symmetry breaking constraints, for example, tend to have a major impact on performance; yet their behavior may vary wildly in CP and IP due to the different nature of these models. For these reasons, we will not be directly comparing the performance of CP and IP on the problems presented in this paper although some hints on solution aspects will be gained indirectly.[2]

[1] Balancing with L_2-DEVIATION rather than L_1-DEVIATION turns an IP model into a (Mixed-)Integer Nonlinear Programming one (MINLP). Although impressive advances in solving MINLPs have been made in the last 15 years [9], MINLPs are still, in general, significantly more difficult to solve than ILPs.

[2] In this paper, the models presented in Appendices B, C, and D were solved via CP.

2 Measures of Balance

There is little uniformity with respect to the labels associated with all things balance in the literature. Marsh and Schilling use the term *equity*, which they equate to *fairness* [10]. They describe their objective of maximizing equity through the process of minimizing *inequities*. Equity is also considered equivalent to fairness by Ogryczak [11], but maximizing equity is achieved by minimizing *inequalities*. This latter author further refers to the former authors' *inequities* as *inequalities*. In neither case is *balance* mentioned. Pesant states in [12] that "in rostering we usually talk of fairness instead of balance, because of the human factor." However, it is not uncommon to find the term *balance* in such contexts [13]. In the present paper, we have opted to use *balance* as an umbrella term encompassing everything that should be fair and, in general, *as equal as possible*.

Given a finite collection of real variables $X = \{x_1, x_2, \ldots, x_n\}$, its *balance* has been defined in several ways in the literature. Some measures only take into account the extremal values, the rationale being that constraining the most extreme points forces the others into a shorter interval. Other measures consider all values, and while usually more computationally expensive this often results in an improved distribution (relative to the aims of the problem). This section covers four common measures of balance.

The MINMAX measure is rather crude and simply minimizes the maximum value

$$\min \max_{i=1}^{n} x_i.$$

For a collection of n points, the global distance between these points and their arithmetic mean μ, according to a norm p, is defined by the concept of L_p-deviation

$$\sum_{i=1}^{n} |x_i - \mu|^p.$$

In particular, L_1-DEVIATION minimizes the sum of absolute deviations from the mean

$$\min \sum_{i=1}^{n} |x_i - \mu|,$$

L_2-DEVIATION minimizes the sum of squared deviations from the mean

$$\min \sum_{i=1}^{n} (x_i - \mu)^2,$$

and L_∞-DEVIATION minimizes the maximum deviation from the mean

$$\min \max_{i=1}^{n} |x_i - \mu|.$$

By using definitions that are standard in statistics [14], we now introduce some characteristics exhibited by these measures of balance. Namely,

– The *dispersion* represents the size of the interval within which the points are located, and by extension is one measure of the sensitivity to outliers.[3]
– A distribution is *smooth* when its points appear evenly within this interval.
– The number of *outliers* near the edges of the dispersion interval is another measure of the sensitivity to outliers.

The above intuitive concepts are defined and discussed formally in Appendix A. In the remainder of the paper, they are used to characterize the performance of the four balance measures in the three case studies that we present.

Generally, when optimized with MINMAX, values are only bounded on one side, and as such they show the largest dispersion. The other measures force bounds on both sides, with L_2- and L_∞-DEVIATION forcing especially tight bounds by nature. MINMAX offers little smoothness as many values will tend to be grouped around the bound. L_∞-DEVIATION is in contrast smoother—nothing is constraining the deviations apart from forcing them to be within the interval, so their associated values will appear somewhat randomly within this interval. Results are more varied for L_1- and L_2-DEVIATION, since there is a natural bias for values to be closer to the mean. The lack of minimum bound for MINMAX makes it robust against outliers, as does the linear expression of deviation for L_1-DEVIATION. Outliers have more influence on L_∞-DEVIATION since both small and large values disproportionately affect the objective, and are also more impactful on the quadratic expression of deviation of L_2-DEVIATION.

Marsh and Schilling [10] have surveyed measures of equity, in particular related to facility location. The authors record and briefly analyze some 20 measures of balance in use in various fields. They propose some guidelines to help in choosing a measure.

Balancing in constraint programming is usually achieved through special constraints. L_1-DEVIATION is handled by the `deviation` constraint, introduced by Schaus et al. [15]. Pesant and Régin [16] balanced with L_2-DEVIATION using the `spread` constraint. The `dispersion` constraint proposed by Pesant [12] encapsulates multiple measures of balance, including L_1-, L_2-, and L_∞-DEVIATION. Other measures, such as MINMAX, can be expressed with classical constraints such as `minimum` and `maximum`.

There does not seem to be any substantial work on general balancing techniques in the context of integer programming, outside of problem-specific cases. This is partly due to the fact that balancing in IP cannot be decoupled from a model as in CP, where balancing is tightly wrapped in a single constraint that can be generically reused. Early work on balancing in mathematical programming includes a short paper by Gaudioso and Legato [17] presenting a few balancing measures, among which MINMAX. Some examples include a mixed-integer linear program with L_1-DEVIATION as its objective that has been used to balance the loads on servers [18], and a quadratic integer program balancing completion times of jobs using L_2-DEVIATION [19]. A recent paper by Olivier et

[3] In this paper, we call an outlier any value equal to one of the two extremal values of the dispersion interval.

al. [20] covers the L_1-, L_2-, and L_∞-DEVIATION measures in the context of IP, and compares these with equivalent CP approaches, with special attention to quadratic versions of knapsack and bin packing problems.

3 Case Studies

This section presents practical problems that require some form of balancing: the assignment of courses to periods such that the loads of the periods are balanced, the assignment of patients to nurses such that the workloads of the nurses are balanced, and the distribution of bikes to stations in a bike sharing system such that the stations are balanced. We argue that when these problems were initially introduced, their measures of balance were deficient; we will show how they have been improved.

3.1 Balanced Academic Curriculum Problem

The *Balanced Academic Curriculum Problem* (BACP) attempts to find an assignment of courses over a number of periods such that the academic load of a student is balanced throughout the curriculum and that course prerequisite constraints are respected. Let[4]

- $\mathcal{P} = \{1, \ldots, m\}$ be the index set of periods,
- w denote the combined loads of all the courses,
- $L = \{L_1, \ldots, L_m\}$ denote the loads of the periods for an assignment.

The objective is to maximize the balance of an assignment of the n courses to the m periods. The BACP was originally introduced by Castro and Manzano [21], whose CP and IP models both achieved balance by minimizing the maximum academic load of the periods (MINMAX). Further papers by Hnich et al. [22, 23] introduced new CP and IP models using the same balancing criterion. Monette et al. [24] not only used MINMAX but also explored other options, namely balancing using the L_1-, L_2-, and L_∞-DEVIATION measures, all with a CP model. The four objectives, in minimization form, studied by Monette et al. can be formalized as

$$\max_{k \in \mathcal{P}} L_k \qquad \text{(MINMAX)}$$

$$\sum_{k \in \mathcal{P}} |L_k - w/m| \qquad (L_1\text{-DEVIATION})$$

$$\sum_{k \in \mathcal{P}} (L_k - w/m)^2 \qquad (L_2\text{-DEVIATION})$$

$$\max_{k \in \mathcal{P}} |L_k - w/m| \,. \qquad (L_\infty\text{-DEVIATION})$$

[4] A complete model for the BACP can be found in Appendix B.

