Extending Collective Intelligence Evolutionary Algorithms: A Facility Location Problem Application

Daniel Cinalli
TecnipFMC
Rio de Janeiro, Brazil
dcinalli@technip.com

Luis Martí, Nayat Sanchez-Pi
Inria Chile Research Center
Santiago de Chile, Chile
{luis.marti,nayat.sanchez-pi}@inria.cl

Ana Cristina Bicharra Garcia
Universidade Federal do Estado do Rio de Janeiro
Rio de Janeiro, Brazil
cristina.bicharra@uniriotec.br

***Abstract*—This work extends current collective intelligence evolutionary algorithms by incorporating a collective-based variation operator. As part of this work, the proposals are compared with state-of-the-art reference-point-based MOEAs: NSGA-II and R-NSGA-II. Another primary objective of the work is to deal with a real-world multi-objective instance of the facility location problem. The experimental results validate the proposal. The new collective intelligence MOEA outperformed NSGA-II and R-NSGA-II for complex scenarios.**

***Index Terms*—collective intelligence; preferences; reference points; evolutionary multi-objective optimization algorithms; facility location**

I. Introduction

Multi-objective optimization problems (MOPs) are problems in which two or more conflicting objective functions must be simultaneously optimized. In the general case, optimization problems and, hence, MOPs, are NP-hard [1]. Therefore, meta-heuristic and/or stochastic approaches are frequently the only viable alternative to handle these problems. The application of evolutionary algorithms to MOPs has prompted the creation of multi-objective optimization evolutionary algorithms (MOEAs) [8].

MOEAs result is a set of points that represent different trade-offs between the objectives. A decision maker (DM) must identify which of those solutions are the ones that satisfy her/his preferences and would be realized in practice. This task can be rather complex and requires in-depth knowledge of the problem being solved, something that is impossible in many practical situations.

When MOPs are particularly complex, instead of approximating the whole set of possible trade-offs, it is convenient to focus the computational efforts on areas of the search space that are of actual practical interest. In this case, the use of a reference point-based approach can aggregate different strategies to drive the search on relevant areas expressed *a priori* or interactively by the decision maker.

The facility location problem [17] is an area of operations research concerned with the localization and assignment of available facilities and resources to achieve the organization's strategic goals. This area has received significant attention due to the number of endeavors that must reduce costs and optimize their operations: manufacturing plants, storage facilities, public transport planning, warehouses, vehicle routing, etc.

Petroleum industry requires an optimal placement and interconnection of extraction and transportation equipment to increase extraction, pumping, and generation of oil, while keeping costs and robustness at optimal levels. Offshore plant operation must balance the competing needs for materials and the application of machinery in an economic way to maximize the different aspects related to operational effectiveness. These circumstances describe a multi-objective optimization and decision-making problem with many stakeholders looking for efficient approaches.

Reference points and interactive techniques can be used to mitigate those inconveniences and support the DM in reaching a proper specification. These approaches allow the optimization algorithm to focus on areas of interest and thus reaching satisfactory solutions at a lower computational cost. The interlace of the search process and DM preferences improves the population quality throughout the evolutionary process and leads to compromise solutions of practical interest.

In real-word optimization problems, it is frequently impossible to define *a priori* reference points or preferences. Collective intelligence (COIN) methods [18] put forward a paradigm that allows to elucidate knowledge from groups of (not necessarily expert) individuals. In this regard, collectively reference points obtained by the interaction and aggregation of multiple opinions can be used to produce an accurate and unbiased representation of preferences and reference points. Built upon the subjectivity of the crowds and human cognition, the intelligence of participatory actions addresses dynamic collective reference points to overcome MOPs difficulties and guide the exploration of preferred solutions.

In previous works [6], [7], we have addressed the issue of how to combine collective intelligence and evolutionary algorithms. In particular, we have incorporated a COIN-based selection operator into three well-known MOEAs. In these cases, COIN was used to generate and progressively update reference points extracted by assembling the suggestions provided by the collective that were used for selecting the best

978-1-7281-6929-3/20/$31.00 ©2020 IEEE

candidate solutions of the MOEA.

In this work, we extend the above results by introducing a novel COIN-based variation operator. The underlying idea here is to submit some of the MOEA candidate solutions to be modified and -hopefully- improved by the collective. The modified solutions are then re-injected to the population and, therefore, go through the rest of the steps of the evolutionary process. This calls for a special problem rendering that allows members of the collective to interact with and modify the solutions. In this case, we reformulate the problem in question in a 'gamified' form.

Besides that, the above mentioned proposals have not yet been compared with other state-of-art reference-point-based MOEAs. This work also provides a better explanation about their operation. That way, the three proposed collective intelligence MOEAs are compared with one another and with respect to the preference-based algorithms: R-NSGA-II [11] and W-HYPE [5].

Experiments with a facility location problem showed the effectiveness of the proposals as they yield better solutions at a lower computational cost. As experiments show, the COIN operator is a competitive advantage as it decreases the number of required function evaluations in the optimization process and provides faster analysis of preferred alternatives only.

Similarly, while working in this task it became evident the lack of adequate performance indicators that take into account preferences. Therefore, this research presents two new performance indicators to evaluate the quality of the approximation set driven by the online collective preferences.

