Exploring Hypervolumes in Multi-Objective Optimization Algorithms

Qiuyu Chen, Uzair Chhapra, Niraj Lavani

School of Computer Science
North Carolina State University
Raleigh, North Carolina

***Abstract*—This research paper investigates the improvement of a robust and well-balanced multi-objective optimizer program, named SWAY, through the use of Hypervolumes paired with the Walking-Fish-Group algorithm. Human decision-making can be quite complex and choosing the wrong solution can be costly. There is ever-growing need for accurate multi-objective optimizers to narrow down the best possible solutions to complex problems. Given that Hypervolumes can find multi-objective solutions with respectable accuracy through the use of an objective volume, we believe there is a case to be made that Hypervolumes will improve SWAY. Specifically, we tested two mutually-exclusive Hypervolume implementations to SWAY against 11 datasets acquired from different industries like sciences, software, vehicle manufacturing, and team management, with the original SWAY as the control. The results showed that both Hypervolume approaches were generally worse than the original SWAY and occasionally better in select objectives. Our findings suggest that our Hypervolume implementation tested in this study did not improve the performance of SWAY, however, this is likely due to majority of the datasets having relatively few objectives and correlating objectives. These objective conditions make Hypervolumes perform poorly compared to Zitzler's Domination Predicate which performs well under such conditions [9][10]. These results highlight the importance of carefully evaluating the datasets to influence the decision of which multi-objective performance metric to choose.**

I. INTRODUCTION

Humans have to make many judgments every day based on a variety of considerations and goals. Making decisions may be difficult, whether they are simple ones like what to eat for breakfast or more difficult ones like finding a home. Certain choices can be unnecessarily difficult, and the best course of action might not always be evident. Numerous factors, such as individual preferences, social expectations, budgetary restraints, and long-term objectives, may play a role in these difficult decisions, making them more difficult to make. Therefore, choosing the best decision is not always easy because human decision-making is nuanced.

By simultaneously optimizing many goals, multi-objective optimization algorithms can filter out the best solutions in complex decision making. Consider a decision with 20 or more factors like deciding what features a car manufacturer will prioritize; it would be challenging for a person to find a set of solutions. Multi-objective functions, paired with a dimension reduction algorithm, are designed to easily manage 20 or more objectives and filter out potentially expensive solutions. Multi-objective optimization can assist us in making more informed decisions and achieving better results by taking into account all pertinent elements and objectives.

Multi-objective optimization algorithms work by searching for a set of solutions that represents the best trade-offs between the different objectives mentioned by N. Riquelme et al. [3]. The imaginary solution decision that has the best outcome of all objectives is called the Utopia. In layman's terms, the optimization algorithm strives to find a set of solutions closest to Utopia. The Pareto Frontier, a collection of Pareto-optimal solutions, shows the best compromises between the competing objectives.

After refactoring Dr. Menzies' optimizer– that we will call "SWAY"– into Python, we found opportunities of improvement to make the multi-objective algorithm perform more efficiently, specifically, when SWAY compares two points to see which point is better. SWAY at its foundation uses a KD-Tree implementation to cluster the data followed by FASTMAP to reduce the multi-objective volume to an objective-prioritizing plane; FASTMAP lifts the "curse of dimensionality" associated with *KD-Trees* processing large multi-dimensional datasets. When optimizing, SWAY uses Zitzler's Domination Predicate to perform comparisons between two selection points to see which point dominates the other. Determined by Zitzler et al., a continuous cost function that returns a scalar value for each point serves as the basis of comparison[4].

We replaced Zitzler's Domination Dredicate with the Hypervolume Indicator calculation, a performance metric introduced by Guerreiro et al. [5] for multi-objective comparisons and Boolean Domination for single-objective comparisons. We tested 2 mutually-exclusive approaches that utilize Hypervolumes. The Hypervolume Indicator tells us how 2 sets of solutions compare to each other in terms of objective quality and diversity in a multi-objective volume. Quality refers to how well the point contributes to the different objectives while diversity refers to how well the set of solutions cover the objective volume. N. Riquelme et al. conducted a review and analysis of 54 existing multi-objective optimization metrics, and found that the Hypervolume, the top one unary metric, is the most widely cited and accurate among them, considering its accuracy, diversity, and cardinality [3]. We aim to explore how Hypervolumes can improve the SWAY algorithm given its accuracy and positive reviews.