Starting with the premise that "neither criterion subsumes the others and there is no *a priori* reason to prefer one of them" [24], Monette et al. aim to determine how well each balance criterion approximates the others. Their findings[5] are reproduced in Table 1, where rows represent optimized criteria and columns represent evaluated criteria. For example, at the intersection of row "L_1-DEVIATION" and column "MINMAX" is the value 2.63. This means that if the problem is optimized with respect to L_1-DEVIATION, and that we then evaluate MINMAX on that solution, it is on average 2.63% higher than if the problem was optimized with respect to MINMAX. In other words, optimizing a problem with respect to L_1-DEVIATION is a decent approximation of MINMAX, as that solution is on average only 2.63% worse than optimizing directly with MINMAX. The average of a row represents how well the balance criterion approximates the others in general, while the average of a column represents how well the balance criterion is approximated by the others in general.

Table 1. Comparison of the balance criteria for the BACP (reproduced from [24]).

	MINMAX	L_1-DEV.	L_2-DEV.	L_∞-DEV.	Average
MINMAX	0.00	10.62	16.53	0.06	9.07
L_1-DEVIATION	2.63	0.00	6.27	0.12	3.00
L_2-DEVIATION	0.28	0.00	0.00	0.00	0.09
L_∞-DEVIATION	10.37	18.07	23.66	0.00	17.36
Average	4.43	9.56	15.48	0.06	

Monette et al. observed that the optimal solutions of L_2-DEVIATION were often also optimal for the other measures, and thus that this measure was generally a good approximation of the others. For this reason, the authors conclude that L_2-DEVIATION is the superior measure of balance for the BACP. Further publications by various authors on this problem and its variants also use L_2-DEVIATION (see for example [12, 25, 26]). We have conducted similar experiments[6] as Monette et al., and reached comparable conclusions. Details of our findings can be found in Table 2.

Takeaway The dispersion interval is more than twice as large for MINMAX than it is for the other measures. Since MINMAX does not use the mean as a point of reference in balancing, low values have no impact on the solutions, yet they may significantly increase the dispersion interval. No measure of balance offers solutions with a smooth distribution of values—this is understandable, as no measure for the BACP is better satisfied by spreading values evenly in the dispersion interval. MINMAX shows many more outliers than the other measures since, for this measure, the values tend to be grouped around the upper bound

[5] Guidelines to generate equivalent instances to those used can be found in [24].
[6] Our dataset can be found at https://github.com/PhilippeOlivier/mobico.

Table 2. Characteristics of the solutions of the BACP. The first column displays the normalized size of the dispersion intervals. In the second column, smoothness is described as the Wasserstein distance (see Appendix A for details) between the distribution of the solutions and a perfectly smooth distribution (lower is smoother). The last column shows the average percentage of values that are outliers (an outlier is characterized as being any value equal to the lower or upper bound of the dispersion interval).

	Dispersion	Smoothness	Outliers
MINMAX	2.25	1.69	59.40%
L_1-DEVIATION	1.01	1.82	14.90%
L_2-DEVIATION	1.00	1.77	17.10%
L_∞-DEVIATION	1.07	1.56	18.50%

of the dispersion interval. As did Monette et al., we conclude that for the BACP, the L_2-DEVIATION performs very well but, considering in close detail Table 2, we also notice that the L_1-DEVIATION is an excellent balance choice. They both offer short dispersion intervals, fairly few outliers, and provide a good appoximation of the other measures.

3.2 Nurse-Patient Assignment Problem

The *Nurse-Patient Assignment Problem* (NPAP) seeks to assign patients to nurses within different zones in a hospital. The patients have various acuities, and should be assigned so as to best balance the workload among the nurses. The workload of a nurse is defined by the sum of their patients' acuities. Let[7]

- $\mathcal{N} = \{1, \ldots, n\}$ be the index set of nurses,
- $\mathcal{P} = \{1, \ldots, m\}$ be the index set of patients,
- a denote the combined acuities of all the patients,
- w_j be the workload of nurse j.

The patients are located in different zones, and as such the NPAP is twofold: Nurses must first be assigned to zones, and then patients to nurses. The objective is to find a staffing of nurses to zones combined with a nurse-patient assignment maximizing the balance of the nurses' workloads. The objectives, in minimization form, can be formalized similarly as for the BACP

[7] A complete model for the NPAP can be found in Appendix C.

$$\max_{j \in \mathcal{N}} w_j \qquad \text{(MINMAX)}$$

$$\sum_{j \in \mathcal{N}} |w_j - a/m| \qquad (L_1\text{-DEVIATION})$$

$$\sum_{j \in \mathcal{N}} (w_j - a/m)^2 \qquad (L_2\text{-DEVIATION})$$

$$\max_{j \in \mathcal{N}} |w_j - a/m| \,. \qquad (L_\infty\text{-DEVIATION})$$

The NPAP was introduced by Mullinax and Lawley [13], whose IP model expressed the measure of imbalance for a zone as the difference between its nurses' lightest and heaviest workloads. The objective was then to minimize the sum of imbalances for all the zones. Schaus et al. [27] have shown that while the previous model may do a good job in balancing the workloads within each zone, its objective function is deficient as this does not necessarily translate into a good balance of workloads between the different zones. The authors constructed CP models to solve this problem, and considered the L_1- and L_2-DEVIATION measures to minimize either the absolute or squared deviations of the workloads. They conclude that L_2-DEVIATION is more appropriate for the NPAP due to its increased sensitivity to outliers.

We have conducted similar experiments[8] on the NPAP as Monette et al. did for the BACP [24] by adapting the CP model of [28] with the four objectives. Our results are reported in Table 3.

Table 3. Comparison of measures of balance for the NPAP.

	MINMAX	L_1-DEV.	L_2-DEV.	L_∞-DEV.	Average
MINMAX	0.00	0.61	7.45	22.81	7.72
L_1-DEVIATION	0.19	0.00	3.79	18.07	5.51
L_2-DEVIATION	0.42	0.87	0.00	1.30	0.65
L_∞-DEVIATION	0.35	0.94	0.29	0.00	0.40
Average	0.24	0.60	2.88	10.55	

The two problems share some similarities but are nevertheless unique in their own way. The BACP imposes assignment restrictions in the form of course prerequisites, while in the NPAP these restrictions are embedded in the staffing problem. Both problems have similar assignment ratios (five courses per period and six patients per nurse, on average), but the range of patient acuities in the NPAP is much wider than the range of course credits in the BACP. L_∞-DEVIATION is the best approximator for the NPAP and the worst for the BACP, indicating that it is sensitive to the problem type. In contrast, L_2-DEVIATION

[8] Our dataset can be found at https://github.com/PhilippeOlivier/mobico.

is a very good approximator for both problems, suggesting robustness against various types of problems. As a general rule, MINMAX itself is not a very good approximator, but it can be well-approximated by the other measures. The opposite is true for L_2-DEVIATION, which is usually a good approximator for other measures of balance but which does not tend to be approximated very well most of the times.