The augmented algorithms implement a collective intelligence operator to bias the search during the optimization phase and restrict the objective space. Besides the four main operators based on the theory of evolution: selection, crossover, mutation and elite-preservation; the COIN operator receives rational collaborations from the participants to improve the overall quality of evolutionary population and generate more appropriated points for DM final choice. There will be presented two types of operators: evaluation and variation operators.

The synergy of actions and the heterogeneity inside collective environments develop creative resolutions based on the crowds' subjectivity and cognition. From a larger perspective, the COIN operator brings a subjective input to the optimization engine and gives a new collective intelligence component to work along the random operators from stochastic methods.

The rest of this paper is organized as follows. Section II covers the theoretical foundations necessary to present the proposal. Section III explains the novel interactive and collective intelligence MOEA. Subsequently, Section IV presents the algorithms based on interactive and collective intelligence techniques. After that, Section V analyse the performance of the algorithms when faced with a facility location case study. Finally, conclusive remarks are put forward.

II. Foundations

Multi-objective optimization problems (MOPs) simultaneously optimize a set of objective functions. Formally posed, a MOP can be defined as $\min \boldsymbol{F}(\boldsymbol{x}) = \{f_1(\boldsymbol{x}), \ldots, f_k(\boldsymbol{x})\}$, where $\boldsymbol{x} = \langle x_1, \ldots, x_n \rangle \in \Omega$ is an n-dimensional vector of decision variables. The solution of a MOP is a (possibly infinite) Pareto-optimal set $P_S = \{\boldsymbol{x} \in \Omega | \nexists \boldsymbol{y} \in \Omega : \boldsymbol{y} \prec \boldsymbol{x}\}$ that contains all the elements of Ω that not Pareto-dominated ($\prec$) by any other element. Elements of P_S represent different trade-offs between the objective functions values. The projection of P_S through $F()$ is known as the Pareto-optimal front, P_F.

A subset X of $\mathbb{R}^n$ is convex if for any two pair of solutions $x^1, x^2 \in X$ and $\alpha \in [0,1]$, the following condition is true: $\alpha x^1 + (1-\alpha)x^2 \in X$. The intersection of all the convex sets containing a given subset X of $\mathbb{R}^n$ is called the convex hull of X. The convex hull of a set of points is the smallest convex set that contains the points.

The convex hull of a set of points in n-dimensional space can be represented as a set of bounding facets and a collection of vertexes for each facet. Convex hull is a well-known geometric object widely used in various fields such as: collision detection, shape analysis, pattern recognition, geographical information systems, image processing, etc.

Let x_i^* be the global minimizers of $f_i(\boldsymbol{x})$, $\forall i \in \{1, \ldots, k\}$. Let $F_i^* = F(x_i^*), \forall i \in \{1, \ldots, k\}$; and Φ be a pay-off matrix k x k whose the i^{th} column is $F_i^* - F^*$. The *convex hull of individual minima* (CHIM) [9] is the set of points in $\mathbb{R}^k$ that are convex combinations of $F_i^* - F^*$:

$$\mathcal{H} = \left\{ \Phi\beta : \beta \in \mathbb{R}^k, \sum_{i=1}^{k} \beta_i = 1, \beta_i \geq 0 \right\}, \qquad (1)$$

A. Preference-Based Interactive Algorithms

Since the decade of 80's, there have been several works on interactive multi-objective methods using reference points and reference directions as preferences. Those approaches were applied mainly in the classical multi-objective programming field. But in the last 15 years they have also emerged in evolutionary multi-objective area.

The reference point approach [23] concentrates the search of Pareto non-dominated solutions in the vicinity of a set of selected preference points. It is based on the achievement scalarizing function that uses a reference point to capture the desired values of the objective functions. Let $\boldsymbol{z}^0$ be a reference point for an n-objective optimization problem of minimizing $F(\boldsymbol{x}) = \{f_1(\boldsymbol{x}), ..., f_k(\boldsymbol{x})\}$, the reference point scalarizing function can be stated as follows:

$$\sigma\left(\boldsymbol{z}, \boldsymbol{z}^0, \boldsymbol{\lambda}, \rho\right) = \max_{i=1,\ldots,k} \left\{\lambda_i (z_i - z_i^0)\right\} + \rho \sum_{i=1}^{k} \lambda_i \left(z_i - z_i^0\right), \qquad (2)$$

where $\boldsymbol{z} \in \mathcal{Z}$ is one objective vector, $\boldsymbol{z}^0 = \left\langle z_1^0, ..., z_k^0 \right\rangle$ is a reference point vector, σ is a mapping from $\mathbb{R}^k$ onto $\mathbb{R}$, $\boldsymbol{\lambda} = \langle \lambda_1, ..., \lambda_k \rangle$ is a scaling coefficients vector, and ρ is an arbitrary small positive number. Therefore, the achievement problem can be rebuilt as: $\min\ \sigma\left(\boldsymbol{z}, \boldsymbol{z}^0, \boldsymbol{\lambda}, \rho\right)$.

