Strategies for Compressing the Pareto Frontier: Application to Strategic Planning of Hydropower in the Amazon Basin

Zhongdi Qu[1](✉), Marc Grimson[1], Yue Mao[1], Sebastian Heilpern[2], Imanol Miqueleiz[2], Felipe Pacheco[2], Alexander Flecker[2], and Carla P. Gomes[1]

[1] Department of Computer Science, Cornell University, Ithaca, USA
{zq84,mg2425,ym277}@cornell.edu, gomes@cs.cornell.edu
[2] Department of Ecology and Evolutionary Biology, Cornell University, Ithaca, USA
{s.heilpern,im298,felipe.pacheco,asf3}@cornell.edu

Abstract. The development of ethical AI decision-making systems requires considering multiple criteria, often resulting in a large spectrum of partially ordered solutions. At the core of this challenge lies the Pareto frontier, the set of all feasible solutions where no solution is dominated by another. In previous work, we developed both exact and approximate algorithms for generating the Pareto frontier for tree-structured networks. However, as the number of criteria grows, the Pareto frontier increases exponentially, posing a significant challenge for decision-makers. To address this challenge, we propose various strategies to efficiently compress the Pareto frontier, including an approximation method with optimality and polynomial runtime guarantees. We provide detailed empirical results on the strategies' effectiveness in the context of strategic planning of the hydropower expansion in the Amazon basin. Our strategies offer a more manageable approach for navigating Pareto frontiers.

Keywords: Multi-objective optimization · Approximation algorithms · Hierarchical clustering

1 Introduction

In recent years, there has been a growing interest in developing AI decision-support systems that can evaluate trade-offs based on multiple criteria, moving away from the conventional single-objective systems. This shift is particularly important when considering more ethical AI decision-making systems that align

This project is partially supported by the Eric and Wendy Schmidt AI in Science Postdoctoral Fellowship, a Schmidt Futures program; the National Science Foundation (NSF); the National Institute of Food and Agriculture (USDA/NIFA); the Air Force Office of Scientific Research (AFOSR); and the Cornell Atkinson Center for a Sustainable Future (ACSF).

B. Dilkina (Ed.): CPAIOR 2024, LNCS 14743, pp. 141–157, 2024.
https://doi.org/10.1007/978-3-031-60599-4_9

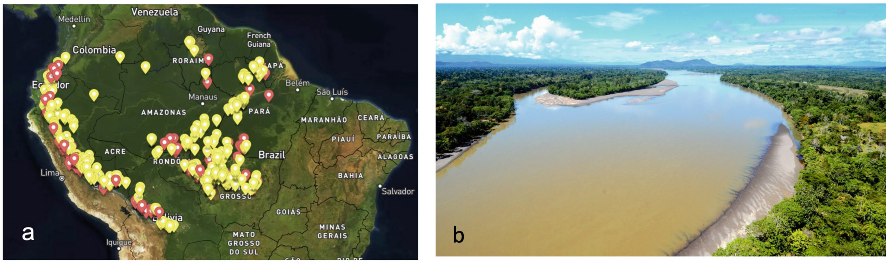


Fig. 1. (a) Existing (red) and proposed (yellow) hydropower dams in the Amazon basin and (b) Rio Santiago, a free-flowing river in the Andean Amazon with large hydropower dams in planning stages (Alvaro del Campo/The Field Museum). (Color figure online)

with multiple human values [24]. It is especially crucial to consider multiple criteria in computational sustainability [13], where balancing economic, environmental, and societal objectives is essential for achieving the Sustainable Development Goals (SDGs) [23].

Multi-objective optimization is computationally challenging. At the core of this challenge lies the Pareto frontier: the set of solutions in a multi-dimensional space representing the trade-offs among different potentially conflicting objectives. In other words, when optimizing for multiple objectives, the result is often a large spectrum of partially ordered solutions. The Pareto frontier is therefore the set of solutions that are not dominated by any other solution. Our previous work focused on developing both exact and approximate algorithms to compute the Pareto frontier in tree-structured networks [1,3,12].

Our research has been motivated by the need for strategic planning of hydropower expansion in the Amazon basin. Hydropower plays a critical role in current and future renewable energy strategies globally. The variation in project sizes and the diverse characteristics of river systems highlight the need for a deeper understanding of the trade-offs between hydropower capacity and ecosystem services. This understanding is key in evaluating dam portfolios across the Amazon river network, where hydropower projects have been proposed at over 350 locations (Fig. 1). Multicriteria optimization is crucial in identifying dam portfolios that balance social-environmental costs with energy production. However, our multiobjective optimization approaches often yield solutions consisting of millions of portfolios. As the number of criteria increases, the Pareto frontier grows exponentially, presenting a substantial challenge for decision-makers. The disparity between our computational approaches producing a vast number of Pareto-optimal solutions and the practical needs of decision-making in dam expansion is a significant hurdle for policymakers striving to construct dams with minimal environmental impact while achieving energy goals. Therefore, innovative approaches that effectively compress the number of optimal Pareto portfolios are critical to finding practical and realistic solutions.

Our Contributions: To facilitate navigating the Pareto frontier, herein we propose different approaches: **(1) A representation of the Pareto frontier,** which consists of a subset of solutions from the Pareto frontier with coverage guarantees and can be generated in time polynomial in the size of the frontier; **(2) An approximation of the Pareto frontier,** based on a dynamic-programming-based strategy, with optimality guarantees and polynomial runtime guarantees; and **(3) An estimation of the Pareto frontier,** based on a dynamic-programming-based strategy, with optimality guarantees, without polynomial runtime guarantees, but with good empirical performance. **(4)** We provide detailed **empirical results** analyzing the trade-offs of our different strategies against various baselines, **in the context of strategic planning of the hydropower expansion in the Amazon basin.** Our strategies offer more manageable ways for navigating Pareto frontiers.