Table 4 reports quantitative measures of dispersion, smoothness and outliers (as in Table 2 for BACP).

Table 4. Characteristics of the solutions of the NPAP.

	Dispersion	Smoothness	Outliers
MINMAX	1.11	1.03	19.27%
L_1-DEVIATION	1.09	1.04	18.81%
L_2-DEVIATION	1.01	1.06	19.60%
L_∞-DEVIATION	1.00	1.03	19.86%

Takeaway As shown in Table 4, the dispersion intervals of MINMAX and L_1-DEVIATION are, on average, around 10% larger than those of L_2- and L_∞-DEVIATION. A short dispersion interval is always preferable for the NPAP, as it ensures that the workloads are never too far apart from each other even when there is a lot of variation between them. While almost a fifth of the workloads are outliers with all measures of balance, they have a limited impact on L_2- and L_∞-DEVIATION due to their shorter dispersion intervals. As for the BACP, we observe that none of the measures offer smooth solutions, for reasons similar to those of the previous problem. We conclude that for the NPAP, the L_2- and L_∞-DEVIATION measures are preferrable, due to their short dispersion intervals and tight approximation of the other measures.

3.3 Balancing Bike Sharing Systems

A bike sharing system is a service that allows users to pick up and return bikes from and to bike stations around a city. This system is prone to imbalance since a station may not see the same number of bikes picked up and returned. To mitigate this issue, a fleet of vehicles rebalances the stations by redistributing the bikes more evenly between them. The *Balancing Bike Sharing Systems* (BBSS) problem aims to find optimal tours of the fleet of vehicles to rebalance bike stations.

While many variants of the BBSS problem can be found in the literature, a popular definition involves the minimization of a weighted combination of bike deviations from a target (using L_1-DEVIATION) and some routing cost combining the time required for loading/unloading and driving. This definition, or something very close to it, can be found in [29–33]. Several other definitions of the

problem also make use of L_1-DEVIATION (see, e.g., [34, 35]). The static version of the BBSS problem, as presented here, assumes that the impact of customers using the service is negligible, as it would be if rebalancing was done during the night. For the purpose of this paper, we will limit ourselves to the static version of the BBSS problem, although a dynamic version of this problem has also been studied [34].

In this section, we solve the BBSS problem variant mostly as it is presented in the previous paragraph. To facilitate the analysis of the results, we move the routing cost of the objective into the constraints to isolate the balancing factor from the rest. Let[9]

- $\mathcal{S} = \{1, \ldots, S\}$ be the set of S stations,
- for a station $s \in \mathcal{S}$, b_s be its initial number of bikes, t_s its target number of bikes, and $service_s$ the number of bikes that have been added or removed from it.

Several vehicles originating from their depots move bikes between stations; These capacitated vehicles are constrained by the duration of their routes. We consider the following alternatives to express our minimization balancing objective :[10]

$$\max_{s \in \mathcal{S}} (b_s + service_s - t_s) \qquad \text{(MINMAX)}$$

$$\sum_{s \in \mathcal{S}} |b_s + service_s - t_s| \qquad (L_1\text{-DEVIATION})$$

$$\sum_{s \in \mathcal{S}} (b_s + service_s - t_s)^2 \qquad (L_2\text{-DEVIATION})$$

$$\max_{s \in \mathcal{S}} |b_s + service_s - t_s| \,. \qquad (L_\infty\text{-DEVIATION})$$

We posit that in the case of the BBSS problem, L_2-DEVIATION is preferable to L_1-DEVIATION. From a practical standpoint, a station is fully functional if it can both provide and accept bikes. As such, it is more desirable to have all stations slightly unbalanced rather than to have most stations very balanced with a few very unbalanced ones. The system as a whole provides a better service in the former case than in the latter, as it boasts more functional stations. This particular understanding of system functionality is found elsewhere in the literature. The objective function of [36], which in part minimizes the number of events associated with a lack or an excess of bikes, suggests that these authors endorse this position. The constraining of station inventory to be within a serviceable interval, as found in [37], indicates that these authors also share a similar opinion. The scenario presented in Example 1 illustrates how L_2-DEVIATION has an increased tendency to prevent substantial imbalances when compared to L_1-DEVIATION.

[9] A complete model for the BBSS problem can be found in Appendix D.

[10] In contrast with the BACP and the NPAP, in the following objectives the mean is implicitly taken into account, since it is always 0.

Example 1. In Fig. 1, the vehicle must depart from and arrive to depot D with a load of 0. With a vehicle capacity of 1 and a maximum duration of 4, the two feasible solutions are D–S_1–S_2–D ($service_1 = -1$ and $service_2 = 1$) and D–S_1–S_3–D ($service_1 = -1$ and $service_3 = 1$). Using L_1-DEVIATION, the solutions have equivalent objectives of $0 + 0 + 2 = 2$ and $0 + 1 + 1 = 2$, since for this norm unloading a bike at either S_2 or S_3 achieves the same purpose. With L_2-DEVIATION, however, the solutions have objectives of $0^2 + 0^2 + 2^2 = 4$ and $0^2 + 1^2 + 1^2 = 2$. Indeed, provided that other things are equal, it is preferable for this norm to visit the most unbalanced stations, as they have an increased impact on the objective. □

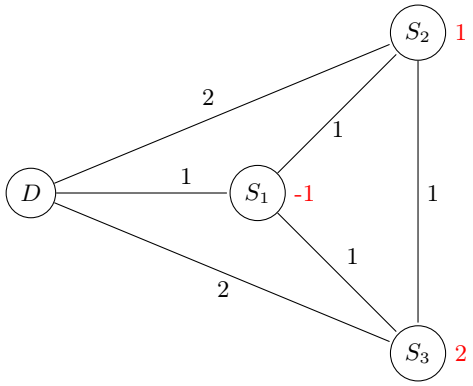

Fig. 1. A simple BBSS problem. The demands of the stations ($t_s - b_s$) are in red (a negative demand implies that the station has an excess of bikes).

We generated 100 instances[11] of 10 stations each, served by a single vehicle and using a single depot.[12] The demands of the stations range from -10 to 10 and they sum to 0 in each instance, i.e., an uncapacitated vehicle unconstrained by the duration of its route would perfectly balance all stations. We solve these instances with a single vehicle in three different contexts: constrained vehicle capacity and unconstrained route length, unconstrained vehicle capacity and constrained route length, constrained vehicle capacity and constrained route length.[13] All instances are solved to optimality with the CP model of [29], which was slightly adjusted to fit our altered objective functions.