II. Related Works

A large number of metrics have been proposed for evaluating multi-objective optimization, and finding a suitable substitute for Zitzler's Domination Predicate could be an important direction of our research. As we discussed in the introduction, the Hypervolume is a good approach to substitute Zitzler's.

The Hypervolume indicator was originally proposed as a method to assess multi-objective optimization and has since gained recognition as a useful indicator [3]. It measures the space that is dominated by a set of solutions, with larger Hypervolumes indicating better solutions. To calculate the Hypervolume, a reference point must be defined. The Hypervolume indicator is particularly suitable for evaluating multi-objective optimizers in stochastic circumstances and may serve as a potential substitute for the Zitzler's domination indicator [5].

However, one of the biggest challenges with the Hypervolume indicator is its computational cost and selective usage, particularly in high-objective spaces. In the 2020 paper "The Hypervolume Indicator: Problems and Algorithms," Guerreiro, Fonseca, and Paquete discuss current issues with the Hypervolume indicator, and introduce several algorithms that may help in efficiency [5]. The Walking-Fish-Group (WFG) algorithm is one such approach that has shown promising results [1].

Hypervolumes are also known to be quite generalized in their performance. Research by H. Ishibuchi et al. suggests Hypervolumes are practical in cases where the target dataset have uncorrelated objectives or have many objectives [9]. Deb, K. et al. suggests Zitzler's Domination Predicate, in the opposite manner, is better when the target dataset has few objectives and has correlated objectives [10]. From this understanding, we see the target dataset plays a sizable role in how well the metric of choice performs in terms of accuracy and speed.

To address the challenge of computation time of Hypervolumes, the WFG algorithm has been proposed as one of the fastest methods for computing the Hypervolume, particularly for more than 7 objectives. The WFG algorithm, introduced by Lyndon While et al. in 2012, is a mutually recursive combination of point-wise and exclusive Hypervolume calculations. Exclusive Hypervolume is the space that is dominated by a given point but not by any of the other points in the set. The WFG algorithm sums the exclusive Hypervolumes of each point by limiting the set to only those points that dominate a given point, which reduces the number and sizes of sets that need to be examined [1].

III. Methods

We looked at all of SWAY'S components in an effort to find areas of improvement. We explored alternatives to dimension reduction, evaluation functions, hyperparameters, and multiple ways to recursively divide clusters in half. We found SWAY to be a well-balanced optimization algorithm, so much so, it was difficult to decide how to make a positive change to SWAY in terms of run-time, budget, and accuracy.

A. Two optimization approaches using Hypervolumes

We landed on 2 mutually-exclusive approaches to improve the SWAY program; both of which include the use of Hypervolumes. Our first approach replaces the Zitzler's Domination Predicate in the "better" function with a Hypervolume calculation. Hypervolumes typically compare 2 sets of points but in this approach, the 2 sets only contain 1 point each. Improving "better" means we are directly improving how the algorithm decides how one point is better than, or dominates, another. When comparing two points, Zitzler's Domination Predicate checks the cost of jumping from one point to the other; the point that loses the least (the lowest cost) when jumping to it is the better point [4]. By implementing an alternative to Zitzler's Domination Predicate in this approach, we can not only aim to maintain budgetary restrictions but also accurately choose the better point through the use of Hypervolumes.

The second approach performs a Hypervolume comparison of two sets of data: *Left.some* and *Right.some*.The version of SWAY designed for our second approach still follows the original SWAY paradigm of dividing the dataset into two halves, *Left* and *Right*, based on two far points, *A* and *B*. This approach also uses the improved "better" function created in the first approach but instead of providing 2 points for comparison, the SWAY function provides two sets of points. We use 80% of the halves in the Hypervolume calculation to reduce the budget. This is represented through the use of *.some* in *Left.some* and *Right.some*. If we didn't take a sample, the Hypervolume function would examine every possible solution at the start of the optimization which is unrealistic.