2 Related Works

To solve unstructured multi-objective optimization problems, genetic algorithms have been widely used, including the family of Non-dominated Sorting Genetic Algorithms (NSGA [22], NSGA-II [8], and NSGA-III [7]) and Multi-objective Evolutionary Algorithm Based on Decomposition (MOEA/D) [28]. However, when it comes to problems with an underlying structure, like the tree-structured river network for the planning of hydropower dams, these algorithms usually are not competitive with algorithms that take advantage of that structure [26]. Moreover, genetic algorithms rarely provide theoretical guarantees on optimality or runtime: so far the theoretical analysis of these algorithms has been restricted to relatively simplistic and few objectives [9–11, 29].

Our work fits into a series of research that exploits the underlying tree-structured river network to approximate the Pareto frontier for the planning of hydropower in the Amazon basin. [26] first proposed the dynamic-programming-based algorithm to find the exact Pareto frontier and a fully polynomial time approximation scheme (FPTAS) to approximate the frontier. Following works [3, 14, 15] further improved the methods through techniques including divide-and-conquer, expansion, compression, and affine transformation. The methods we propose here can be fully incorporated into the developed approaches.

Our methods employ hierarchical clustering techniques. The idea of leveraging clustering in multi-objective optimization to improve algorithm performance or to help interpret the Pareto frontier has been explored before, but mostly in the context of genetic algorithms. In [6, 19, 27, 30] clustering algorithms, including hierarchical clustering, have been used to discover the population structure and aid in parent selection and offspring retention. Clustering helps with discovering solutions that are distributed more widely and uniformly.

Our work relates to Binary Decision Diagrams (BDDs) [4]. To solve a multi-objective discrete optimization problem, the BDD method uses decision diagrams to represent exactly the feasible set of the problem and then uses a multicriteria shortest path algorithm for finding the set of non-dominated solutions [4]. However, the size of the diagram could grow exponentially. Approximate decision diagrams that have a polynomial limit on the size have been developed [2]. A crucial difference between this method and ours is that BDD assumes linear separability in the objective functions, whereas, for our problem domain, the objective functions are non-separable and are dependent upon all decisions.

3 Preliminaries

3.1 Multi-objective Optimization

A multi-objective optimization problem consists of optimizing several often conflicting objectives simultaneously. Therefore, typically there does not exist a single solution that optimizes all the objectives at the same time. Accordingly, we give the definitions of optimality in the multi-objective scenario.

Pareto Dominance. Without loss of generality, assume we are maximizing d objectives at the same time. For a solution π, $z(\pi) = (z^1(\pi), \ldots, z^d(\pi))$ is the values of the d objectives. A solution π dominates another solution π', written as $z(\pi) \succ z(\pi')$, if and only if for all $1 \leq i \leq d, z^i(\pi) \geq z^i(\pi')$, and there exists $1 \leq j \leq d$ such that $z^j(\pi) > z^j(\pi')$.

Pareto Frontier. Our goal in the multi-objective optimization problem is to find the set of non-dominated solutions, which we define to be the Pareto frontier: let $\mathcal{S}$ be the set of all feasible solutions, the Pareto frontier is $\{\pi \in \mathcal{S} | z(\pi) \nprec z(\pi'), \forall \pi' \in \mathcal{S}\}$.

In practice, the size of the Pareto frontier may be exponential even for a fixed number of objectives. As a result, finding or interpreting the entire frontier may be computationally expensive. Therefore, a more realistic goal is to find a good approximation or representation of the Pareto frontier.

ϵ-approximation. Given a Pareto frontier P, a set of solutions S ϵ-approximates P if and only if for every $\pi \in P$, there exists a solution $\pi' \in S$ such that $z^i(\pi') \geq (1-\epsilon)z^i(\pi)$ for all $1 \leq i \leq d$, and S is found in polynomial time.

Note that an ϵ-approximation is found in polynomial time. When a set with such optimality guarantee is found in superpolynomial time, we call it an ϵ-*estimation.*

γ-representation. Given a Pareto frontier P, a subset of the frontier $P' \subseteq P$ γ-represents P if and only if for every $\pi \in P$, there exists a solution $\pi' \in P'$ such that $z^i(\pi') \geq (1-\gamma)z^i(\pi)$ for all $1 \leq i \leq d$.

Note that a crucial difference between ϵ-approximation/estimation and γ-representation is that, while a solution in the ϵ-approximation/estimation set is not necessarily Pareto-optimal, a solution in the γ-representation is always Pareto-optimal since the γ-representation set is a subset of the Pareto frontier.

3.2 Strategic Planning of Hydropower in the Amazon Basin

The Problem. Construction of hydropower dams provides electricity but can cause significant adverse environmental impacts including disruption of fish

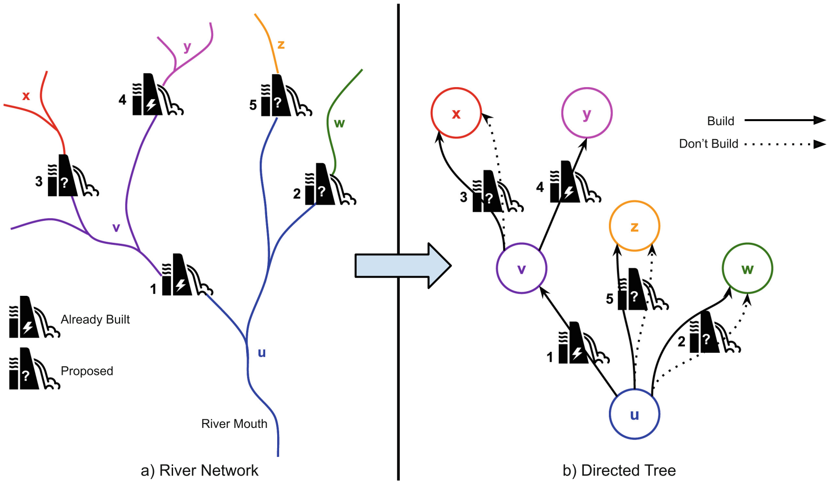


Fig. 2. Converting (a) a river network to (b) a directed multi-edged tree. Contiguous sections of the river uninterrupted by dam sites become nodes in the tree. Dam sites, both proposed and already built, are edges that connect upstream and downstream portions of the river, with a unique edge per decision.

migration routes and greenhouse gas emissions. Planning of the hydropower dam placements requires the balancing of energy production and ecosystem impacts. Accordingly, given a set of proposed dam sites, a solution is a subset of the dams to be built and our goal is to find a set of solutions that approximates or represents the Pareto frontier with respect to the following 6 objectives: **(1)** hydropower generation, **(2)** connectivity (the total length of the un-obstructed stream segments that a fish can travel starting from the river mouth without passing any dam site), **(3)** sediment (the amount of sediment and nutrients transported to the river mouth every year considering the fact that each dam traps a certain percentage of total sediment from upstream), **(4)** biodiversity (the overall impact on the fish population caused by dam construction), **(5)** degree of regulation (the total degree of flow regime alteration caused by dam construction), and **(6)** greenhouse gas emissions (the total greenhouse gas emissions caused by dam construction).