[11] Our dataset can be found at https://github.com/PhilippeOlivier/mobico.

[12] We are forced to keep the problem size small in order to solve all instances to optimality in a reasonable amount of time, for illustrative purposes.

[13] When constrained, the capacity is set to 5, which is half the maximum demand of a station. When constrained, the route length is set to 100, which is roughly half the length of an optimal TSP tour of the stations.

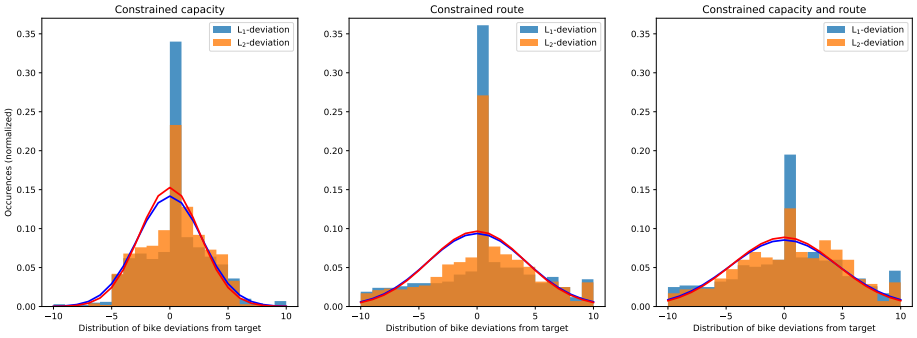

Fig. 2. Normalized distributions of bike deviations from balance target, in various contexts.

Fig. 2 shows the normalized distributions of bike deviations from the balance targets of their stations, in the three contexts mentioned previously. The fitted curves indicate that L_2-DEVIATION offers bike deviations slightly more closely grouped near the mean and with fewer outliers than L_1-DEVIATION. While the former measure offers an arguably marginal improvement in solution quality over the latter, this improvement is constant across variously constrained instances. As for the BACP and NPAP previously, Table 5 shows that L_2-DEVIATION approximates L_1-DEVIATION much better than the reverse.

Table 5. Approximation quality of L_1- and L_2-DEVIATION with respect to each other when the vehicle **C**apacity and/or **R**oute length are constrained.

	C	R	C&R
L_1-DEVIATION	15.45	7.97	9.18
L_2-DEVIATION	1.84	1.78	1.22

Similarly to the previous two sections, Table 6 shows how the various measures of balance approximate each other, and Table 7 examines their characteristics. Considering the combined results of Tables 1, 3, and 6, we can confidently conclude that L_2-DEVIATION is a good approximator of the other measures, independent of context.

Takeaway Due to the small size of the instances, the dispersion interval is the same for both L_1- and L_2-DEVIATION. However, for the 100 instances in the three contexts, 3% of the values are outliers for L_1-DEVIATION, compared to 2.2% for L_2-DEVIATION. For the BBSS problem, this directly translates to an increased number of functional stations. By extension, this means that the whole system offers a better service with L_2-DEVIATION than with L_1-DEVIATION. Smoothness

Table 6. Comparison of measures of balance for the BBSS problem (capacity and route length are constrained).

	MINMAX	L_1-DEV.	L_2-DEV.	L_∞-DEV.	Average
MINMAX	0.00	14.71	23.97	9.30	12.00
L_1-DEVIATION	21.03	0.00	9.18	10.03	9.94
L_2-DEVIATION	6.15	1.22	0.00	2.59	2.52
L_∞-DEVIATION	15.56	17.78	26.23	0.00	14.89
Average	10.69	8.46	14.73	5.48	

Table 7. Characteristics of the solutions of the BBSS problem (capacity and route length are constrained).

	Dispersion	Smoothness	Outliers
MINMAX	1.00	4.05	4.20%
L_1-DEVIATION	1.00	3.57	4.20%
L_2-DEVIATION	1.00	3.61	3.30%
L_∞-DEVIATION	1.00	4.15	3.30%

is 3.57 and 3.61 for L_1- and L_2-DEVIATION, respectively. As for the two previous problems, this is unsurprising, as good solutions will not necessarily tend to be smooth.

4 Practical Considerations

Examination of the characteristics of balance, coupled with the conclusions of the case studies, indicate that no measure of balance is systematically better than the others. Nevertheless some general guidelines concerning the choice of a measure can be derived from the lessons learned in the previous sections.

Rules of Thumb. As a general rule, MINMAX has a large dispersion interval and is robust against outliers. L_1-, L_2-, and L_∞-DEVIATION tend to be more sensitive to outliers, but these outliers have a limited impact due to the shorter dispersion intervals. L_2-DEVIATION is particularly appealing for its good approximation of the other measures (which is not necessarily true the other way around), but it may suffer in performace in an IP model due to its nonlinearity and increased complexity.

Reconsidering Balance. When balance is initially introduced in a problem, it is often chosen to fit more the optimization method used to solve the problem than the problem requirements themselves. It also tends to become the *de facto* standard, and further research into the problem usually achieves balance using the same means, as *this is the way that balance is done for this problem.* The fact that reusing the same measure of balance makes it easier to compare

with previous work also contributes to perpetuate this problem. Balance should nonetheless be reevaluated—the most popular way of achieving balance for a problem is not necessarily the best.

Linearity. If a problem is modeled with IP and already has nonlinear constraints or a nonlinear objective, L_2-DEVIATION can be used without introducing much more complexity. Otherwise, it may be better to compromise and use a linear measure of balance to remain in the linear realm. If using L_2-DEVIATION cannot be avoided, a CP model offers high flexibility.

Practical Constraints. While in theory a given measure of balance can be the best one, in practice there may be time and resource restrictions to consider. A non-optimal solution using an ideal measure of balance may be inferior to an optimal solution using a non-ideal measure of balance. The theoretical quality of a solution may not correlate with its quality in practice.

Neutral Characteristics. Most characteristics, such as the size of the dispersion interval, the number of outliers, and so on, are neutral—they are not inherently desirable nor detrimental. For instance, the distribution of well-balanced workloads would form a narrow bell curve around the mean, since we are interested in having each worker share a similar workload. However, diversity in the occurrences of values could also be desirable, for example to ensure a wide coverage in the test suites of some large software projects [38].

Hybridization. Multiple measures of balance can be combined to suit particular needs. For example, the most extreme values could be bounded with MINMAX, and the result further balanced with L_1-DEVIATION. With enough domain knowledge, such hybrids could present characteristics specifically tailored to a particular problem.

Other Considerations. Marsh and Schilling [10] consider other types of characteristics, some of which have to do with the human factor. For instance, they consider that a user should understand the measures of balance well enough to realize their implications, allowing them to make an informed choice among them. This highlights the fact that in a real-world setting, there is often more to balance than the mathematical facet.