Deb et al. [11] proposed a reference-point-based NSGA-II procedure (R-NSGA-II) to find a set of solutions in the neighborhood of the corresponding Pareto optimal. The synchronous R-NSGA-II [14] is a similar approach, but uses three different scalarizing functions to bias the selection operator. The Light Beam Search based EMO [10] modified the NSGA-II crowding operator by the light beam search to incorporate *a priori* preferences and produce a set of solutions in the region of interest. In many-objective optimization problems, the NSGA-III [12] uses reference points on a hyperplane to overcome problems with selection pressure for non-dominated solutions. W-HYPE [5] applied the weighted hypervolume indicator in an interactive fashion to change the optimization goal of the algorithm. On the other hand, DF-SMS-EMOA [21] maps the objectives to desirability functions normalized in the domain $[0,1]$, $d : Y \rightarrow [0,1]$. Then, values of different objectives and units become comparable. Finally, iMOEA/D [15], an interactive version of the decomposition based MOEA, asks the DMs to analyze some current solutions and use their feedback to renew the preferred weight region in the following optimization.

B. Performance Indicators

Many performance indicators have been proposed to evaluate the quality of MOEA outcomes. The performance of algorithms are usually measured on the following features: closeness of the approximation set to the Pareto-optimal front; the diversity and the spread of the points; the volume of the objective space dominated by the solutions.

In this regard, the hypervolume or S-metric indicator [13] calculates the volume of the union of hypercubes a_i defined by a non-dominated point m_i and a reference point x_{ref} defined as

$$\begin{aligned} S(M) &= \Lambda(\{\bigcup_i a_i | m_i \in M\}) \\ &= \Lambda(\bigcup_{m \in M} \{x | m \prec x \prec x_{ref}\}) . \end{aligned} \tag{3}$$

It is a quantitative metric that computes the region space covered by all non-dominated points. This performance indicator does not require knowledge of the true Pareto-optimal front on beforehand, which is an advantage for real-world problems. But the main disadvantage is the computational cost which grows exponentially on the number of objectives.

The Pareto-optimal front coverage indicator, $D_{S \rightarrow P_F}$, is a proximity indicator that defines the distance between an achieved approximation set S and their closest counterpart in the current Pareto-optimal front:

$$D_{S \rightarrow P_F}(S) = \frac{1}{|S|} \sum_{\boldsymbol{x} \in S} \min_{\boldsymbol{x'} \in P_S} \{d(\boldsymbol{x}, \boldsymbol{x'})\} , \tag{4}$$

where d is the Euclidean distance between two points. Small values of $D_{S \rightarrow P_F}$ indicate proximity to the P_F.

Some indicators, like the coverage of two sets [25], map the percentage of domination from one set to another in the interval $[0,1]$. Other measures try to capture the diversity in the pool of final solutions. The spread indicator (Δ) provides information related to the uniformity of the distribution of the obtained approximation set.

C. Collective Intelligence

MOEAs usually confront difficulties with complex, high-dimension and large problem space. The potential number of objectives necessary to describe the environment or the incapacity to comprehend and map all the variables to a correct fitness function can prevent a solution in a reasonable time and quality. On the other hand, human beings are used to multi-objective situations in their everyday lives. Those complex scenarios that are hard for computer might be easier or natural for the human mind. Persons are able to improve the multi-objective algorithms with cognitive and subjective evaluation to find better solutions.

COIN can contribute to make MOEAs go beyond their current reach. Human characteristics such as perception, spatial reasoning, strategy, weighting factors, among others subjectivities might be introduced into the algorithm to generate a better pool of answers and enhance the optimization process. A group of people can understand conflicting situation involving multiples objectives and may use their collective intelligence to trump expert's abilities. The wisdom arisen from the diversity of many individuals is able to discover creative resolutions.

There are plenty of examples that promote the collaboration of many participants to achieve better outcomes than individual efforts. The Amazon Mechanical Turk site outsources digital tasks that are difficult for computers, but not for humans, such as: tagging images, writing product descriptions, identifying performers on music and so on. Foldit is a puzzle game about protein folding. It uses the human brain's natural three-dimensional pattern matching to solve the problem of protein structure prediction and has already helped to decipher the crystal structure of the Mason-Pfizer monkey virus retroviral protease [16]. The free and easy-to-use application VizWiz [4] recruits web volunteers to help blind and visually impaired people with recorded questions or photos about text labels, colors or objects. Xprize stimulates prize competitions on subjects like: life science, energy, climate change and education; to encourage a global collectivity to invest their intellectual capital on difficult problems.

Instead of a good spread of solutions along P_F, the method proposed in this work wants to obtain subsets of solutions close to the collective reference point. In this context, a small cluster variance means the individuals from the sample $Y = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N\}$ are clustered closely around the population mean (μ) or the reference point ($\boldsymbol{z}^0$). A low dispersion for a group of preferred points in P_F denotes a better efficiency of the approach tested. The referential cluster variance indicator κ is represented as follows:

$$\kappa = \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{y_i} - \mu)^2 \tag{5}$$

In cases with more than one collective reference point ($\boldsymbol{z}^j$), the points are clustered based on the closest distance to one of the reference points

$$C_j = \{\boldsymbol{a} \in \mathbb{R}^k : \|\boldsymbol{a} - \boldsymbol{z}^j\| \leq \|\boldsymbol{a} - \boldsymbol{z}^i\|, \forall i\}\,. \qquad (6)$$

Cluster C_j consists of all points for which $\boldsymbol{z}^j$ is the closest. The referential cluster variance is calculated to each cluster separately.

There are a few multi-objective evolutionary approaches designed to work with this geometric concept in the optimization. The convex hull can be applied as a geometric ranking procedure for non-dominated comparisons [22] or a mechanism to discover non-dominated solutions by projecting elements of the CHIM $\mathcal{H}$ towards the boundary $\partial\mathcal{Z}$ of the objective space $\mathcal{Z}$ through an normal vector N [19].