The Algorithm. Previous works [14,15,26] model the river network as a multi-edged directed tree structure (see Fig. 2). In the multi-edged directed tree representation, each edge represents a possible decision at a potential dam site, and its two vertices are respectively the river regions directly upstream and downstream of the site. Thus, each pair of parent/child nodes may have one or more edges depending on the number of decisions relevant to a given dam location. Every node v in the tree is associated with a non-negative node reward r_v^i for each objective i. Each edge is represented by (u, v, j) with parent and child nodes u and v and index j to distinguish the edge from sibling edges. Additionally, each edge is associated with a non-negative edge reward s^i_{uvj} and a non-negative transfer coefficient p^i_{uvj}, for each objective i. A solution (or partial solution) is defined as a spanning tree of the multi-edged tree (or partial spanning tree of a sub-tree). The i-th objective value of a partial solution at a leaf node v, π_v is its corresponding reward, i.e., $z^i(\pi_v) = r^i_v$. The i-th objective value of a partial solution at a non-leaf node u, π_u is defined recursively:

$$z^i(\pi_u) = r^i_u + \sum_{(u,v,j)\in\pi} s^i_{uvj} + p^i_{uvj} z^i(\pi_v) \tag{1}$$

Based on the tree-structure formulation, [26] proposed a dynamic programming algorithm that can find the exact Pareto frontier, based on the crucial observation, proven in [26], that

Theorem 1. *Let u be a node in the tree and $u_1, \ldots, u_k$ be its children. Any Pareto-optimal partial solution at u can be constructed by combining one Pareto-optimal partial solution from child u_i for each $i \in [1, \ldots, k]$ and the choice of edges connecting u and u_i.*

As a result, the algorithm recursively computes the Pareto-optimal partial solutions from leaf nodes to the root. At each node u, the algorithm comes up with the candidate solutions by combining the Pareto-optimal partial solutions at u's children. Then, the algorithm discards any dominated solutions to obtain the Pareto-optimal partial solutions at u.

Given that the size of the frontier could be exponential, they also proposed a fully polynomial-time approximation scheme (FPTAS) that approximates the Pareto frontier within an arbitrarily small ϵ and runs in time polynomial in the size of the instance and $1/\epsilon$. The FPTAS introduces a hyperparameter $K^i_u = \epsilon r^i_u$ for each node u and each objective i, and defines the rounded objective value $\hat{z}^i(\pi_u)$ recursively as

$$\hat{z}^i(\pi_u) = r^i_u + \left\lfloor \frac{\sum_{(u,v,j)\in\pi} s^i_{ujv} + p^i_{uvj}\hat{z}^i(\pi_v)}{K^i_u} \right\rfloor K^i_u. \tag{2}$$

In [26], it was proven that the Pareto frontier on tree-structured networks can be ϵ-approximated, namely:

Theorem 2. *Let P_s be the set of (partial) Pareto-optimal solutions for a node s and $\hat{P}_s$ be the set of (partial) Pareto-optimal solutions computed via the dynamic programming algorithm using the rounded objective function 2. We must have $\hat{P}_s$ is an ϵ-approximation of P_s.*

4 A Representation of the Pareto Frontier

Given a Pareto frontier P, the problem of finding the γ-representation of P is to find a subset P' of P such that for every solution $\pi \in P$, there exists a solution $\pi' \in P'$ such that $z^i(\pi') \geq (1-\epsilon)z^i(\pi)$ for all $1 \leq i \leq d$. To this end, we have designed an algorithm based on hierarchical clustering [16,20]. Hierarchical clustering takes a set of data points and seeks to build a hierarchy of clusters of the data points. It has been widely applied to fields including taxonomy [21], bioinformatics [17,25], and social network analysis [18]. The agglomerative version of the algorithm starts with each data point as a separate cluster, and pairs of clusters are greedily merged as one moves up in the hierarchy. To decide which clusters should be combined, a measure of distance between sets of data points is required. Typically, this measure includes a distance metric between single points of the data set and a linkage method that specifies the distance of two sets as a function of the pairwise distances between the data points across the two sets. What distance metric to use depends on the underlying application, and some examples include the Euclidean distance and the Hamming distance. The linkage method, on the other hand, influences the shape of the clusters. For example, complete linkage, i.e., the distance of two sets is the maximum distance between any two data points across the sets, tends to produce more spherical clusters than single linkage, where the minimum distance is used. For our case, Euclidean distance between the objective values normalized to $[0, 1]$, and average linkage, i.e., the distance of two sets is the average distance between the pairs of data points across the sets, are used.

The algorithm to find γ-representation

- Input: A Pareto frontier $P = \{\pi_1, \ldots, \pi_n\}$, and a parameter γ.
- Output: A subset of the Pareto frontier $P' \subseteq P$ such that $\forall \pi \in P$ there is a $\pi' \in P'$ such that $z^i(\pi') \geq (1-\gamma) z^i(\pi)$ for all $1 \leq i \leq d$.