B. Hypervolume Requirements

Hypervolume indicator has been identified as a recently well-liked multi-objective indicator, as we covered in the Related Works section. To put our strategy into practice, we used the PYGMO library[7]. A scientific package called PYGMO offers optimization methods for use in highly parallel systems. It facilitates a simple integration of Hypervolume into our programs by offering well-implemented solutions.

To use the Hypervolume, we must first define the reference point in order to compare two data rows. A user-defined reference point is a point that represents the best potential outcome. The range of our data, which has been normalized is between 0 and 1. In this instance, we set the reference point to 1 in order to maximize the goal. If we want to minimize that instead, we take the row data's negation and set the reference point number to 0. As an example, let's say there are three attributes: Weight, Mpg, and Acc.

We would like to minimize the weight, and maximize the Mpg and Acc. We can calculate the Hypervolume contribution by negating the sample data [0.2, 0.5, 0.3] to -0.2 and setting the reference point to [0, 1, 1]. This allows us to compare the contributions of two data rows when comparing them. PYGMO does not allow the reference point to be set to the largest value. In this scenario, we can adjust the reference point by adding 0.1 to either 0 or 1, depending on the specific problem.

C. The Walking-Fish-Group Algorithm

As emphasized in the Related Works section, Hypervolumes are great at solving multi-objective optimization problems. The Walking-Fish-Group (WFG) technique, the high-objective algorithm behind Hypervolume, can solve this problem. The PYGMO library determines the algorithm by identifying the number of objectives of the input data. It calculates the standard 2D or 3D volume for data in 2 and 3 objectives, respectfully. The WFG algorithm is applied to any data that has objectives more than 3.

The WFG algorithm, shown as pseudo-code in Figure 2, works by restricting the underlying sets and constructing exclusive Hypervolumes. The region of the objective space that is dominated by p but not by any other member of S is known as the exclusive Hypervolume of a point p with regard to an underlying set S [5]. In Layman's terms, the exclusive Hypervolume is the Hypervolume of a single solution in a solution set and is the relative volume compared to the reference point. The Hypervolume is the sum of exclusive Hypervolumes. In order to determine the sum of exclusive Hypervolumes, LebMeasure developed the Point-Wise approach [6] in Figure 2. We can efficiently acquire the Hypervolume by restricting the underlying set with the contributing point p and deducting the inclusive Hypervolume of p from the inclusive Hypervolume of the modified set [6].

$$Hyp(\{p_1, \dots, p_m\}) = \sum_{i=1}^{m} ExcHyp(p_i, \{p_{i+1}, \dots, p_m\}).$$

Fig. 1. Point-Wise Approach in Evaluating Hypervolume

```
wfg(pl):
  return sum {exclhv(pl, k) | k in {1 .. |pl|}}

exclhv(pl, k):
  return inclhv(pl[k]) - wfg(nds(limitset(pl, k)))

inclhv(p):
  return product {|p[j] - refPoint[j]| | j in {1 .. n}}

limitset(pl, k):
  for i = 1 to |pl| - k
    for j = 1 to n
      ql[i][j] = worse(pl[k][j], pl[k+i][j])
  return ql

nds(pl) returns the non-dominated subset of pl
```

Fig. 2. WFG Algorithm

D. Hypervolume Limitations

One limitation of Hypervolume, as a metric for multi-objective optimization, is it cannot be applied to single objective data. In such cases, a Boolean Domination comparison can be added to determine which solution is better. Boolean Domination provides a simple criterion - A dominates B only if A is just as good as B in all objectives and A is better than B in at least one objective [8]. However, when dealing with more than three objectives, it can become challenging to distinguish which solution is better. If there is only one objective, and Hypervolume cannot handle this situation, a Boolean Bomination comparison is used to compare the points. This way, we cover all possible objective counts; Boolean Domination for single-objective solution comparisons, and Hypervolume for multi-objective solution comparisons.