The convex hull volume method can be extended to measure the quality of the non-dominated points in the desired region of interest. The idea behind this is to combine the points around each reference point to form a convex facet of the P_F preferred area. Thereafter, the volume of the convex hull is calculated and used as a scalar indicator for the distribution of points in P_F. Small values of the hull volume (Ψ) indicate concentrate points around the reference points. The *quickhull* method [2] uses a divide and conquer approach similar to *quicksort*. It has the average case complexity of $O(n \log n)$.

III. Collective Intelligence in Evolutionary Algorithms

While some MOEAs techniques construct a partial order of preferences based on *a priori* reference points to give a stronger selection pressure among Pareto-equivalent solutions, others progressive methods combine simultaneously the preferences information and the search for solutions.

Few MOEAs consider more than one user for reference point selection or evolutionary interaction. A collective scenario where many users could actively interact and take part of the decision process throughout the optimization has yet to be properly addressed. The association of collective intelligence features to multi-objective optimization field raises the understanding of preferences from an individual context to a collective perception. This work presents a collective intelligence operator to bias the search during the optimization phase and restrict the objective space.

The main idea underlying this method is to drive the DM's search towards relevant regions in Pareto-optimal set and, also, promote the usage of COIN as a creative search for new individuals. By means of people's heterogeneity and common sense, the COIN operator iteratively refine the search parameters with rational collaborations to improve the overall quality of evolutionary population. The suggested approach decreases the number of function evaluations, accelerate the convergence and achieve relevant regions of Pareto front at a lower computational cost.

IV. Algorithms

In [7] the authors extended some classical MOEAs: NSGA-II [8], SPEA2 [8] and SMS-EMOA [3]. The continuous evolutionary process of the original methods were transformed into an interactive one and the collective reference points were adopted to drive the search towards relevant regions in Pareto-optimal front. At this time, the main change on the current version of the algorithms is the incorporation of the collective intelligence variation operator. Figure 1 illustrates the three collective intelligence MOEAs.

The new CI-NSGA-II converts the original NSGA-II into an interactive process. The inner while loop runs a certain number of times without interruption. Until the first interaction step with the participants through the *COINcontrib()* procedure, the algorithm uses the standard crowding distance in order to come up with a good spread of the solutions before the COIN operator (*COINselec()*) starts focusing on preferred areas of the search space.

The subroutine *COINcontrib()* suspends the evolution progress and submits some individuals from the current approximation set to the users' collaboration. Collective intelligence is applied in two different manners: a selection operator that compares the individuals and chooses the best candidate, or a variation operator that improves current individuals from the population. Particularly, in this research, the individuals received can be analysed in two different ways: a) a pairwise comparison allows the selection of the best candidate between two or more individuals; b) a dynamic game scenario stimulates the participant's creativity to improve or produce new individuals to be placed back in the population. Both approaches discover online collective reference points with the support of a genuine collective intelligence of many users. This approach allows the collective to act as DMs to choose multiple reference points simultaneously whether they are feasible (deducible from a solution vector) or infeasible points.

In a collective environment contributions come from different individuals. Assuming the Central Limit Theorem the inputs have a distribution that is approximately Gaussian. Therefore, after each collective interaction, the subroutine *EM()* gets the users' collaboration as a Gaussian Mixture model to emulate the evaluation landscape of all participants' preferences.

Finally, the procedure *RefPointDist()* calculates the minimum distance from each point in the population to the nearest collective reference points in Θ. This way, the point near the reference point is favoured and stored in the new population. The *COINselec()* procedure develops a partial order similar to the NSGA-II procedure, but replaces the crowding distance operator by the distance to collective reference points (i_{ref}). The partial order $\prec_c$ between two individuals i and j, for example, prefers the minor domination rank (i_{rank}) if they are from different fronts or otherwise, the one with lower values of reference point distance.

The algorithm CI-SMS-EMOA converts the original SMS-EMOA into an interactive process. The *COINcontrib()* and

```
gen   = max_gen
block = subset_gen
while i < gen do
  while block do
    offs = Crossover(pop)
    offs = Mutation(offs)
    pop = COINselec(pop + offs)
    i++
  end while
  front    = PF(pop)
  contrib = COINcontrib(front)
  pop<-contrib
  Θ = EM(contrib)
  pop = RefPointDist(pop, Θ)
end while
```

a) CI-NSGA-II

```
gen   = max_gen
block = subset_gen
while i < gen do
  while block do
    offs = Crossover(pop)
    offs = Mutation(offs)
    pop = COINselec(pop + offs)
    i++
  end while
  front    = PF(pop)
  contrib = COINcontrib(front)
  Θ = EM(contrib)
  pop = Hyper-RPDist(pop, Θ, S)
end while
```

b) CI-SMS-EMOA

```
gen   = max_gen
block = subset_gen
while i < gen do
  while block do
    arq = COINselec(pop + arq)
    offs = Crossover(arq)
    offs = Mutation(offs)
    pop = offs
    i++
  end while
  front    = PF(pop)
  contrib = COINcontrib(front)
  Θ = EM(contrib)
  pop, arq = RPDist(pop, arq, Θ)
end while
```

c) CI-SPEA2

Fig. 1: The COIN MOEAs pseudocode: CI-NSGA-II, CI-SMS-EMOA, CI-SPEA2.