1. Perform hierarchical clustering on P:
   (a) For each objective, normalize the objective values to $[0, 1]$.
   (b) Initialize $C_1 = \{\pi_1\}, C_2 = \{\pi_2\}, \ldots, C_n = \{\pi_n\}$, and $\mathcal{C} = \{C_1, \ldots, C_n\}$.
   (c) Find the two clusters in $\mathcal{C}$, C_i and C_j, with the smallest distance, as defined by Euclidean distance and average linkage, among all pairs of clusters.
   (d) $C_{|\mathcal{C}|+1} = C_i \cup C_j$, $\mathcal{C} = \mathcal{C} \cup \{C_{|\mathcal{C}|+1}\} - \{C_i\} - \{C_j\}$.
   (e) Repeat steps $1(c)$ to $1(d)$ until $\mathcal{C} = \{P\}$.
2. Run Algorithm 1 on the final cluster $\{P\}$.

Theorem 3. *The runtime of the algorithm to find γ-representation on a Pareto frontier P with $|P| = n$ is $O(n^3)$ and the algorithm returns a set $P' \subseteq P$ such that for every solution $\pi \in P$, there exists a solution $\pi' \in P'$ such that $z^i(\pi') \geq (1-\gamma) z^i(\pi)$ for all $1 \leq i \leq d$.*

Proof. The for loop from line 3 to line 11 of Algorithm 1 makes sure that Coverage($\{P\}$) is indeed a γ-representation of P. The time complexity for doing hierarchical clustering on P is $O(n^3)$. The hierarchy of clusters can be represented as a binary tree where a node u having children l and r means clusters l and r are merged to form cluster u. The leaves of this binary tree are the individual solutions in P. Therefore, the size of the tree is $2n-1$. In the worst case, Coverage($\{P\}$) will traverse every cluster in the tree and look at every solution in the clusters. Thus, Coverage($\{P\}$) runs in $O(n^2)$.

Algorithm 1: Coverage

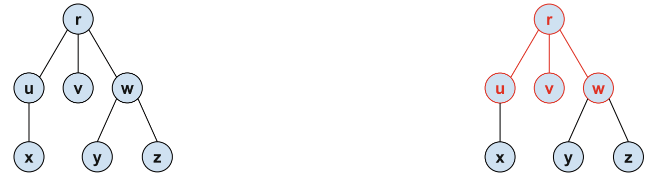
```
Data: A cluster of solutions $C = C_i \cup C_j$ where $C_i$ and $C_j$ have been
      merged in the hierarchical clustering process to form $C$.
Result: A subset $C'$ of $C$ such that for all $\pi \in C$ there is a $\pi' \in C'$ such
        that $z^i(\pi') \geq (1-\gamma)z^i(\pi)$ for all $1 \leq i \leq d$
$\pi' \leftarrow rand(C)$ ;    /* $rand(C)$ returns a random sample from $C$. */
$failed \leftarrow$ False;
foreach $\pi \in C$ do
    for $1 \leq i \leq d$ do
        if $z^i(\pi') < (1-\gamma)z^i(\pi)$ then
            $failed \leftarrow$ True;
            break;
        end
    end
    if $failed$ then break ;
end
if $failed$ then return Coverage($C_i$) $\cup$ Coverage($C_j$) ;
else return $\{\pi'\}$ ;
```

5 An Estimation of the Pareto Frontier

The method described in Sect. 4 works as a post-processing step after the Pareto frontier has been discovered. Alternatively, we consider incorporating the representation method into the dynamic programming algorithm proposed in [26] to estimate the Pareto frontier. The algorithm models the river network as a tree structure and recursively computes the Pareto-optimal partial solutions from the leaf nodes to the root of the tree. At every node, candidate solutions are formed by combining the Pareto-optimal partial solutions at the node's children. Dominated partial solutions are then discarded. We argue that if we apply the γ-representation algorithm after the pruning of the dominated solutions at some levels of nodes in the tree, then we have an estimation of the Pareto frontier, with optimality guarantees, but not necessarily polynomial runtime guarantees.

Note that for a tree T_u rooted at node u, we call the level of node u level 1, the level of u's children level 2, etc. We apply the γ-representation algorithm to L levels of the nodes in T_u, which means that to all the nodes from level L to level 1, after the dominated partial solutions have been discarded, we run the algorithm to find the γ-representation of the Pareto-optimal partial solutions and use the representation set, instead of all the Pareto-optimal partial solutions, to assemble the solutions at the node's parent (Fig. 3b).

Theorem 4. *Consider a node u in a run of the dynamic programming algorithm proposed in [26], and the subtree rooted at u, T_u. Suppose to L levels of the nodes in T_u we apply the γ-representation algorithm, then at node u, we obtain a set of solutions S_u such that for every Pareto-optimal partial solution at u, π_u, there is a solution $\bar{\pi}_u \in S_u$ such that $z^i(\bar{\pi}_u) \geq (1-\gamma)^L z^i(\pi_u)$ for all $1 \leq i \leq d$.*



(a) tree structure

(b) 2 levels of the nodes

Fig. 3. Strategy for applying the γ-representation on the tree. (a) shows the underlying tree structure, with (b) showing the representation applied to 2 levels of the nodes. Nodes in red are nodes that have the γ-representation applied. (Color figure online)

Proof. We prove the theorem by induction. For $L = 0$ and $L = 1$, the statements are direct results of Theorems 1 and 3 respectively. Suppose the statement is true for $L' = L - 1 \geq 1$, and consider a Pareto-optimal partial solution π_u at u. By Theorem 1 and Eq. 1, we have that for all $1 \leq i \leq d$,

$$z^i(\pi_u) = r_u^i + \sum_{(u,v,j)\in\pi_u} s_{uvj}^i + p_{uvj}^i z^i(\pi_v)$$

where π_v is a Pareto-optimal partial solution at node v. By the induction hypothesis, the algorithm has found at v a set of partial solutions S_v that includes a solution π'_v such that $z^i(\pi'_v) \geq (1-\gamma)^{(L-1)} z^i(\pi_v)$ for all $1 \leq i \leq d$. By substituting π_v with π'_v, and combining all the π'_v with the same edges as in π_u, we obtain a partial solution π'_u such that for all $1 \leq i \leq d$

$$z^i(\pi'_u) = r_u^i + \sum_{(u,v,j)\in\pi_u} s_{uvj}^i + p_{uvj}^i z^i(\pi'_v) \geq (1-\gamma)^{(L-1)} z^i(\pi_u)$$