E. Datasets

In order to evaluate the effectiveness of our Hypervolume implementation, we conducted a research study using 11 distinct datasets. As seen in Figure 3, these datasets come from a variety of applications like car data, team management scores, software effort estimation techniques, and more. High-level notes can be see in Figure 3 that further describe the datasets. These datasets have 4 or less objectives. Some of these datasets have aligning objectives. For example, *auto93.csv* is a dataset of car data. The objectives are to find a car with the lowest weight, highest MPG, and highest acceleration. The objectives suggest the goal is to find a fast, nimble, and efficient vehicle which is more realistic than a vehicle that is extremely heavy with great acceleration and MPG.

We compared the original SWAY with the new updated SWAY, as well as all other data, by running explanations and sorting the results to find the best results. Our goal was to determine how well our two approaches perform when applied to a variety of datasets using the original SWAY as our control.

F. Performing Tests

We used the central tendency to get the median or mode of an attribute from the entire dataset, the best after SWAY and the samples based on the explanation rules. We compare the difference using non-parametric statistical tests after keeping the results of 20 runs with various seeds. Due to only having 20 medians in the result of each dataset optimization, the assumption of the normal distribution cannot be simply applied. In order to compare the medians and determine whether or not our new technique improved upon SWAY, we use both CLIFFSDELTA and BOOTSTRAP. CLIFFSDELTA measures the effect-size of in the results which tells us if the difference between SWAYs is not trivial. BOOTSTRAP is a non-parametric significance test that tells us if the difference in results are statistically significant or not.

IV. RESULTS

We developed a controller program to manage 11 datasets and 20 trials for each. The program automatically executed 20 repetitions for each dataset, using the current time as the random seed. For each repetition, the program generated a single table containing the median value of each attribute. We stored the results of all 20 trials and compared their differences, such as SWAY1 to SWAY2, and ALL to SWAY1, using both CLIFFSDELTA and BOOTSTRAP methods. SWAY1 refers to the original SWAY, while SWAY2 utilizes Hypervolume for comparing two points (our first approach), and SWAY3 employs Hypervolume to compare two sets of solutions, which are sampled from the left and right (our second approach). We considered the result of SWAY1 to be statistically equal to that

Dataset name	Category	No. of Objectives	Notes
auto2	Vehicle Data	4	Information about a car such as the manufacturer, seats, class, city and highway MPGs, etc.
auto93		3	Similar to auto2, this dataset has car information such as weight, acceleration, and MPG
china	Software Effort Estimation	1	Data that represents software estimation to compare new estimation techniques to old techniques. Can use Boolean Domination since there is only 1 objective.
coc1000		4	Data that represents software estimation to compare new estimation techniques to old techniques. Columns represent different aspects of a development process such as RAM utilization, product complexity, and developer turnover. Application and Platform experience are unique objectives
coc10000		3	Data that represents software estimation to compare new estimation techniques to old techniques. Columns represent different aspects of a development process such as RAM utilization, product complexity, and developer turnover.
nasa93dem	Software Effort and Defect Estimation	3	Data that represents software estimation to compare new estimation techniques to old techniques. Columns represent different aspects of a development process, defects, and construction time in months
healthCloseIsses12mths0001-hard	Issue Close Time	3	Dataset consists of rows that represent parameters to ExtraTreesClassifier(), a meta estimator function that fits a number of randomized decision trees from SciKit Learn Library.
healthCloseIsses12mths0011-easy		3	
pom	Agile Project Management	3	Explores Agile Development cycle from a team management perspective. Balances the following objectives: idle rates of members, completion rate of tasks, and cost.
SSM	Computational Physics	2	Dataset about TriMesh, a library for manipulating triangle meshes. Objectives are Iteration count and time to solution
SSN		2	Objectives are configuration space parameters of X-264 a video encoder

Fig. 3. Dataset Information

of SWAY2 for a specific attribute only if both CLIFFSDELTA and BOOTSTRAP indicated they were equal. Our standard for statistical significance was a 95% confidence level, and we considered the results to be statistically different if they were significantly better. Therefore, we can make the conclusion that we were 95% confident that one is better than the other.

```
X---------------------SSM.csv----------------------X

       NUMBERITERATIONS-  TIMETOSOLUTION-
title
all                 7.00           134.89
sway1               5.10           120.04
sway2               5.40           120.22
sway3               5.15           104.69
top                 4.00            59.97
xpln1               5.45           125.68
xpln2               5.60           120.10
xpln3               5.65           103.89
```