EM() subroutines have the same purpose and work as the CI-NSGA-II. The selection operation, performed by the *COIN-Selec()* procedure, prefers individuals with minor domination rank (i_{rank}). If they belong to the same front, the one with the maximum contribution to the hypervolume of the set and the closest reference point distance (i_{ref}) is selected and stored in the new population. The *Hyper-RPDist()* calculates the minimum distance to the nearest collective reference points in Θ and sets the hypervolume values of the point.

SPEA2 implements elitism by keeping an external population $\bar{P}_t$ of size N. The archive preserves the best solutions since the beginning of the evolution. In the algorithm CI-SPEA2, subroutine *COINSelec()* computes the strength of all individuals and the non-dominated members are copied to the archive *arq*. The k-th nearest data point used to calculate the original density function was substituted by the collective reference points Θ. If the archive $\mid arq \mid \leq N$, the algorithm chooses the nearest individuals to the collective reference point until the archive size is reached. Otherwise, if $\mid arq \mid > N$, it removes the more distant ones proportionally to the number of individuals in each reference point cluster.

The CI-NSGA-II, CI-SMS-EMO and CI-SPEA2 prioritize the points close to the online collective reference point. The algorithms consume preference information to explore satisfactory solutions for DMs.

V. Petroleum Field Facility Location Problem

The location of operational facilities is a strategic goal for many companies. The petroleum industry must extract oil from resource areas and allocate offshore platforms in such a way that optimizes its operational costs and production capacity. More generally, they transform the management of resources into a multi-objective problem where it must balanced the use of facility capacities to operate in an economic way and maximize the operational performance.

Let μ be the cost of one processing unit, v the productive capacity of one processing unit linked to one resource area, M a set of available positions to processing units, N a set of available positions to resource area and D a distance matrix $(d_{ef})_{nxm}$, where $n \in N$ and $m \in M$. The decision variables are the processing unit c_j ($j \in M$) that assumes 1 if it is placed at position j or 0 otherwise and σ_{ij} that assumes 1 if there is a link between the resource area at position $i \in N$ and the processing unit at position $j \in M$.

The problem is to find a good solution for positioning the processing units according the resource area. It is formally represented as the two following optimization problems:

$$\begin{aligned} &\min \sum_{i=1}^{N}\sum_{j=1}^{M} \sigma_{ij} d_{ij} + \sum_{j=1}^{M} c_j \mu\,, \text{ and} \\ &\max \sum_{i=1}^{N}\sum_{j=1}^{M} \sigma_{ij} v_j\,. \end{aligned} \tag{7}$$

Different constraints from real life and several new interdependencies among the variables will increase the search complexity of this facility location problem. The situation described is a candidate for this experiment due to some reasons: a) as a real-world case example, the objectives and decision variables are meaningful to the group; b) the problem interacts with crowd's cognition and requires a 3D spatial reasoning to avoid natural obstacles in the scenario; c) the users' feedback can be made parallel in synchrony with the evolution of individuals in an evolutionary algorithm; d) incentive engines and gamification can be used to retain the users' interest on the interaction during the optimization.

In this context, the facility location problem was designed as a game where players compete among themselves to obtain points and recognition of success. This is usually intended to increase engagement of players, create gameful and playful user experiences, motivate them and set clear objectives to guide a cooperative or competitive behaviour. The game was implemented in a web-based platform and is open to all public.[1] The game elements were transformed to preserve the sensitive details of the industrial partner. Trucks represent the

[1] http://playcanv.as/p/1ARj738G

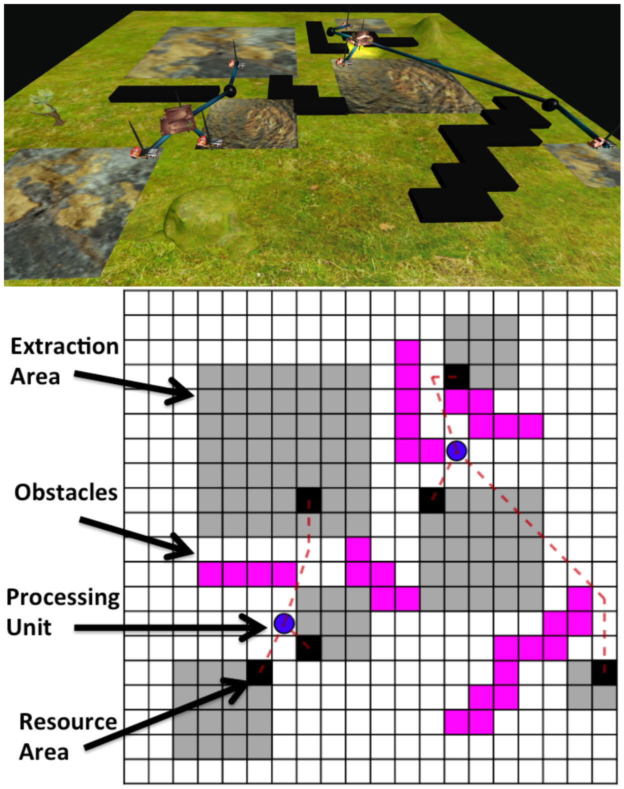


Fig. 2: Game and Computational representation of the facility location problem.

resource areas and the warehouses or barracks symbolize the processing units with two different types.