When the dynamic programming algorithm prunes the solutions at node u, either π'_u is kept as a Pareto-optimal partial solution, or π'_u is discarded because the algorithm has found a π''_u such that for all $1 \leq i \leq d$, $z^i(\pi''_u) \geq z^i(\pi'_u) \geq (1-\gamma)^{(L-1)} z^i(\pi_u)$. Either way, after pruning at node u, we must have kept a partial solution $\tilde{\pi}_u$ such that $z^i(\tilde{\pi}_u) \geq (1-\gamma)^{(L-1)} z^i(\pi_u)$. Then when we apply the algorithm to find the γ-representation at node u, by Theorem 3, we will be guaranteed to have in the representation set S_u a partial solution $\bar{\pi}_u$ such that for all $1 \leq i \leq d$, $z^i(\bar{\pi}_u) \geq (1-\gamma) z^i(\tilde{\pi}_u) \geq (1-\gamma)^L z^i(\pi_u)$.

Applying Theorem 4 to the root node, we obtain the following Lemma.

Lemma 1. *Suppose during a run of the dynamic programming algorithm proposed in [26], the algorithm to find the γ-representation is applied to L levels of the nodes, then at the root node we obtain a set S such that for every Pareto-optimal solution $\pi \in P$, there exists a solution $\pi' \in S$ such that for all $1 \leq i \leq d$, $z^i(\pi') \geq (1-\gamma)^L z^i(\pi)$.*

6 An Approximation of the Pareto Frontier

The algorithm described in Sect. 5 finds an estimation of the Pareto frontier with optimality guarantee, but the algorithm could run in exponential time since at each node there are a potentially exponential number of partial solutions to consider. To mitigate that problem, [26] has applied a rounding technique to the exact dynamic programming algorithm and the result is an FPTAS that ϵ-approximates the Pareto frontier P. We argue that if we apply the γ-representation algorithm to some levels of the nodes in the FPTAS, we obtain a further compressed approximation of the Pareto Frontier, with optimality and polynomial runtime guarantees.

Theorem 5. *Consider a node u in a run of the FPTAS proposed by [26] with parameters ϵ and $K_v^i = \epsilon r_v$. If for L levels of the subtree T_u rooted at u, we apply to the nodes the γ-representation algorithm in the rounded objectives, and the parameters L, γ, and ϵ satisfy that if $L > 1$, then $(1-\gamma)^{L-1} + \epsilon \leq 1$, then at u we obtain a set S_u such that for every Pareto-optimal partial solution at u, π_u, there is a solution $\bar{\pi}_u \in S_u$ such that for all $1 \leq i \leq d$, we have $z^i(\bar{\pi}_u) \geq (1-\gamma)^L(1-\epsilon)z^i(\pi_u)$.*

Proof. We first prove again by induction that Theorem 4 still holds for the rounded objectives. The base cases for $L = 0$ and $L = 1$ are direct consequences of Theorem 2 and Theorem 3. We prove the induction step where $L > 1$. Consider the root node u and its children again. Write $\sum_{(u,v,j)\in\pi_u} s_{uvj}^i + p_{uvj}^i \hat{z}^i(\pi_v)$ as $NK_u^i + R$ for some non-negative integer N and some non-negative real number $R < K_u^i$. Then by Eq. 2 we have

$$\hat{z}^i(\pi_u) = r_u^i + \left\lfloor \frac{NK_u^i + R}{K_u^i} \right\rfloor K_u^i = r_u^i + NK_u^i$$

Similarly as in the proof for Theorem 4, by substituting all the π_v's with their γ-representations and choosing the same edges between u and v, we obtain a partial solution at u, π_u' such that

$$\hat{z}^i(\pi_u') \geq r_u^i + \left\lfloor \frac{(1-\gamma)^{L-1}(NK_u^i + R)}{K_u^i} \right\rfloor K_u^i > r_u^i + ((1-\gamma)^{L-1}N - 1)K_u^i$$

Then substituting $K_u^i = \epsilon r_u^i$, we get

$$\begin{aligned}
&\hat{z}^i(\pi_u') - (1-\gamma)^{L-1}\hat{z}^i(\pi_u) \\
&\qquad > r_u^i + (1-\gamma)^{L-1}N\epsilon r_u^i - \epsilon r_u^i - (1-\gamma)^{L-1}r_u^i - (1-\gamma)^{L-1}N\epsilon r_u^i \\
&\qquad = r_u^i(1 - \epsilon - (1-\gamma)^{L-1}) \geq 0
\end{aligned}$$

i.e., $\hat{z}^i(\pi_u') \geq (1-\gamma)^{L-1}\hat{z}^i(\pi_u)$. Similarly to the proof of Theorem 4, after discarding the dominated solutions at u, we are guaranteed to be left with a $\tilde{\pi}_u$ such that $\hat{z}^i(\tilde{\pi}_u) \geq (1-\gamma)^{L-1}\hat{z}^i(\pi_u)$. Then after running the γ-representation algorithm at node u, we have in the representation set a $\bar{\pi}_u$ such that $\hat{z}^i(\bar{\pi}_u) \geq (1-\gamma)^L \hat{z}^i(\pi_u)$. Given that $\hat{z}^i(\pi_u) \geq (1-\epsilon)z^i(\pi_u)$, proved in [26], and $z^i(\bar{\pi}_u) \geq \hat{z}^i(\bar{\pi}_u)$, as a result of taking the floor operation, we have that

$$z^i(\bar{\pi}_u) \geq \hat{z}^i(\bar{\pi}_u) \geq (1-\gamma)^L \hat{z}^i(\pi_u) \geq (1-\gamma)^L (1-\epsilon) z^i(\pi_u)$$