Fig. 4. Sample output SSM(1)

```
                 NUMBERITERATIONS- TIMETOSOLUTION-
all to all                       =               =
all to sway1                     ≠               =
all to sway2                     ≠               ≠
all to sway3                     ≠               ≠
sway1 to sway2                   =               =
sway1 to sway3                   =               =
sway1 to xpln1                   =               =
sway2 to xpln2                   =               =
sway3 to xpln3                   ≠               =
sway1 to top                     ≠               ≠
sway2 to top                     ≠               ≠
sway3 to top                     ≠               ≠

   sway1_time  sway2_time  sway3_time
0    4.957352    4.941271    5.766362
```

Fig. 5. Sample output SSM(2)

Figure 4 and 5 show a sample output for the SSM dataset. The Fig4 table displays the average of the medians from 20 runs of each attribute. The Fig5 top table presents the CLIFFSDELTA and BOOTSTRAP results to determine if there are any statistical differences between the attributes. In this particular sample output, we cannot conclude that SWAY2 or SWAY3 is significantly different from SWAY1, even though their means differ numerically. The Fig5 bottom table provides the average run-time for each SWAY.

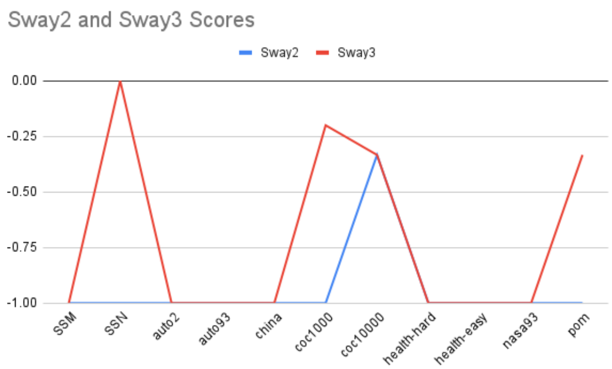

Fig. 6. Sway2 and Sway3 Score

Figure 6 illustrates the score difference between SWAY2 and SWAY3. The score is determined using the following formula:

$$\text{SCORE} = \frac{n_A + n_B}{n_T} \tag{1}$$

where: n_A = No. of attributes better than SWAY1
n_B = No. of attributes worse or equal than SWAY1
n_T = Total number of attributes

The diagram indicates that neither SWAY2 nor SWAY3 is superior to SWAY1. While SWAY3 generally performs better than SWAY2, this comes at the cost of increased run-time and budgets. The highest score obtained was for the SSN dataset in SWAY3, which is one attribute better than SWAY1 and one attribute equal to it.

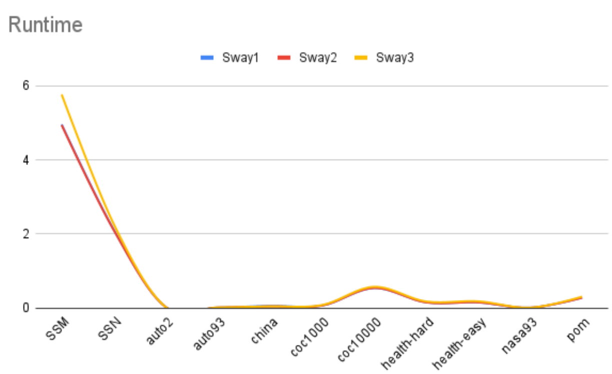

Fig. 7. Run-time

We did not observe significant differences in run-time between the methods. SWAY3 does take slightly longer for larger datasets such as SSM and SSN, which is expected as it compares two sets of solutions rather than just two points, as in SWAY1 and SWAY2. However, we did not observe a significant difference in run-time between SWAY1 and SWAY2.