There are two options in the game: a) pairwise comparison, which implements the selection operator; b) free design mode, which implements the variation operator. In the pairwise comparison mode, the players must vote on the best candidate (individual from population) between two or more facility location scenarios. As votes on the scenarios happen, the Gaussian Mixture model calculates the collective reference point to restrict the search to relevant areas in Pareto front. The players who have chosen the individuals near the collective reference point receive a higher score. They compete at every evolution interval for choices around the collective mean.

In the free design mode, some individuals from population are distributed to the players who have to fix and change their position arrangement. The dynamic game scenario allows the creation of objects like trucks or warehouses, changing their arrangements and rebuilding their connections with straight lines or zigzag lines. This game mode uses the collaboration of people to apply rational improvements in the quality of EA population. Figure 2 shows the dynamic board scene and its internal representation inside the algorithm.

There are four different scenarios available in the game: *easy*, *easy obstacles*, *medium obstacles* and *hard obstacles*. The first one allows only straight lines to connect objects and ignores any sort of obstacles. From second to the fourth scenario, the zigzag lines have to be used to avoid the obstacles. The level of difficulty increases according to the number of obstacles in the game scene. The *medium* and *hard* scenarios are complex and simulate aspects of the real world. The placement of facilities has to consider factors like competition for shared resources and obstructed paths.

According the test results in previous studies [6], [7], CI-NSGA-II outperformed the algorithms CI-SMS-EMOA and CI-SPEA2. Thus, the experiment with a true collectivity compares the new CI-NSGA-II with the original NSGA-II and R-NSGA-II. The main goal is to analyse the performance of CI-NSGA-II facing distinct environments and to identify when the collective intelligence has a positive influence in the results.

In this regard, the original NSGA-II runs independently, whereas the R-NSGA-II uses the ideal point as a fixed reference point. The CI-NSGA-II receives the collective collaboration at specific moments of the optimization process. The *CI-NSGA-II Vote* variation interrupts the evolution process and asks the players to choose the best candidate between two scenarios. In the *CI-NSGA-II Fix* variation, the participants can interactively update and redesign all the elements in the game scene. The *CI-NSGA-II Igni* variation accepts the collective contributions only in the beginning of the evolution (first generation), then it runs to the end without interference. The time interval for human collaboration is 30 seconds for pairwise comparisons and 60 seconds for game scene update.

The experiment was applied in two different computer labs: a Brazilian professional education center with more than 30 students' attendance and a private company training room. Two different methods were used to evaluate performance: fixed distance and fixed time. In the fixed distance, a minimum distance between the current approximation set S and the Pareto-optimal front is measured by the front coverage indicator, $D_{S\to P_F}$. A proximity of $D_{S\to P_F} = 20$ is the criteria to stop the evolution and compare the algorithms. In the fixed time, the algorithms run in a time interval previously defined. Table I presents the results for fixed time and distance evaluation.

Based on the results, the original R-NSGA-II won in the *Easy* and *Easy Obstacles* scenarios. The problem without obstacles is so simple that the algorithm took only two seconds to reach a convergence of $D_{S\to P_F} = 20$. In the case of fixed time (5"), there was not sufficient time to involve a collective participation of users, so the CI-NSGA-II was not applicable (NA).

The *CI-NSGA-II Fix* and *Igni* had a better performance in the *Medium Obstacles* scenario. Although the R-NSGA-II required less number of function evaluations to reach the convergence $D_{S\to P_F} = 20$, the *CI-NSGA-II Igni* results were close and obtained the lowest referential cluster variance indicator κ, which means the points are clustered closely around the collective reference point. *CI-NSGA-II Fix* dominated the values of the fixed time evaluation.

The most interesting result appears in the *Hard Obstacles* scenario. The *CI-NSGA-II Fix* succeeded in all three indicators with the support of the collective intelligence. Considering the

TABLE I: Results of fixed time evaluation.

Algorithms	Easy				Easy Obstacles			
	Time:5”				Time:70”			
	$D_{S \to P_F}$	$Num.Eval.$	κ	Ψ	$D_{S \to P_F}$	$Num.Eval.$	κ	Ψ
NSGA-II	**7,1**	**15.440**	**16,5 e05**	**2590,0**	**16,0**	91.680	16,2 e05	2974,8
CI-NSGA-II Vote	NA	NA	NA	NA	63,6	19.240	22,4 e05	6632,5
CI-NSGA-II Fix	NA	NA	NA	NA	47,0	**8.820**	5,9 e05	1881,6
CI-NSGA-II Ignition	NA	NA	NA	NA	31,6	39.520	**0,7**	**113,0**
	Medium Obstacles				**Hard Obstacles**			
	Time:300”				Time:900”			
NSGA-II	11,2	68.000	18,2 e05	3268,4	57	8.000	17,4 e05	4720,0
CI-NSGA-II Vote	16,5	55.000	22,9 e05	3456,5	76,3	**7.640**	16,6 e05	4089,8
CI-NSGA-II Fix	**1,9**	**29.580**	**2,1 e05**	**1196,5**	**16,2**	8.640	**10,1 e05**	**1109,3**
CI-NSGA-II Ignition	7,7	55.480	7,2 e05	1656,7	28,9	23.300	10,8 e05	1298,2

TABLE II: Results of fixed distance evaluation.