Lemma 2. *Suppose during a run of the FPTAS proposed in [26] with parameters ϵ and $K_v^i = \epsilon r_v$, for L levels of the tree, we apply to the nodes the γ-representation algorithm in the rounded objectives, and the parameters L, γ, and ϵ satisfy that if $L > 1$, then $(1-\gamma)^{L-1} + \epsilon \leq 1$, then the algorithm in time $O((\frac{n}{\epsilon})^{3d})$ returns a set S such that for every solution $\pi \in P$, there is a solution $\pi' \in S$ such that for all $1 \leq i \leq d$, we have $z^i(\pi') \geq (1-\gamma)^L (1-\epsilon) z^i(\pi)$.*

Proof. Applying Theorem 5 to the root node, we obtain the optimality guarantee. The FPTAS with the the γ-representation algorithm incorporated still runs in polynomial time: [26] has shown that at each node u, there are $O((\frac{n_u}{\epsilon})^d)$ partial solutions to consider, where n_u is the number of nodes in T_u, so by Theorem 3, running the γ-representation algorithm on u takes $O((\frac{n_u}{\epsilon})^{3d})$. [26] has further shown that the runtime to compute all the solutions at u is $O((\frac{n_u}{\epsilon})^{2d})$. If the γ-representation algorithm is run at u, then the total runtime at u becomes $O((\frac{n_u}{\epsilon})^{3d})$. At the root node, the total runtime is $O((\frac{n}{\epsilon})^{3d})$, where n is the number of nodes in the tree.

7 Experiments

We report experimental results on using the γ-representation algorithm to find representations, estimations and approximations of Pareto frontiers for hydropower planning in the Amazon River. To accelerate the experiments, we apply the γ-representation algorithm to independent chunks of solutions in parallel. The parallelized algorithm preserves the theoretical guarantees but might return representation sets bigger than using the non-parallelized version, since clustering and choice of representative points are local to each chunk. The bigger the chunks, the slower the algorithm runs but the less the impact on the size of the representation set. For all our experiments, we used chunks of size 50000 and distribute them across 12 threads.

Representation - Table 1 shows the results of running the γ-representation algorithm as described in Sect. 4 to find representation sets of exact Pareto frontiers for different values of γ and different criteria on the full Amazon. The number of solutions decreases substantially as γ increases. To evaluate the quality of the representation sets, we calculate their hypervolumes using the framework introduced in [5] and compare them with the baseline where $\gamma = 0$, i.e., the entire exact Pareto frontier. Specifically, we normalize each objective value $z_i(\pi)$ to $[0, 1]$ by scaling it to $z_i(\pi)^* = \frac{|z_i(\pi) - z_i(\pi)_{\text{worst}}|}{|z_i(\pi)_{\text{best}} - z_i(\pi)_{\text{worst}}|}$, where $z_i(\pi)_{\text{best}}$ and $z_i(\pi)_{\text{worst}}$ are the maximum and minimum (or minimum and maximum, if the criterion is to be minimized) that can be achieved across the whole feasible solution space, i.e. building all dams or building no dam in our case. Then, we compute for each representation set the hypervolume of the objective space dominated by the solutions in the set, with zero vector as the reference point. In general, a greater hypervolume indicates a better quality. As γ increases, the hypervolume decreases, but not significantly. For example, the hypervolume decrease for the biggest γ, i.e., $\gamma = 0.1$, ranges from 1.3% to 9.7% for the different criteria, while the reduction in the number of solutions ranges from 946 to 1620 folds.

We also report the runtime for each experiment. We see that running the γ-representation algorithm requires extra processing time after the Pareto frontier has been found. The increase in runtime is polynomial, as proved in Sect. 4.

Estimation - Tables 2, and 3 contain the results of running the γ-representation algorithm at different levels of the tree during the dynamic programming

Table 1. Representing the two-criteria Pareto frontier for the full Amazon river for energy (E), connectivity (C), and greenhouse gas emission (G). The Pareto frontier is found by the exact dynamic programming algorithm, so, the reported solutions, including the representations, are guaranteed to be exactly Pareto-optimal.

Criteria	γ	Number of Solutions	Hypervolume	Runtime (s)
EC	0	33127	0.833	7.2098
	0.001	4594	0.833	49.4176
	0.01	311	0.832	27.5256
	0.1	35	0.822	25.1264
EG	0	58762	0.807	305.5793
	0.001	11090	0.803	427.3345
	0.01	1007	0.802	358.49
	0.1	60	0.792	349.2219

Table 2. γ-representation at different levels when optimizing for energy and connectivity for the full Amazon.

γ	Level	Optimality Guarantee	Number of Solutions	Hypervolume	Runtime (s)
0	N/A	1	33127	0.833	7.2098
0.001	1	1	4594	0.833	49.4176
	2	0.998	1996	0.832	24.4195
	3	0.997	1795	0.832	19.5634
0.01	1	1	311	0.832	27.5256
	2	0.980	108	0.815	1.8864
	3	0.970	102	0.815	1.3826
0.1	1	1	35	0.822	25.1264
	2	0.810	11	0.702	0.8607
	3	0.729	10	0.677	0.5394

algorithm as described in Sect. 5. Running the γ-representation algorithm for more levels of the tree shrinks the size of the solution set drastically. On the other hand, the qualities of the estimations as measured by hypervolume when compared with the exact Pareto frontier are 8.3% to 18.7% worse for the most aggressive setting ($\gamma = 0.1$ and $L = 3$), where decreases of more than 2000 folds in the sizes of the solution sets are observed.

Moreover, when the γ-representation algorithm is applied to level 2 and 3 nodes, the number of partial solutions at those nodes decreases too. The decreases at the intermediate nodes help with reducing the runtime, since on the smaller levels, fewer combinations of partial solutions need to be considered. Overall, even though the estimation algorithm is not guaranteed to run in polynomial time, empirically we observe that it has good runtime performance.

Note that when $L = 1$, the process is equivalent to finding the exact Pareto frontier and then applying the γ-representation algorithm on the exact full frontier. Therefore, the resulting solutions are still all Pareto-optimal.

Approximation - Table 4 displays the results of the γ-representation algorithm at different levels of the tree during the FPTAS as described in Sect. 6 for the

Table 3. γ-representation at different levels when optimizing for energy and greenhouse gas emission for the full Amazon.