Limitations. In the three cases studied, all concluded that L_2-DEVIATION was a good measure of balance for those problems due to its sensitivity to outliers. However, this specific characteristic is not necessarily desirable at all times. For instance, consider a factory with an equal number of workers and machines. Workers have various proficiencies on the machines, and a commodity is produced when all machines have been operated. Intuition may dictate that balancing the proficiencies of worker-machine pairs will ensure an efficient production. However, Fig. 3 shows that avoiding outliers may be counterproductive in some

situations. The inverse of sensitivity to outliers, robustness against outliers, is itself a desirable characteristic at times as shown in this example.

$$\begin{array}{c} \\ w_1 \\ w_2 \\ w_3 \end{array} \begin{array}{c} \begin{array}{ccc} m_1 & m_2 & m_3 \end{array} \\ \begin{pmatrix} 10 & 8 & 5 \\ 1 & 10 & 9 \\ 7 & 3 & 10 \end{pmatrix} \end{array}$$

Fig. 3. Time required for each worker w_i to operate machine m_j. Perfect balance is achieved when all workers operate the machines on which they are the least proficient (red). In contrast, optimal throughtput is attained when the workloads are most unbalanced (blue).

5 Conclusion

Oftentimes balance is attached to a problem as a side constraint or as a secondary objective without much thought. This paper shows that balancing a solution is not as straightforward as it seems, and highlights a few properties for some types of measures of balance. Three problematic modeling choices have been shown and studied, after which general guidelines have been proposed to prevent modelers from succumbing to pitfalls when selecting a measure of balance.

Acknowledgements

Financial support for this research was provided by NSERC Discovery Grant 218028/2017 and CERC, École Polytechnique de Montréal.

References

1. Whiteacre, K.W.: Testing the level of service inventory–revised (LSI-R) for racial/ethnic bias. Criminal Justice Policy Review **17**(3) (2006) 330–342
2. Corbett-Davies, S., Pierson, E., Feller, A., Goel, S., Huq, A.: Algorithmic decision making and the cost of fairness. CoRR **abs/1701.08230** (2017)
3. Brams, S.J., Jones, M.A., Klamler, C.: Better ways to cut a cake. Notices of the American Mathematical Society **53**(11) (December 2006)
4. Pattanaik, P.K. In: Social Welfare Function. Palgrave Macmillan UK, London (2017) 1–7
5. Sen, A.: Rawls versus Bentham: An axiomatic examination of the pure distribution problem. Theory and Decision **4**(3) (February 1974) 301–309
6. Nash, J.: Two-person cooperative games. Econometrica **21**(1) (1953) 128–140
7. Rossi, F., van Beek, P., Walsh, T., eds.: Handbook of Constraint Programming. Volume 2 of Foundations of Artificial Intelligence. Elsevier (2006)

8. Lodi, A.: Mixed integer programming computation. In Jünger, M., Liebling, T.M., Naddef, D., Nemhauser, G.L., Pulleyblank, W.R., Reinelt, G., Rinaldi, G., Wolsey, L.A., eds.: 50 Years of Integer Programming 1958-2008. Springer, Berlin, Heidelberg (2010) 619–645
9. D'Ambrosio, C., Lodi, A.: Mixed integer nonlinear programming tools: an updated practical overview. Annals of Operations Research **204**(1) (2013) 301–320
10. Marsh, M.T., Schilling, D.A.: Equity measurement in facility location analysis: A review and framework. European Journal of Operational Research **74**(1) (1994) 1–17
11. Ogryczak, W.: Inequality measures and equitable approaches to location problems. European Journal of Operational Research **122**(2) (2000) 374 – 391
12. Pesant, G.: Achieving domain consistency and counting solutions for dispersion constraints. INFORMS Journal on Computing **27**(4) (2015) 690–703
13. Mullinax, C., Lawley, M.: Assigning patients to nurses in neonatal intensive care. Journal of the Operational Research Society **53**(1) (January 2002) 25–35
14. Everitt, B.S., Skrondal, A.: The Cambridge Dictionary of Statistics. Volume 4th Edition. Cambridge University Press (2010)
15. Schaus, P., Deville, Y., Dupont, P., Régin, J.C.: The deviation constraint. Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (2007) 260–274
16. Pesant, G., Régin, J.C. In: SPREAD: A Balancing Constraint Based on Statistics. Springer, Berlin, Heidelberg (2005) 460–474
17. Gaudioso, M., Legato, P.: Linear programming models for load balancing. Computers & Operations Research **18**(1) (1991) 59–64
18. Walla, J., Ruthmair, M., Raidl, G.R.: Solving a video-server load re-balancing problem by mixed integer programming and hybrid variable neighborhood search. In Blesa, M.J., Blum, C., Di Gaspero, L., Roli, A., Sampels, M., Schaerf, A., eds.: Hybrid Metaheuristics, Berlin, Heidelberg, Springer Berlin Heidelberg (2009) 84–99
19. Weng, M.X., Ventura, J.A.: A quadratic integer programming method for minimizing the mean squared deviation of completion times. Operations Research Letters **15**(4) (1994) 205 – 211
20. Olivier, P., Lodi, A., Pesant, G.: The quadratic multiknapsack problem with conflicts and balance constraints. INFORMS Journal on Computing (to appear)
21. Castro, C., Manzano, S.: Variable and value ordering when solving balanced academic curriculum problems. Proceedings of 6th Workshop of the ERCIM WG on Constraints (Prague, June 2001) (November 2001)
22. Hnich, B., Kiziltan, Z., Walsh, T.: Modelling a balanced academic curriculum problem. In Jussien, N., Laburthe, F., eds.: Proceedings of the Fourth International Workshop on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimisation Problems (CP-AI-OR'02), Le Croisic, France (2002) 121–131
23. Hnich, B., Kiziltan, Z., Miguel, I., Walsh, T.: Hybrid modelling for robust solving. Annals of Operations Research **130**(1) (August 2004) 19–39
24. Monette, J.N., Schaus, P., Zampelli, S., Deville, Y., Dupont, P.: A CP approach to the balanced academic curriculum problem. Symcon'07, The Seventh International Workshop on Symmetry and Constraint Satisfaction Problems (July 2007)
25. Chiarandini, M., Di Gaspero, L., Gualandi, S., Schaerf, A.: The balanced academic curriculum problem revisited. Journal of Heuristics **18**(1) (February 2012) 119–148
26. Ceschia, S., Di Gaspero, L., Schaerf, A.: The generalized balanced academic curriculum problem with heterogeneous classes. Annals of Operations Research **218**(1) (July 2014) 147–163