X------------------coc1000.csv-------------------X

title	LOC+	AEXP-	PLEX-	RISK-	EFFORT-
all	1060.00	3.00	3.00	5.00	19641.00
sway1	1121.40	2.80	3.00	4.60	20114.00
sway2	1051.45	2.65	3.00	3.95	17797.10
sway3	1062.50	3.05	3.00	2.00	13279.25
top	1540.00	2.00	1.00	3.00	30241.00
xpln1	1042.25	2.90	3.05	5.10	20568.50
xpln2	1044.35	3.00	3.00	4.80	18256.65
xpln3	1056.00	3.00	3.00	4.85	18220.10
samp tax1	-61.40	0.20	0.00	0.40	-473.00
xpln tax1	79.15	-0.10	-0.05	-0.50	-454.50
samp tax2	8.55	0.35	0.00	1.05	1843.90
xpln tax2	7.10	-0.35	0.00	-0.85	-459.55
samp tax3	-2.50	-0.05	0.00	3.00	6361.75
xpln tax3	6.50	0.05	0.00	-2.85	-4940.85

Fig. 8. coc1000

Additionally, we outputted the Sampling Tax and Explanation Tax while generating the tables, as shown in Figure 8.

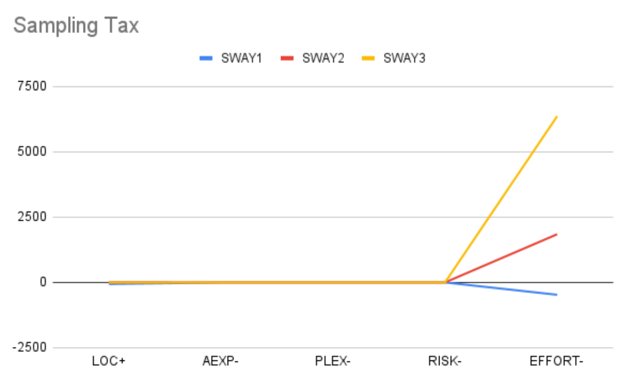

Fig. 9. Sampling Tax

Figure 9 illustrates the sampling tax previously mentioned in Figure 7. It shows that there is little difference between the sampling taxes until reaching the EFFORT- level. Among the three methods, SWAY3 has the highest sampling tax, while SWAY1 has the lowest, indicating that SWAY1 suffered the least from the sampling. However, for the dataset COC1000, SWAY2 and SWAY3 did not perform well.

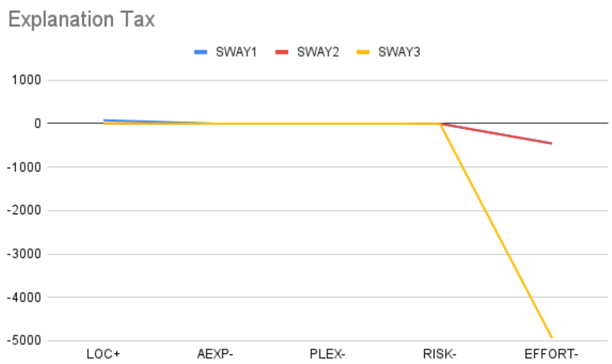

Fig. 10. Explanation Tax

On the other hand, Figure 10 shows the explanation tax, which reveals that the rules generated from SWAY3 result in a lower explanation tax compared to SWAY2 and SWAY1, which perform similarly.

V. Discussion

Based on the 11 tables stored in our Github repository, we found that there was no significant improvement or difference between SWAY1, SWAY2, and SWAY3. In most cases, the results indicated that SWAY1 and SWAY2 had similar means, sampling tax, and explanation tax. However, some datasets showed that SWAY1 performed much better or worse than SWAY2. We conducted further research to determine the possible reasons for such results and found that Hypervolume performs better on non-correlated high-objectives data, as mentioned in [9] and [10]. However, Zitzler's indicator is better suited for fewer objectives and correlated datasets.

Therefore, we conducted another study to evaluate the performance of SWAY in relation to the correlation of specific datasets. In this study, we developed a new scoring system to assess the effectiveness of SWAY. The problem with SCORE1 is that it judges the algorithm too harshly in cases where SWAY2 & SWAY3 are statistically equal to SWAY1.