γ	Level	Optimality Guarantee	Number of Solutions	Hypervolume	Runtime (s)
0	N/A	1	58762	0.807	305.5793
0.001	1	1	11090	0.803	427.3345
	2	0.998	8114	0.807	493.4981
	3	0.997	8285	0.807	518.3726
0.01	1	1	1007	0.802	358.49
	2	0.980	725	0.798	33.6507
	3	0.970	798	0.799	33.5599
0.1	1	1	60	0.792	349.2219
	2	0.810	32	0.74	6.9024
	3	0.729	25	0.74	5.9851

Table 4. γ-representation when optimizing three criteria (energy, connectivity, and sediment) for the full Amazon. The Pareto frontiers are approximated by running the FPTAS with $\epsilon = 0.005$.

γ	Level	Optimality Guarantee	Number of Solutions	Hypervolume	Runtime
0	N/A	0.995	4279265	0.535	25652
0.001	1	0.994	295516	0.535	31287
0.01	1	0.985	6202	0.506	39833
0.1	1	0.896	98	0.480	34146

Table 5. γ-representation at different levels of the tree when optimizing six criteria (energy, connectivity, sediment, degree of regulation, biodiversity, and greenhouse gases) for the Marañón, a sub-basin of the Amazon. The Pareto frontiers are approximated by running the FPTAS with $\epsilon = 0.2$.

γ	Level	Optimality Guarantee	Number of Solutions	Hypervolume	Runtime (s)
0	N/A	0.8	700791	0.320	1383
0.005	1	0.796	315808	0.320	4358
	2	0.792	24554	0.313	2920
	3	0.788	22772	0.311	2002
0.1	1	0.72	15803	0.316	3264
	2	0.648	4345	0.308	1747
	3	0.583	3362	0.307	730
0.2	1	0.64	2690	0.311	2898
	2	0.512	325	0.299	1891
	3	0.410	205	0.298	640

full Amazon. We see that the effect of γ-representation is preserved when run on top of the approximation using rounded objectives. Notably for optimizing three criteria on the full Amazon, with $\gamma = 0.1$ we can decrease the number of solutions from over 4 million to 98, while the hypervolume only decreases by 10.3%.

Figure 4 plots baselines (exact or approximated Pareto frontier) and their representations, estimations, or approximations from applying the γ-representation algorithm at different levels of the tree for 2 and 3 criteria for the full Amazon. We see that the representation set is well-distributed across the Pareto frontier. They are sparser on the ends of the frontier because there we have small values for at least one of the objectives, making the $(1 - \gamma)$ bound easier to achieve.

To analyze the γ-representation algorithm for a larger number of objectives, we have also experimented with optimizing six objectives for the Marañón, a sub-basin of the Amazon. The choice of the smaller basin allows us to run more objectives, in a reasonable amount of time. The results are reported in Table 5. For a large number of objectives, applying the γ-representation algorithm also results in a significant decrease in the number of solutions. Similarly, the decreased number of solutions at the intermediate nodes has improved the runtime.

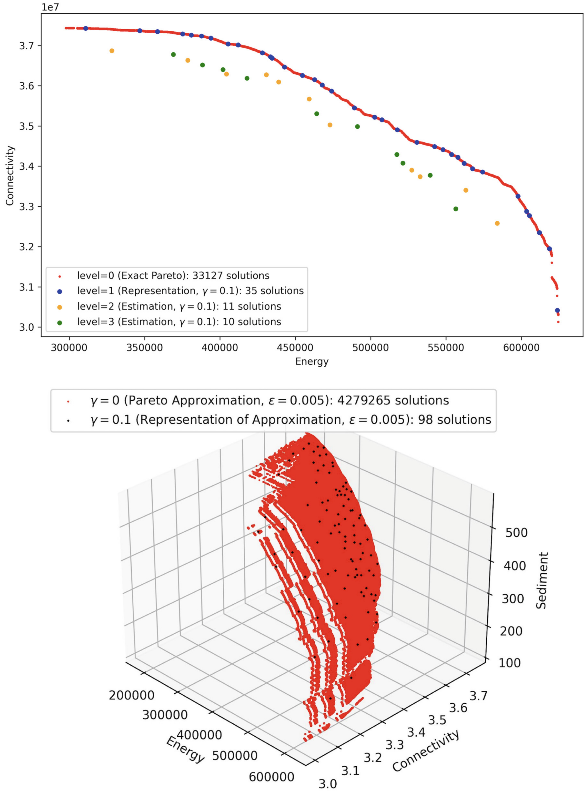


Fig. 4. Top panel: Exact Pareto frontier (# solutions: 33127) and its representation (# solutions: 35) and estimations (# solutions for level = 2: 11, # solutions for level = 3: 10) from applying the γ-representation algorithm at different levels of the tree for energy and connectivity for the full Amazon with $\epsilon = 0$ and $\gamma = 0.1$. Bottom panel: Approximated Pareto frontiers for energy, connectivity, and sediment for the full Amazon with $\epsilon = 0.005, \gamma = 0$ (# solutions: 4279265) and $\epsilon = 0.005, \gamma = 0.1$ (# solutions: 98)

8 Conclusion

We propose a clustering-based algorithm to find a **representation** set from the Pareto frontier with a coverage guarantee, which runs in time polynomial in the size of the frontier. We also consider two different strategies for incorporating the representation algorithm into a dynamic-programming-based approach: an **approximation** strategy, with polynomial runtime and optimality guarantee, and an **estimation** strategy with optimality guarantee and good empirical runtime performance, but without polynomial runtime guarantee. The three methods provide different ways to compress the Pareto frontier, resulting in solution sets significantly smaller than the full Pareto frontier, which are

γ-representations of the exact Pareto frontier or close to the frontier. Our main goal is to equip policymakers with streamlined approaches for effectively navigating Pareto frontiers, thus facilitating a more efficient decision-making process. Moreover, we hope our work will catalyze further research on the computation and visualization of Pareto frontiers. Multi-objective Pareto optimization is key to understanding trade-offs among various objectives, thus playing a pivotal role in the development of AI decision-support systems for informed decision-making.