27. Schaus, P., Van Hentenryck, P., Régin, J.C.: Scalable load balancing in nurse to patient assignment problems. In van Hoeve, W.J., Hooker, J.N., eds.: Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems, Berlin, Heidelberg, Springer (2009) 248–262
28. Pesant, G. In: Balancing Nursing Workload by Constraint Programming. Springer International Publishing, Cham (2016) 294–302
29. Di Gaspero, L., Rendl, A., Urli, T.: A hybrid ACO+CP for balancing bicycle sharing systems. In Blesa, M.J., Blum, C., Festa, P., Roli, A., Sampels, M., eds.: Hybrid Metaheuristics, Berlin, Heidelberg, Springer Berlin Heidelberg (2013) 198–212
30. Di Gaspero, L., Rendl, A., Urli, T.: Constraint-based approaches for balancing bike sharing systems. In Schulte, C., ed.: Principles and Practice of Constraint Programming, Berlin, Heidelberg, Springer Berlin Heidelberg (2013) 758–773
31. Rainer-Harbach, M., Papazek, P., Hu, B., Raidl, G.R.: Balancing bicycle sharing systems: A variable neighborhood search approach. In Middendorf, M., Blum, C., eds.: Evolutionary Computation in Combinatorial Optimization, Berlin, Heidelberg, Springer Berlin Heidelberg (2013) 121–132
32. Rainer-Harbach, M., Papazek, P., Raidl, G.R., Hu, B., Kloimüllner, C.: PILOT, GRASP, and VNS approaches for the static balancing of bicycle sharing systems. Journal of Global Optimization **63**(3) (November 2015) 597–629
33. Di Gaspero, L., Rendl, A., Urli, T.: Balancing bike sharing systems with constraint programming. Constraints **21**(2) (April 2016) 318–348
34. Contardo, C., Morency, C., Rousseau, L.M.: Balancing a dynamic public bike-sharing system. Technical report, CIRRELT (2012)
35. Chemla, D., Meunier, F., Wolfler Calvo, R.: Bike sharing systems: Solving the static rebalancing problem. Discrete Optimization **10**(2) (2013) 120–146
36. Raviv, T., Tzur, M., Forma, I.A.: Static repositioning in a bike-sharing system: models and solution approaches. EURO Journal on Transportation and Logistics **2**(3) (August 2013) 187–229
37. Schuijbroek, J., Hampshire, R., van Hoeve, W.J.: Inventory rebalancing and vehicle routing in bike sharing systems. European Journal of Operational Research **257**(3) (2017) 992–1004
38. Hemmati, H., Arcuri, A., Briand, L.: Achieving scalable model-based testing through test case diversity. ACM Trans. Softw. Eng. Methodol. **22**(1) (March 2013)
39. Levina, E., Bickel, P.: The earth mover's distance is the mallows distance: some insights from statistics. In: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001. Volume 2. (July 2001) 251–256 vol.2

A Characteristics of Balance

In this appendix, we provide some formal statistical definitions and basic results [14] of the concepts used in the paper in the attempt of making the reading more self contained.

Definitions

Let

- n be the number of balancing variables $x_1, x_2, \ldots, x_n$,
- $v = \sum_{i=1}^{n} x_i$, the sum of values to be distributed to the balancing variables,
- $\mu = v/n$ be the (fixed) mean,
- d be the deviation allowed for the problem,
- ℓ and u be the (nonnegative) lower and upper bounds of variables $x_1, x_2, \ldots, x_n$.

All of the previously-defined variables are integers, except for the mean which could happen to be fractional. The values of d, ℓ, and u are assumed to be feasible, i.e.,

- $0 \leq \ell \leq \lceil \mu \rceil$ and $u \geq \lceil \mu \rceil$,
- $d \geq \lceil \mu \rceil$ for MINMAX,
- $d \geq 0$ if the mean is integral, or $d \geq 1$ if the mean is fractional, for L_∞-DEVIATION,
- The feasible lower bound of d for L_1- and L_2-DEVIATION needs some explanation due to the complexity introduced by the possibility of having a fractional mean. In such a case, the lowest deviation can be achieved by assigning $\overline{f} = v \bmod n$ variables to a value of $\lceil \mu \rceil$, and $\underline{f} = n - \overline{f}$ variables to a value of $\lfloor \mu \rfloor$. The absolute deviation of a variable which is assigned a value of $\lceil \mu \rceil$ is $\overline{g} = \underline{f}/n$, while for a variable which is assigned a value of $\lfloor \mu \rfloor$ it is $\underline{g} = 1 - \overline{g}$. The feasible lower bound d of L_p-DEVIATION is thus $d = \overline{f} \times \overline{g}^p + \underline{f} \times \underline{g}^p$.

Dispersion

The dispersion characteristic represents the interval within which values of the variables can be found.

For MINMAX, the worst-case dispersion interval is

$$\left[\max\left\{\ell, \left\lceil \mu - (n-1) \times (\min\{u, d\} - \mu) \right\rceil\right\}, \min\left\{u, d, \left\lfloor \mu + (n-1) \times (\mu - \ell) \right\rfloor\right\}\right].$$

We set all variables except one to the maximum allowed deviation. The remaining variable determines the lower bound of the dispersion interval. That bound cannot be negative, nor lower than ℓ. A similar reasoning applies to the upper bound.

For L_1-DEVIATION, the worst-case dispersion interval is

$$\left[\max\left\{\ell, \left\lceil \mu - \frac{d}{2} \right\rceil\right\}, \min\left\{u, \left\lfloor \mu + \frac{d}{2} \right\rfloor\right\}\right].$$

In the worst case, one variable can account for at most half of the deviation.

For L_2-DEVIATION, the worst-case dispersion interval is

$$\left[\max\left\{\ell, \left\lceil \mu - \sqrt{d \times \frac{n-1}{n}} \right\rceil\right\}, \min\left\{u, \left\lfloor \mu + \sqrt{d \times \frac{n-1}{n}} \right\rfloor\right\}\right].$$

In contrast with L_1-DEVIATION, for L_2-DEVIATION we need to take into account the number of variables in order to tightly bound the dispersion interval.

For L_∞-DEVIATION, the worst-case dispersion interval is

$$\begin{aligned}&[\max\{\ell, \lceil\mu - d\rceil\}, \lceil\mu - (\min\{u, \lfloor\mu + d\rfloor\} - \mu) \times (n-1)\rceil\},\\ &\min\{u, \lfloor\mu + d\rfloor, \lfloor\mu + (\mu - \max\{\ell, \lceil\mu - d\rceil\}) \times (n-1)\rfloor\}].\end{aligned}$$

Here, we have to take into account three cases. First, a simple case of ℓ or u providing the bound. Second, another simple case of the deviation d bounding the interval. Third, a more complicated case with a similar reasoning as for MINMAX (explained previously).

Outliers

As stated, in this paper, we call an outlier any value equal to one of the two extremal values of the dispersion interval (see previous section). In practice, the understanding of the nature of an outlier is more subtle, as it could be any value *far enough* from the mean, for some definition of *far enough*. Let $i_{\min}$ and $i_{\max}$ be, respectively, the minimum and maximum values of the intervals defined in the previous section.

