Proceedings of the IISE Annual Conference & Expo 2023
K. Babski-Reeves, B. Eksioglu, D. Hampton, eds.

Classification of Human Enrichment and Refinement in Interactive Optimization

Michelle Zhang, Jennifer Pazour, Sandipan Mishra, John E. Mitchell, Damsara Jayarathne
Rensselaer Polytechnic Institute, Troy, NY

Abstract

Optimization models can recommend the best systematic decisions in the face of exponentially many options, decision trade-offs, systematic interactions and constraints on finite resources, tasks notoriously difficult for human decision makers (DM). Yet, a model âĂŞ by definition âĂŞ is only a representation of the system that a DM is interested in optimizing. Human DMs typically have access to important local context, domain knowledge, and goals. Thus, interactive optimization (IO) tools, which are âĂIJhuman-in-the-loopâĂİ approaches that iteratively involve the human DM in the optimization process, are critical for decision making. This paper presents a framework for classifying the different approaches in IO to elicit and embed human DMâĂŹs feedback into a mathematical program. Towards this goal, we build on a previous taxonomy proposed by Meignan et al. in 2015, focusing on problem-oriented interactions. We propose a classification scheme that captures these aspects, where the current model formulation may be incomplete and the human DM provides feedback that can enrich the optimization model and problem instantiation data. Using this scheme, we review the existing literature in this category to find prior work that focuses on eliciting preference information for multiobjective problems. Based on this review, we then present our conclusions and discuss opportunities for creating optimization methods that capture more richly the human side of knowledge.

Keywords
Human-in-the-loop, Interactive Optimization, Math Programming, Operations Research

1. Introduction

Interactive optimization (IO) is a field of research that supports a human decision maker's (DM) active participation in the optimization process by incorporating their feedback in an iterative fashion. This human interaction may occur during any stage of the optimization process. A DM can be asked to update the problem domain with new data, update their preferences, and add or remove constraints (among other interactions). Generally, interactions between a human DM and the optimization system are provided through an interface and interactions continue in an iterative manner until the DM is satisfied with the solution found.

IO is a key tool for addressing several challenges in the integration of advanced optimization methods into decision support tools [1]. First, it is often difficult for human DMs to specify, beforehand, all facets of their real world problem, whether that is applicable constraints, available data, or selection criteria for the problem. Thus, asking the DM pointed questions and to provide updated information can help elicit a richer model formulation. Further, as DMs must understand and trust the solutions generated by the optimization system in order to implement, modify, and justify them, IO approaches also aid in building a DM's understanding of the methodology and the system being optimized. Therefore, IO is an advantageous approach that can help bridge the gap between the information captured in the optimization model and the real world problem and can increase the likelihood for deployment.

Consequently, there has been significant research and prior literature on IO methods. IO approaches range from rudimentary strategies like trial-and-error [2, 3] to complex methods, such as multiobjective (MO) optimization [4] and human-guided search [5]. In 2015, Meignan et al. [1] reviewed the literature in IO and proposed a useful classification of interactive approaches based on a DM's interaction with the optimization system. Specifically they classified IO methods in terms of the purpose of the interaction and the role of the DM, and characteristics of the model, including the type of feedback integration, the preference information lifetime, and the type of optimization procedure. Follow up reviews such as Dudley et al. in 2018 [6] provide a detailed overview on interactive machine learning systems, highlighting user interface designs. Although different from IO, these concepts and associated definitions are closely related. Other subsequent review studies have focused specifically on interactive MO optimization, see [7, 8].

In this work, we develop a classification scheme to emphasize the potential for IO approaches that more richly engage with a human DM. To do so, we use the taxonomy proposed by Meignan et al. 2015 [1] as a guide-post to define certain terms and characteristics in IO. We then build a new classification scheme focusing on the different ways IO methods capture and decode the knowledge from a DM and use this additional information to refine the optimization system, and then apply this scheme to review existing IO methods and to identify promising future research areas.

2. Classification

Our review's scope is limited to IO papers that meet three criteria. First, a human DM must have an interaction with the optimization system. We include papers that use artificial DMs only if the intended purpose of the methodology is for a human DM. Methods designed to interact with nonhuman DMs are out of scope. Second, the paper must have a mathematical programming formulation that describes in mathematical form specific objective function(s), decision variables, and constraints. Meignan et al. 2015 [1] classifies the purpose of DM interaction to be problem-oriented or search-oriented. Third, the paper must have a problem-oriented interaction where the DM has additional knowledge of the problem domain not yet captured by the formulation. We do not consider search-oriented papers as the interaction with the DM is to improve the solution approach efficiency (but assumes the model formulation is fixed) [5].