References

1. Almeida, R.M., et al.: Reducing greenhouse gas emissions of amazon hydropower with strategic dam planning. Nat. Commun. **10**(1), 1–9 (2019)
2. Andersen, H.R., Hadzic, T., Hooker, J.N., Tiedemann, P.: A constraint store based on multivalued decision diagrams. In: Bessière, C. (ed.) CP 2007. LNCS, vol. 4741, pp. 118–132. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74970-7_11
3. Bai, Y., Shi, Q., Grimson, M., Flecker, A., Gomes, C.P.: Efficiently approximating high-dimensional pareto frontiers for tree-structured networks using expansion and compression. In: Cire, A.A. (eds.) Integration of Constraint Programming, Artificial Intelligence, and Operations Research. CPAIOR 2023. LNCS, vol. 13884, pp. 1–17. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-33271-5_1
4. Bergman, D., Cire, A.A.: Multiobjective optimization by decision diagrams. In: Rueher, M. (ed.) CP 2016. LNCS, vol. 9892, pp. 86–95. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-44953-1_6
5. Cao, Y., Smucker, B.J., Robinson, T.J.: On using the hypervolume indicator to compare pareto fronts: applications to multi-criteria optimal experimental design. J. Stat. Plan. Inference **160**, 60–74 (2015). https://doi.org/10.1016/j.jspi.2014.12.004, https://www.sciencedirect.com/science/article/pii/S0378375814002006
6. Chen, W., Ishibuchi, H., Shang, K.: Clustering-based subset selection in evolutionary multiobjective optimization. In: 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 468–475. IEEE (2021)
7. Deb, K., Jain, H.: An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: solving problems with box constraints. IEEE Trans. Evol. Comput. **18**(4), 577–601 (2013)
8. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: nsga-ii. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
9. Doerr, B., Qu, Z.: A first runtime analysis of the nsga-ii on a multimodal problem. IEEE Transactions on Evolutionary Computation (2023)
10. Doerr, B., Qu, Z.: From understanding the population dynamics of the nsga-ii to the first proven lower bounds. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, pp. 12408–12416 (2023)
11. Doerr, B., Qu, Z.: Runtime analysis for the nsga-ii: Provable speed-ups from crossover. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 12399–12407 (2023)
12. Flecker, A.S., et al.: Reducing adverse impacts of amazon hydropower expansion. Science **375**(6582), 753–760 (2022)
13. Gomes, C., et al.: Computational sustainability: computing for a better world and a sustainable future. Commun. ACM **62**(9), 56–65 (2019)

14. Gomes-Selman, J.M., Shi, Q., Xue, Y., García-Villacorta, R., Flecker, A.S., Gomes, C.P.: Boosting efficiency for computing the pareto frontier on tree structured networks. In: van Hoeve, W.-J. (ed.) CPAIOR 2018. LNCS, vol. 10848, pp. 263–279. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93031-2_19
15. Grimson, M., et al.: Scaling up pareto optimization for tree structures with affine transformations: Evaluating hybrid floating solar-hydropower systems in the amazon. In: Proceedings of the AAAI Conference on Artificial Intelligence (submitted)
16. Johnson, S.C.: Hierarchical clustering schemes. Psychometrika **32**(3), 241–254 (1967). https://doi.org/10.1007/bf02289588, http://dx.doi.org/10.1007/BF02289588
17. Murtagh, F., Legendre, P.: Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion? J. Classif. **31**(3), 274–295 (2014). https://doi.org/10.1007/s00357-014-9161-z, http://dx.doi.org/10.1007/s00357-014-9161-z
18. Newman, M.E.J.: Finding community structure in networks using the eigenvectors of matrices. Phys. Rev. E **74**(3) (2006). https://doi.org/10.1103/physreve.74.036104, http://dx.doi.org/10.1103/PhysRevE.74.036104
19. Sahraei, S., Asadzadeh, M.: Cluster-based multi-objective optimization for identifying diverse design options: application to water resources problems. Environ. Model. Softw. **135**, 104902 (2021)
20. Sibson, R.: Slink: an optimally efficient algorithm for the single-link cluster method. Comput. J. **16**(1), 30–34 (1973). https://doi.org/10.1093/comjnl/16.1.30, http://dx.doi.org/10.1093/comjnl/16.1.30
21. Sokal, R.R.: Numerical taxonomy. Sci. Am. **215**(6), 106–117 (1966). http://www.jstor.org/stable/24931358
22. Srinivas, N., Deb, K.: Muiltiobjective optimization using nondominated sorting in genetic algorithms. Evol. Comput. **2**(3), 221–248 (1994)
23. United Nations General Assembly: Transforming our world: the 2030 agenda for sustainable development (2015). https://sdgs.un.org/2030agenda
24. Vamplew, P., Dazeley, R., Foale, C., Firmin, S., Mummery, J.: Human-aligned artificial intelligence is a multiobjective problem. Ethics Inf. Technol. **20**, 27–40 (2018)
25. Wei, D., Jiang, Q., Wei, Y., Wang, S.: A novel hierarchical clustering algorithm for gene sequences. BMC Bioinform. **13**(1) (2012). https://doi.org/10.1186/1471-2105-13-174, http://dx.doi.org/10.1186/1471-2105-13-174
26. Wu, X., et al.: Efficiently approximating the pareto frontier: hydropower dam placement in the amazon basin. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)
27. Zhang, H., Song, S., Zhou, A., Gao, X.Z.: A clustering based multiobjective evolutionary algorithm. In: 2014 IEEE Congress on Evolutionary Computation (CEC), pp. 723–730. IEEE (2014)
28. Zhang, Q., Li, H.: MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Trans. Evol. Comput. **11**(6), 712–731 (2007)
29. Zheng, W., Liu, Y., Doerr, B.: A first mathematical runtime analysis of the Non-Dominated Sorting Genetic Algorithm II (NSGA-II). In: Conference on Artificial Intelligence, AAAI 2022. AAAI Press (2022). preprint at https://arxiv.org/abs/2112.08581
30. Zhou, S., et al.: A multi-objective evolutionary algorithm with hierarchical clustering-based selection. IEEE Access **11**, 2557–2569 (2023)