For MINMAX and L_∞-DEVIATION, the worst-case number of outliers is (let $i_{\text{low}} = \min\{\mu - i_{\min}, i_{\max} - \mu\}$ and $i_{\text{high}} = \max\{\mu - i_{\min}, i_{\max} - \mu\}$)

$$\left\lfloor \frac{n}{1 + \frac{i_{\text{high}}}{i_{\text{low}}}} \right\rfloor + \left\lfloor \frac{n}{1 + \frac{i_{\text{low}}}{i_{\text{high}}}} \right\rfloor.$$

By looking at the ratio of the deviations between the extremes of the dispersion interval and the mean, we can infer the fractions of variables which will be equal to $i_{\min}$ and $i_{\max}$. On the left is the number of outliers which are below the mean, and on the right the number of outliers which are above the mean.

For L_1- and L_2-DEVIATION, things are much more difficult due to the added complexity of managing a (possibly) fractional mean.[14] In fact, we have not been able to devise a closed-form expression for the worst-case number of outliers for these two measures. Such a closed-form expression, *if it exists*, is likely to be overly complicated. We present instead an informal algorithm which achieves the same purpose. For L_1- and L_2-DEVIATION, then, the worst-case number of outliers is[15] computed as

[14] The complexity stems from the fact that an under-mean value and an over-mean value may have distinct fractional parts, and that we do not know beforehand how many values will be under the mean, and how many will be over the mean.

[15] Here, we are assuming that $\mu - \ell \leq u - \mu$. If instead $\mu - \ell > u - \mu$, the logic would be reversed.

1. Make the variables as balanced as possible. They will either be equal to the mean (if the mean is integral), or be equal to one of the two nearest integers of the mean (if the mean is fractional).
2. While the distribution of values is below the prescribed deviation threshold (also taking into account the potential effects of the actions below), and that at least two non-outlier values can still be found among the variables:
   (a) Pick the variable x_i with the lowest value (yet still greater than the value of an outlier), and lower its value by 1.
   (b) If it exists, pick the variable (not x_i and not already an outlier) with the value closest to and lower than $\lfloor \mu \rfloor$. If it does not exist, pick the variable (not x_i and not already an outlier) with the value closest to and greater than $\lceil \mu \rceil$. Increase the value of this variable by 1.
3. Count and return the number of outliers.

This algorithm starts with a balanced distribution of values, and maximizes the number of low-valued outliers (recall that in this particular case, we assume that $\mu - \ell \leq u - \mu$, and as such low-valued outliers are easier to reach than high-valued outliers). When the lowest non-outlier value goes down (step 2a), the highest non-outlier value goes up (step 2b), keeping the sum of values constant. This is done for as long as the deviation threshold allows it.

Smoothness

In order to assess the smoothness of a solution, we compare the distribution of its values to a perfectly smooth distribution, using the Wasserstein distance. This distance is equivalent to the so-called Earth Mover's distance [39]. Given two mounds of earth (in other words, two distributions), this metric represents the effort required to transform one mound of earth into the other. If two distributions are the same, their Wasserstein distance is zero. As the differences between two distributions increase, so does their Wasserstein distance.

B BACP Model

This BACP model, written in a logical form, uses a similar notation as [24]. Let

- $\mathcal{C} = \{1, \ldots, n\}$ be the index set of courses,
- $\mathcal{P} = \{1, \ldots, m\}$ be the index set of periods,
- w_i denote the load of course i with $w = \sum_{i \in \mathcal{C}} w_i$ representing the combined loads of all the courses,
- $\mathcal{Q} \subset \mathcal{C} \times \mathcal{C}$ denote the set of prerequisites, where an element (i, j) indicates that course i is a prerequisite to course j,
- $L = \{L_1, \ldots, L_m\}$ denote the loads of the periods for an assignment,
- $P_i \in \mathcal{P}$ denote the period course i is assigned to,
- B_{ik} denote if course i is assigned to period k.

The model is defined by

$$P_i < P_j, \qquad \forall (i,j) \in \mathcal{Q} \tag{1}$$
$$(P_i = k) \Leftrightarrow (B_{ik} = 1), \qquad \forall i \in \mathcal{C}, k \in \mathcal{P} \tag{2}$$
$$L_k = \sum_{i \in \mathcal{C}} B_{ik} w_i, \qquad \forall k \in \mathcal{P} \tag{3}$$
$$B_{ik} \in \{0,1\}, \qquad \forall i \in \mathcal{C}, k \in \mathcal{P} \tag{4}$$

and its objectives are

$$\max_{k \in \mathcal{P}} L_k \qquad \text{(MINMAX)}$$
$$\sum_{k \in \mathcal{P}} |L_k - w/m| \qquad (L_1\text{-DEVIATION})$$
$$\sum_{k \in \mathcal{P}} (L_k - w/m)^2 \qquad (L_2\text{-DEVIATION})$$
$$\max_{k \in \mathcal{P}} |L_k - w/m| \,. \qquad (L_\infty\text{-DEVIATION})$$

Constraints (1) ensure that course prerequisite requirements are met, and period loads L are tracked with the help of auxiliary variables B (2)–(4).

C NPAP Model

This NPAP model, written in a logical form, uses a similar notation as [28].[16] Let

- $\mathcal{N} = \{1, \ldots, n\}$ be the index set of nurses,
- $\mathcal{P} = \{1, \ldots, m\}$ be the index set of patients,
- a_i denote the acuity of patient i with $a = \sum_{i \in \mathcal{P}} a_i$ representing the combined acuities of all the patients,
- $p_{\min}$ and $p_{\max}$ denote the minimum and maximum number of patients that can be assigned to a nurse,
- n_i denote the nurse to which patient i is assigned,
- w_j denote the workload of nurse j,
- t_{ij} denote if patient i is assigned to nurse j.

The model is defined by

[16] In the interest of simplicity, we have left aside the staffing part of the NPAP as it is not particularly meaningful for our purposes. The reader may refer to the cited paper for details.

$$(n_i = j) \Leftrightarrow (t_{ij} = 1), \qquad \forall i \in \mathcal{P}, j \in \mathcal{N} \tag{5}$$

$$t_{ij} \in \{0, 1\}, \qquad \forall i \in \mathcal{P}, j \in \mathcal{N} \tag{6}$$

$$w_j = \sum_{i \in \mathcal{P}} a_i t_{ij}, \qquad \forall j \in \mathcal{N} \tag{7}$$

$$p_{\min} \leq \sum_{i \in \mathcal{P}} t_{ij} \leq p_{\max}, \qquad \forall j \in \mathcal{N} \tag{8}$$

and its objectives are

$$\max_{j \in \mathcal{N}} w_j \tag{MINMAX}$$

$$\sum_{j \in \mathcal{N}} |w_j - a/m| \tag{L_1-DEVIATION}$$

$$\sum_{j \in \mathcal{N}} (w_j - a/m)^2 \tag{L_2-DEVIATION}$$

$$\max_{j \in \mathcal{N}} |w_j - a/m| \,. \tag{L_∞-DEVIATION}$$

Auxiliary variables t (5)–(6) are used to track nurse workloads w (7), as well as to ensure the nurses are assigned a proper number of patients (8).