2.1 Classification Scheme

Our review focuses on IO approaches and the different ways they capture and interpret the knowledge from a DM and use this additional information to refine the optimization system. To do so, we create a classification schema that considers the following characteristics, which we define as a set of questions.

(1) Does the human DM play the role of an adjuster, enricher, or both? (We define these terms based on the concepts introduced by Meignan et al. 2015 [1].)

A DM is an adjuster if they provide feedback on the values of some parameters of the constraints or the objectives. Notably, the human DM does not change the functional form of the optimization model, just provides feedback on the input parameter values, or expanding the elements of proposed sets. Hence, the adjuster will only be refining the model with data (see Question 2 for further classification). A DM is an enricher if they provide feedback that leads to structural changes to the optimization formulation. This is typically in the format of adding or removing some constraints or objectives, or changing or expanding previously defined sets. This is different from the role of an adjuster since the assumption here is that the proposed optimization model may be incomplete and does not capture all the facets that the DM's real problem possesses. Thus, the DM will only be refining the model with domain knowledge (see Question 2). A DM is both if they play the role of an adjuster and an enricher.

(2) What type of information is asked from the human DM to refine the optimization system?

Data is refined when a DM, as an adjuster updates parameter values, such as adjusting the optimization model's objective coefficients or the constraints' input parameter coefficients, or changing or expanding the elements of previously defined sets. Notably, this type of refinement does not change the model formulation's functional form. However by asking the DM to provide refined input data, it could lead to better quality or new data. This could be in the form of different parameter values or redefining sets. For example, if the DM adds a new supplier, this will only be an additional element in a well defined set. It will not lead to an additional constraint in the model. A special type of data is based on DM's preference. The majority of IO papers focus on preference updates, where in interactive MO optimization, the human DM is asked to provide information that the system can use to update weights across the set of multiple objectives or criteria. The system asks the DM for domain information in an attempt to refine some aspects about the functional form of the optimization model. Given human DMs typically have access to important local context, domain knowledge, and goals, this type of refinement uses such new information to enhance the optimization formulation. There may be more restrictions on the optimization problem that needs to be specified or a new decision criteria that was previously not captured in the model. In this case, the DM has information that the optimization system does not have. For instance, the DM may know from previous cases that a specific supplier will be closed on weekends. Hence, the system adds a constraint on availability for this supplier.

(3) What is the level of knowledge the DM is expected to have?

A math programming expert (MPE) is expected to know mathematical formulations and how adding or removing constraints/objectives will affect the model. They have extensive knowledge about the math background of the proposed problem. A decision-making context expert (DMCE) understands the domain and decision being made, but is not expected to be an expert in math programming. Both if the DM has math programming and decision-making knowledge.

(4) What is the type of optimization problem?

We first classify if a model is single objective (SO), or multiobjective (MO). We further classify the model based on the problem structure : discrete (D) vs. continuous (C) , and linear (L) vs. nonlinear (NL) . We use the classification defined in neos Guide `https://neos-guide.org/guide/types/`.

(5) Is the interaction question type asked of the human DM static or adaptive?

A static approach will ask the DM the same type of question in all interactions (albeit the context of the question typically changes over time as the output from the optimization model changes). An adaptive approach asks the DM different types of questions, depending on the response of the user and the nature of information needed for refinement.

(6) How do human DMs provide feedback?

DMs provide feedback by answering targeted questions and we classify approaches in terms of if the human DM provides this feedback by (i) list if they select from a pre-populated list of potential options or more specifically (i.1) a specialized list is yes/no if they are only allowed to respond back with an answer of "yes" or "no", (ii) value if they are asked to enter numerical values, (iii) rank if they are asked to rank information provided from the optimization model, or (iv) other if they are asked to provide feedback in another way (which may be to upload new data).

(7) What approach is used in creating the refinement?

Approaches can be math programming informed, such as using dual variables to recommend refinement actions, preference such as trade-off information, reference point approaches, or classification-based methods (see [4]), or other sophisticated approaches.

2.2 Literature Review using the Classification Scheme

In January 2023, we reviewed papers on Google Scholar that cited Meignan et al. 2015 [1]. We combine the 15 papers we found with the papers that Meignan et al. 2015 [1] reviewed, which incorporate all three of the features identified in Section 2. In Table 1, we classify these 28 papers, using the scheme identified in Section 2.1.