A better scoring function SCORE2 can be used which ignores objectives that are statistically EQUAL to SWAY1. This approach allows us to determine which datasets are better suited for ZITZLER'S (0-0.5 range) or HYPERVOLUME (0.5-1 range).

$$\text{SCORE2} = \begin{cases} - \quad , & (\mathrm{n}_A + n_B) = 0 \\ \dfrac{n_A}{(n_A + n_B)}, & \text{otherwise} \end{cases}$$

where: $n_A =$ No. of attributes better than SWAY1
$n_B =$ No. of attributes worse or equal than SWAY1

	sway2 to sway1	sway3 to sway1
SSM.csv	-	-
SSN.csv	-	0.0
auto2.csv	-	-
auto93.csv	-	-
china.csv	-	-
coc1000.csv	-	1.0
coc10000.csv	0.5	1.0
healthCloseIsses12mths0001-hard.csv	0.333333	-
healthCloseIsses12mths0011-easy.csv	-	0.5
nasa93dem.csv	-	-
pom.csv	-	0.0

Fig. 11. Score Table

Based on Fig11 it is evident that SWAY3 performs well for the COC1000 dataset, outperforming SWAY1 in some attributes while not performing worse in any attributes. However, for the POM dataset, SWAY3 performs worse, underperforming in all attributes.

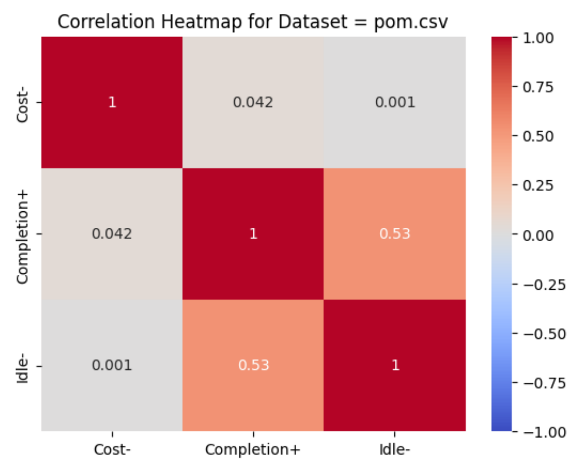

Fig. 12. Pom Correlation

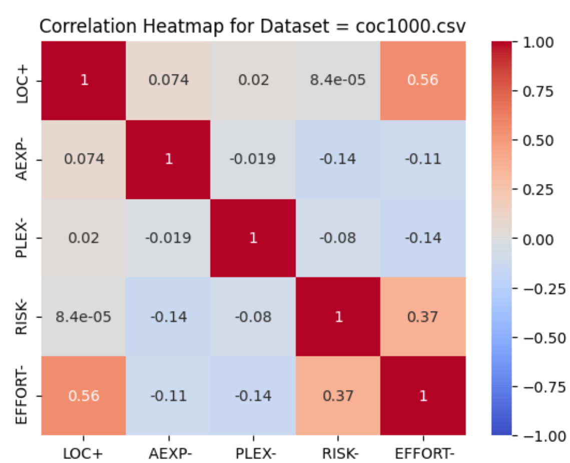

Fig. 13. Coc1000 Correlation

Based on our findings that the correlation between attributes may affect hypervolume performance, we conducted further investigation. As shown in Fig12 and Fig13, we plotted the correlation diagram for both the pom and coc1000 datasets. It is evident that pom has a larger number of correlated attributes compared to coc1000, indicating that it may be better suited for Hypervolume rather than Zitzler's. This suggests that Hypervolume performance may vary depending on the dataset used.

Overall, our analysis suggests that Hypervolume may not necessarily provide significant improvements for the specific 11 datasets analyzed. We found that Hypervolume yielded comparable results to Zitzler's in terms of output, sampling tax, explanation tax, and run-time when compared to SWAY1 and SWAY2. While SWAY3 did show some differences, this came at the cost of budget and run-time. However, our further investigation revealed that Hypervolume performance may be influenced by the objective size and correlation of the data. Thus, SWAY2 or SWAY3 with Hypervolume may be more effective for datasets with larger objectives and non-correlated data.

A. Future Work

If we have a limited budget, our second approach of comparing two datasets by comparing a random set of two halves of a cluster can be quite problematic. In real-world situations, retrieving objective values can be costly or even impossible. We, therefore, seek to investigate additional strategies that allow us to better assess data without relying on objective values.