Algorithms	Easy				Easy Obstacles			
	$D_{S \to P_F}$: 20				$D_{S \to P_F}$: 20			
	Time	$Num.Eval.$	κ	Ψ	Time	$Num.Eval.$	κ	Ψ
NSGA-II	**3,0**	3.440	e05	2896,2	**48,6**	62.160	18,2 e05	2889,6
CI-NSGA-II Vote	46,0	3.200	16,5 e05	**2272,3**	196,3	78.520	28,4	451,6
CI-NSGA-II Fix	63,3	1.380	**14,0 e05**	4101,7	86,1	**20.660**	4,3 e05	896,9
CI-NSGA-II Ignition	63,0	**1.280**	14,8 e05	5581,8	106,9	29.680	**2,2**	**10,25**
	Medium Obstacles				**Hard Obstacles**			
NSGA-II	372,4	68.080	17,5 e05	3743,4	2661,9	22.000	18,3 e05	6270,6
CI-NSGA-II Vote	300,0	57.640	25,1 e05	2830,5	3058,7	34.820	17,5 e05	3098,4
CI-NSGA-II Fix	249,5	**25.180**	2,5 e05	1800,0	**823,1**	**4.600**	**7,7 e05**	**1267,3**
CI-NSGA-II Ignition	**179,6**	25.280	**1,9 e05**	**1164,3**	1448,8	44.100	15,02 e05	2890,9

fixed distance evaluation, the algorithm required 5 times less function evaluation and performed 3 times faster than NSGA-II. In terms of the fixed time evaluation, it managed to find a convergence 3.5 times better than R-NSGA-II.

Altogether, *CI-NSGA-II Fix* iteratively refines the search parameters and adopts players collaborations to achieve more appropriated points in the final trade-off set. It encourages the creativity and cognition to produce new solutions. Figure 3 demonstrates how the collective intelligence contributions in *CI-NSGA-II Fix* outperform the regular NSGA-II and R-NSGA-II from the *Medium Obstacles* scenario onwards (low values are desired). This concludes that collective intelligence and reference points enhance the MOEA results when faced with more complex scenarios.

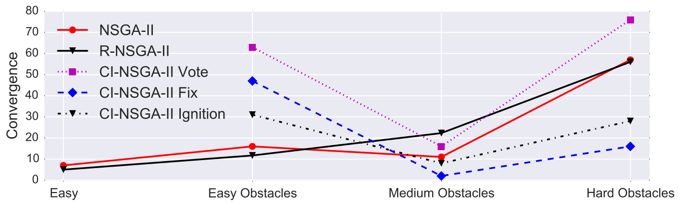
Figure 3c shows the number of function evaluations for each scenario. The *CI-NSGA-II Fix* consistently presented the lowest values of function evaluations. From a practical point of view, the arrangement of facilities involves large sums of capital resources and their economic effects are long term. This approach may allow a fast analysis of many manufacturing alternatives enabling the company to take rapid decisions both in the design and in the operation phases, and to obtain competitive advantages in costs control. The interactive MOEA could benefit from human characteristics, such as 3D spatial reasoning and strategic thinking, to find a handful of preferred solutions and give the company a competitive advantage.

VI. Final Remarks

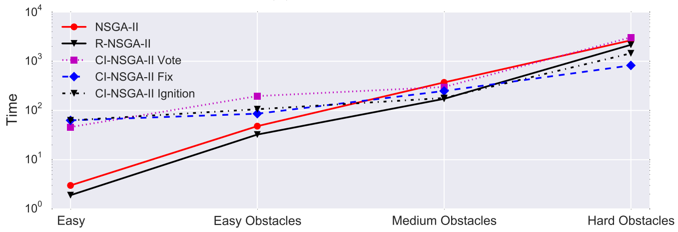
MOEAs can take advantage of decision makers' preferences to guide the search through relevant regions of Pareto-optimal front. Suitable techniques of preference-based and interactive multi-objective algorithms were pointed out as an alternative to handle the dynamism not expected by *a priori* methods.

The presented algorithms apprehend people's heterogeneity and common sense to improve the successive stages of evolution in a direct crowd sourcing fashion. Consequently, instead of the entire front, it reaches a smaller sub-set of the front and uses the collective preferences to support decisions upon multi-objective situations.

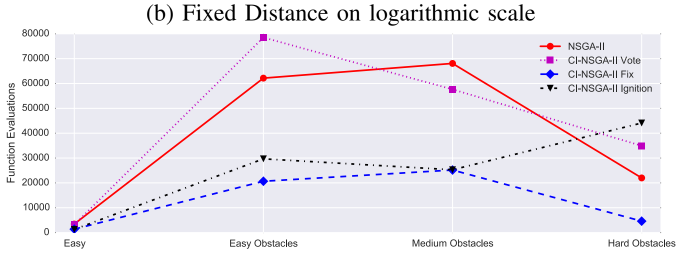
The approaches have been tested successfully in a real-world case study regarding facility location. The continuity of this research will compare with and possibly extend other MOEAs, such as MOEA/D [24] and FEMOEA [20].



(a) Fixed Time



(b) Fixed Distance on logarithmic scale



(c) Number of function evaluations for each scenario.

Fig. 3: Results of distance and time measurements for each game scenario.

Two different performance indicators (referential cluster variance and hull volume) were used with the intention to measure the proportion of occupied area in P_F.

The combination of collective intelligence in MOEAs has an advantage over traditional iterative approaches because their results are driven not by one DM, but a group of people that delimits their collective area of interest and preferences in the objective space.