D BBSS Model

We present the CP-based BBSS model of [29], using that notation (an example of a non-CP formulation can be found in [32]). Let

- $\mathcal{S} = \{1, \ldots, S\}$ be the set of S stations,
- $\mathcal{D} = \{S + 1, \ldots, S + D\}$ be the set of D depots,
- for a station $s \in \mathcal{S}$, $C_s > 0$ be its capacity, b_s its initial number of bikes, and t_s its target number of bikes,
- $\mathcal{V} = \{1, \ldots, V\}$ be the set of V vehicles,
- for a vehicle $v \in \mathcal{V}$, $c_v > 0$ be its capacity, $\hat{b}_v \geq 0$ its initial load, and $\hat{t}_v > 0$ its available time,
- travel time matrix tt_{uv} with $u, v \in \mathcal{S} \cup \mathcal{D}$ (which includes the processing time of serving a station).

This model requires the stations and depots to be grouped into an ordered set of nodes. The depots are duplicated as we need distinct starting and ending nodes, and a dummy vehicle (with an associated depot) is introduced. The nodes $\mathcal{U} = \mathcal{V}_s \cup \mathcal{S} \cup \mathcal{V}_e = \{0, \ldots, V, V + 1, \ldots, V + S, V + S + 1, \ldots, 2V + S + 2\}$ begin with the starting depots $\mathcal{V}_s$ (including that of the dummy vehicle), then the stations $\mathcal{S}$, and finally the ending depots $\mathcal{V}_e$ (again including that of the dummy vehicle).

This formulation makes use of a successor-predecessor dynamic. Several auxiliary variables are required for this purpose

– $succ_i \in \mathcal{U}$ denotes the successor of node $i \in \mathcal{U}$,
– $pred_i \in \mathcal{U}$ denotes the predecessor of node $i \in \mathcal{U}$,
– $vehicle_i \in \mathcal{V}$ denotes the vehicle serving node $i \in \mathcal{U}$,
– $service_i \in \{-b_i, \ldots, C_i - b_i\}$ denotes the bike delta of node $i \in \mathcal{U}$ after being served,
– $load_i \in \{0, \ldots, c_v\}$ denotes the load of vehicle $v \in \mathcal{V}$ after serving node $i \in \mathcal{U}$,
– $time_i \in \{0, \ldots, \hat{t}_v\}$ denotes the time at which vehicle $v \in \mathcal{V}$ arrives at node $i \in \mathcal{U}$.

The model is defined by

$$\texttt{alldifferent}(succ) \tag{9}$$
$$\texttt{alldifferent}(pred) \tag{10}$$
$$pred_{succ_s} = s, \qquad \forall s \in \mathcal{S} \tag{11}$$
$$succ_{pred_s} = s, \qquad \forall s \in \mathcal{S} \tag{12}$$
$$pred_v = V + S + v, \qquad \forall v \in \mathcal{V}_s \tag{13}$$
$$succ_{V+S+v} = v, \qquad \forall v \in \mathcal{V}_s \tag{14}$$
$$pred_i \neq i, \qquad \forall i \in \mathcal{U} \tag{15}$$
$$succ_i \neq i, \qquad \forall i \in \mathcal{U} \tag{16}$$
$$vehicle_v = v, \qquad \forall v \in \mathcal{V}_s \tag{17}$$
$$vehicle_{V+S+v} = v, \qquad \forall v \in \mathcal{V}_s \tag{18}$$
$$vehicle_{succ_i} = vehicle_i, \qquad \forall i \in \mathcal{U} \tag{19}$$
$$vehicle_{pred_i} = vehicle_i, \qquad \forall i \in \mathcal{U} \tag{20}$$
$$load_v = \hat{b}_v, \qquad \forall v \in \mathcal{V}_s \setminus \{V\} \tag{21}$$
$$loadV = 0 \tag{22}$$
$$load_{succ_i} = load_i - service_i, \qquad \forall i \in \mathcal{U} \tag{23}$$
$$load_v = 0, \qquad \forall v \in \mathcal{V}_e \tag{24}$$
$$(vehicle_s \neq V) \Leftrightarrow (service_s \neq 0), \qquad \forall s \in \mathcal{S} \tag{25}$$
$$load_s \leq c_{vehicle_s}, \qquad \forall s \in \mathcal{S} \tag{26}$$
$$service_s \leq 0, \qquad \forall s \in \mathcal{S} : b_s > t_s \tag{27}$$
$$service_s \geq 0, \qquad \forall s \in \mathcal{S} : b_s < t_s \tag{28}$$
$$service_i = 0, \qquad \forall i \in \mathcal{V}_s \cup \mathcal{V}_e \tag{29}$$
$$b_s + service_s \leq C_s, \qquad \forall s \in \mathcal{S} \tag{30}$$
$$b_s + service_s \geq 0, \qquad \forall s \in \mathcal{S} \tag{31}$$
$$time_v = 0, \qquad \forall v \in \mathcal{V}_s \tag{32}$$
$$time_v = time_{pred_v} + tt_{pred_v,v}, \qquad \forall v \in S \cup \mathcal{V}_e \tag{33}$$
$$time_{succ_v} = time_v + tt_{v,succ_v}, \qquad \forall v \in \mathcal{V}_s \cup S \tag{34}$$
$$time_{V+S+v} \leq \hat{t}_v, \qquad \forall v \in \mathcal{V}. \tag{35}$$

All values of the successors and predecessors are distinct (9)–(10); these constraints, coupled with the time constraints of the vehicles, ensure the absence of subtours. The successor-predecessor chain must be consistent (11)–(14), and loops are forbidden (15)–(16). Depots are assigned to the vehicles (17)–(18), and the vehicle chain must be consistent (19)–(20). The initial loads of the vehicles are set (21)–(22), the load chain must be consistent (23), and the vehicles must be empty at the end of their routes (24). A station receiving no service will be visited by the dummy vehicle (25), and the loads of the other vehicles must not exceed their capacities (26). Stations visited by a vehicle must see their bike counts altered in some way (27)–(28), while depots remain unserved (29). Stations cannot be served in excess of their capacities (30)–(31). The routes start at the depots with times of zero (32), the time chain must be consistent (33)–(34), and the durations of the routes must remain within the specified limits (35). The objectives are defined by

$$\max_{s \in \mathcal{S}} (b_s + service_s - t_s) \qquad \text{(MINMAX)}$$

$$\sum_{s \in \mathcal{S}} |b_s + service_s - t_s| \qquad (L_1\text{-DEVIATION})$$

$$\sum_{s \in \mathcal{S}} (b_s + service_s - t_s)^2 \qquad (L_2\text{-DEVIATION})$$

$$\max_{s \in \mathcal{S}} |b_s + service_s - t_s| \,. \qquad (L_\infty\text{-DEVIATION})$$