Without dimension reduction, we may also take different clustering algorithms like *K-means* or *DBScan*. One approach is to keep the recursive structure, divide the dataset in half each time, and then assess the outcomes. There are several ways to do this, including sampling from the dataset and comparing with Hypervolume indicators or comparing the medians of two groups using Zitzler's.

The direct clustering of complete datasets into groups of more than two and sorting of those groupings is an additional option. The settings of hyper-parameters like *K* for *K-means* and *eps* and *min* samples for *DBScan* need to be studied, however.

Additionally, SWAY can be used for hyper-parameter optimization tasks. SWAY can be used, for instance, to determine the ideal hyper-parameters for *DBScan* and *K-means* or to determine the best data sampling and comparison percentages for Hypervolume indicators.

Another avenue to explore in the future is to investigate datasets with a larger number of objectives to optimize (up to 7) or datasets with non-correlating objectives and contrast the effectiveness of Zitzler's Domination Predicate and Hypervolume. Thus, we may determine which methodology is more appropriate for many objectives and use this knowledge to enhance SWAY's evaluation techniques for various datasets. Since the majority of our datasets now have about 4 objectives or less, integrating datasets with more objectives can be incredibly insightful for upcoming studies.

VI. Conclusion

When performing multi-objective comparisons, Hypervolume is a good multi-objective indicator that performs as well as or comparably to Zitzler's method in terms of accuracy and run-time based off of the 11 datasets tested. However, examining a large number of solutions has a substantial impact on run-time efficiency since Hypervolume is more effective in evaluating a collection of solutions as opposed to comparing two solutions. Although Hypervolume is a well-recognized multi-objective indicator, SWAY was not enhanced by it. Both approaches failed to improve SWAY. Our implementation either produces similar results or better results in select objectives at the cost of increased budget and time.

References

[1] Lyndon While, Lucas Bradstreet, and Luigi Barone, *A Fast Way of Calculating Exact Hypervolumes.* IEEE Transactions on Evolutionary Computation 16, 1 (Feb. 2012), 86–95.

[2] Renaud Lacour, Kathrin Klamroth, and Carlos M. Fonseca, *A box decomposition algorithm to compute the Hypervolume indicator.* Computers & OperationsResearch 79 (March 2017), 347–360.

[3] N. Riquelme, C. Von Lücken and B. Baran *Performance metrics in multi-objective optimization* 2015 Latin American Computing Conference (CLEI), Arequipa, Peru, 2015, pp. 1-11

[4] Zitzler, E., Künzli, S. *Indicator-Based Selection in Multiobjective Search* Parallel Problem Solving from Nature - PPSN VIII. PPSN 2004. Lecture Notes in Computer Science, vol 3242. Springer, Berlin, Heidelberg.

[5] Guerreiro, A. P., Fonseca, C. M., & Paquete, L. *The Hypervolume Indicator: Problems and Algorithms.* 2020 arXiv:2005.00515

[6] M. Fleischer *The measure of Pareto optima: Applications to multi-objective metaheuristics* in Proc. Evol. Multiobjective Optimization, LNCS 2632. 2003, pp. 519–533.

[7] Biscani et al. *A parallel global multiobjective framework for optimization: pagmo.* Journal of Open Source Software, 5(53), 2338, https://doi.org/10.21105/joss.02338

[8] Wagner, T., Beume, N., Naujoks, B. *Pareto-, Aggregation-, and Indicator-Based Methods in Many-Objective Optimization.* Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T. (eds) Evolutionary Multi-Criterion Optimization. EMO 2007. Lecture Notes in Computer Science, vol 4403. Springer, Berlin, Heidelberg.

[9] H. Ishibuchi, M. Yamane and Y. Nojima *Effects of duplicated objectives in many-objective optimization problems on the search behavior of hypervolume-based evolutionary algorithms* 2013 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making (MCDM), Singapore, 2013, pp. 25-32, doi: 10.1109/MCDM.2013.6595440.

[10] Deb, K., Agrawal, S., Pratap, A., Meyarivan, T. *A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II* Parallel Problem Solving from Nature PPSN VI. PPSN 2000. Lecture Notes in Computer Science, vol 1917. Springer, Berlin, Heidelberg. doi:10.1109/4235.996017