3. Conclusions and Discussion of Promising Research Directions within IO.

An extensive amount of work in IO has been done in the past two decades. Most reviewed papers (21 out of 28) view the DM's role as an adjuster that provides feedback on the values of some parameters used in the constraints or the objectives. The primary focus in problem-domain IO methods has been on MO problems, where the interaction is to ask the DM to provide new preference data among multiple objectives. Existing IO methods are well designed to gain and integrate additional information about a human's preferences among a set of criteria into optimization models.

All reviewed papers in Table 1 take a static approach to asking the DM questions. Static approaches provide useful structure, which is often exploited to create sophisticated, math programming informed refinement approaches that integrate the human's additional knowledge in such a way that preserves or improves some performance metric of interest. Yet, given existing approaches focus on a single type of refinement action, repeatedly asked to the human DM, this also limits the ability of existing IO methods to extract a very specific and narrow piece of knowledge from the human DM. The majority of the existing work assumes the question to ask the DM is known a priori and is not something that the IO method is deciding over time based on the state of the solutions being produced or satisfaction with the solution by the DM. Further, DMs are typically incorporated after a mathematical program has been formulated. Thus, there is a need to include the DM throughout the design and decision process.

A general observation is that the current state of IO focuses on how the optimization formulation can be enhanced by a human, however, the existing approaches tend to under-utilize the human's additional knowledge and goals, and primarily focus on obtaining better estimates of a DM's preferences among a set of criteria. This is valuable information, yet, the human DM has access to a wide range of additional information. Needed are new approaches that can extract a broader range of knowledge from the DM. To do this effectively, these approaches should be adaptive, in that they can dynamically change what type of information to ask from the DM, and math programming informed, in that what additional information asked of the human DM is guided by properties of the optimization problem.

Acknowledgements

This research was supported by the Office of Naval Research under grant number N00014-22-1-2542.

Table 1: Problem-Oriented Interactive Optimization Papers Classified Using the Scheme in Section 2.1.

	Characteristics						
LITERATURE	**Role of DM?**	**Type of information asked from DM to refine the model?**	**Level of knowledge the DM is expected to have?**	**Type of optimization problem**	**Type of interaction question?**	**How do DMs provide feedback?**	**What approach is used in creating refinement?**
	Adjuster, Enricher, or Both	Data, Preference, Domain	MPE, DMCE, or Both	SO or MO - D vs. C, & L vs. NL	Static, or Adaptive	(i) List, (ii) Value, (iii) Rank, (iv) Other	(i) Math program, (ii) Preference, (iii) Other
Hamel et al. 2012 [10]	Both	Preference	DMCE	SO - D & L	Static	(i) List	(iii) Optimality range
van Vliet et al. 1992 [11]	Both	Data, Preference, and Domain	DMCE	MO - D & L	Static	(i) List, (ii) Value, (iv) Fix or change part of the solution	(i) Math program, (iii) Checks feasibility of changed solutions
Meignan, 2014 [12]	Enricher	Domain	DMCE	MO - D & L	Static	(ii) Value	(iii) Introduces soft constraint
Meignan, 2015 [13]	Enricher	Domain	DMCE	MO - D & L	Static	(ii) Value	(iii) Add two supplementary objectives
Deb and Sundar, 2006 [14]	Adjuster	Preference	DMCE	MO - D, C & L, NL	Static	(ii) Value	(ii) Preference - Reference point
Jaszkiewicz and SÅĆowiÅĎski, 1999 [15]	Adjuster	Preference	DMCE	MO - C & L, NL	Static	(ii) Value	(ii) Preference - Reference point
Miettinen and MÃďkelÃď, 2000 [16]	Adjuster	Preference	DMCE	MO - C & NL	Static	(iv) Other	(ii) Preference - Classification based
Greco et al. 2008 [17]	Adjuster	Preference	DMCE	MO - D & L	Static	(i) List, (iv) Other	(ii) Preference - Pareto optimal set
Miettinen et al. 2010 [18]	Adjuster	Preference	DMCE	MO - D & L	Static	(ii) Value, (iii) Rank	(iii) NAUTILUS method
Laukkanen et al. 2012 [19]	Adjuster	Preference	DMCE	MO - D & NL	Static	(ii) Value, (iii) Rank	(iii) Bilevel optimization method
Miettinen, 2007 [20]	Adjuster	Preference	DMCE	MO -D & NL	Static	(i) List	(iii) Scalarization-based method and genetic algorithm
Ruotsalainen et al. 2010 [21]	Adjuster	Preference	DMCE	MO - D & L	Static	(i) List, (iv) Classify objective functions	(ii) Preference
Tveit et al. 2012 [22]	Adjuster	Preference	DMCE	MO - D & L	Static	(iv) Classify objective functions	(iii) Generate regression models for each objective
Ahani et al. 2021 [23]	Adjuster	Data	DMCE	SO - D & L	Static	(iv) Matching process and fine-tune the optimization results	(iii) Crossreferencing
Trachanatzi et al. 2020 [24]	Adjuster	Preference	DMCE	MO - D & L	Static	(iii) Rank	(iii) Preference
He et al. 2019 [25]	Enricher	Domain	DMCE	SO - D & L	Static	(iv) Adding or deleting structural elements	(i) Math program
Ye et al. 2022 [26]	Adjuster	Preference	DMCE	MO - D & L	Static	(i) List	(iii) Adding auxiliary factor values of solutions
Feit et al. 2021 [27]	Both	Both	Both	SO - D & L	Static	(iv) Other	(i) Math program, (ii) Preference
Piemonti et al. 2017 [28]	Adjuster	Preference	DMCE	MO - D & L	Static	(i) List (iii) Rank	(i) Math program
Ruiz et al. 2019 [29]	Adjuster	Preference	DMCE	MO - D & L	Static	(i) List, (ii) Value, (iv) Reference point	(iv) NAUTILUS method
Hu et al. 2021 [30]	Adjuster	Preference	DMCE	MO - D & L	Static	(iii) Rank	(ii) Preference
Jatschka et al. 2021 [31]	Adjuster	Preference	DMCE	SO - D & L	Static	(i) List	(i) Math program
Hanine et al. 2021 [32]	Adjuster	Preference	DMCE	MO - D & L	Static	(i) List	(i) Math program
Jatschka et al. 2019 [33]	Adjuster	Preference	DMCE	SO - D & L	Static	(i) List, (ii) Value	(i) Math program
Jatschka et al. 2019 [34]	Adjuster	Preference	DMCE	SO - D & L	Static	(ii) Value	(i) Math program
Jatschka et al. 2022 [35]	Adjuster	Preference	DMCE	SO - C & L	Static	(i) List, (ii) Value	(i) Math program
Figueiredo et al. 2022 [36]	Adjuster	Data, Preference	DMCE	MO - C & L	Static	(ii) Value	(i) Math program
Meignan 2018 [37]	Enrincher	Domain, Preference	DMCE	SO - D & L	Static	(ii) Value	(iii) Reoptimization