There is a particular interest in more complex scenarios with many constraints and non-explicit objectives hidden in the problem. It is important to validate if the complexity of the environment will favor even more the integration of COIN in MOEAs. Also, as future work, we plan to create an open platform and others web-scenarios to apply collective intelligence in different multi-objective real-world problems.

References

[1] T. Bäck, *Evolutionary algorithms in theory and practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. NY: Oxford University, 1996.

[2] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software (TOMS)*, vol. 22, no. 4, pp. 469–483, 1996.

[3] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, 2007.

[4] J. P. Bigham, C. Jayant, H. Ji, G. Little, A. Miller, R. C. Miller, R. Miller, A. Tatarowicz, B. White, S. White *et al.*, "Vizwiz: nearly real-time answers to visual questions," in *Proceedings of the 23nd annual ACM symposium on User interface software and technology*. ACM, 2010, pp. 333–342.

[5] D. Brockhoff, J. Bader, L. Thiele, and E. Zitzler, "Directed multiobjective optimization based on the weighted hypervolume indicator," *Journal of Multi-Criteria Decision Analysis*, vol. 20, no. 5-6, pp. 291–317, 2013.

[6] D. Cinalli, L. Martí, N. Sanchez-Pi, and A. C. B. Garcia, "Collective preferences in evolutionary multi-objective optimization: Techniques and potential contributions of collective intelligence," in *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, ser. SAC '15. NY, USA: ACM, 2015, pp. 133–138.

[7] ——, "Bio-inspired algorithms and preferences for multi-objective problems," in *11th International Conference on Hybrid Artificial Intelligence Systems*, ser. HAIS '16, 2016.

[8] C. C. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary algorithms for solving multi-objective problems*. Springer, 2007.

[9] I. Das and J. E. Dennis, "Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems," *SIAM Journal on Optimization*, vol. 8, no. 3, pp. 631–657, 1998.

[10] K. Deb and A. Kumar, "Light beam search based multi-objective optimization using evolutionary algorithms," in *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*. IEEE, 2007, pp. 2125–2132.

[11] K. Deb, J. Sundar, N. Udaya Bhaskara Rao, and S. Chaudhuri, "Reference point based multi-objective optimization using evolutionary algorithms," *International Journal of Computational Intelligence Research*, vol. 2, no. 3, pp. 273–286, 2006.

[12] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach: solving problems with box constraints," *Evolutionary Computation, IEEE Transactions*, vol. 18, no. 4, pp. 577–601, 2014.

[13] M. Emmerich, N. Beume, and B. Naujoks, "An emo algorithm using the hypervolume measure as selection criterion," in *Evolutionary Multi-Criterion Optimization*. Springer, 2005, pp. 62–76.

[14] E. Filatovas, O. Kurasova, and K. Sindhya, "Synchronous R-NSGA-II: An extended preference-based evolutionary algorithm for multi-objective optimization," *Informatica*, vol. 26, no. 1, pp. 33–50, 2015.

[15] M. Gong, F. Liu, W. Zhang, L. Jiao, and Q. Zhang, "Interactive moea/d for multi-objective decision making," in *Proceedings of the 13th annual conference on Genetic and evolutionary computation*. ACM, 2011, pp. 721–728.

[16] F. Khatib, F. DiMaio, S. Cooper, M. Kazmierczyk, M. Gilski, S. Krzywda, H. Zabranska, I. Pichova, J. Thompson, Z. Popović *et al.*, "Crystal structure of a monomeric retroviral protease solved by protein folding game players," *Nature structural & molecular biology*, vol. 18, no. 10, pp. 1175–1177, 2011.

[17] A. Klose and A. Drexl, "Facility location models for distribution system design," *European Journal of Operational Research*, vol. 162, no. 1, pp. 4–29, 2005.

[18] T. W. Malone, R. Laubacher, and C. Dellarocas, "Harnessing crowds: Mapping the genome of collective intelligence," Alfred P. Sloan School of Management, Massachusetts Institute of Management, Cambridge, MA, Tech. Rep., 2009.

[19] S. Z. Martínez and C. A. C. Coello, "An archiving strategy based on the convex hull of individual minima for moeas," in *IEEE Congress on Evolutionary Computation*. IEEE, 2010, pp. 1–8.

[20] J. L. Redondo, J. Fernández, and P. M. Ortigosa, "Femoea: a fast and efficient multi-objective evolutionary algorithm," *Mathematical Methods of Operations Research*, pp. 1–23, 2012.

[21] T. Wagner and H. Trautmann, "Integration of preferences in hypervolume-based multiobjective evolutionary algorithms by means of desirability functions," *Evolutionary Computation, IEEE Transactions on*, vol. 14, no. 5, pp. 688–701, 2010.

[22] P. Wang, M. Emmerich, R. Li, K. Tang, T. Bäck, and X. Yao, "Convex hull-based multiobjective genetic programming for maximizing receiver operating characteristic performance," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 2, pp. 188–200, 2015.

[23] A. P. Wierzbicki, "Reference point approaches and objective ranking," *Practical approaches to multi-objective optimization*, no. 06501, 2006.

[24] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *Evolutionary Computation, IEEE Transactions on*, vol. 11, no. 6, pp. 712–731, 2007.

[25] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *Evolutionary Computation, IEEE Transactions on*, vol. 3, no. 4, pp. 257–271, 1999.