References

[1] Meignan, D., Knust, S., Frayret, J.M., Pesant, G. and Gaud, N., 2015. "A review and taxonomy of interactive optimization methods in operations research," ACM Transactions on Interactive Intelligent Systems, 5(3), 1-43.

[2] Cesta, A., Cortellessa, G., Oddi, A. and Policella, N., 2003. "A CSP-based interactive decision aid for space mission planning," In Congress of the Italian Association for Artificial Intelligence (511-522). Springer, Berlin.

[3] Cesta, A., Cortellessa, G., Denis, M., Donati, A., Fratini, S., Oddi, A., Policella, N., Rabenau, E. and Schulster, J., 2007. "Mexar2: AI solves mission planner problems," IEEE Intelligent Systems, 22(4), 12-19.

[4] Miettinen, K., Ruiz, F. and Wierzbicki, A.P., 2008. "Introduction to multiobjective optimization: interactive approaches," In Multiobjective optimization (27-57). Springer, Berlin, Heidelberg.

[5] Klau, G.W., Lesh, N., Marks, J. and Mitzenmacher, M., 2010. "Human-guided search," Journal of Heuristics, 16(3), 289-310.

[6] Dudley, J.J. and Kristensson, P.O., 2018. "A review of user interface design for interactive machine learning," ACM Transactions on Interactive Intelligent Systems (TiiS), 8(2), 1-37.

[7] Afsar, B., Miettinen, K. and Ruiz, F., 2021. "Assessing the performance of interactive multiobjective optimization methods: a survey," ACM Computing Surveys (CSUR), 54(4), 1-27.

[8] Xin, B., Chen, L., Chen, J., Ishibuchi, H., Hirota, K. and Liu, B., 2018. "Interactive multiobjective optimization: A review of the state-of-the-art," IEEE Access, 6, 41256-41279.

[16] Miettinen, K. and MÃď'kelÃď', M.M., 2000. "Interactive multiobjective optimization system WWW-NIMBUS on the Internet," Computers and Operations Research, 27(7-8), 709-723.

[10] Hamel, S., Gaudreault, J., Quimper, C.G., Bouchard, M. and Marier, P., 2012, October. "Human-machine interaction for real-time linear optimization," In 2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 673-680). IEEE.

[11] van Vliet, A., Boender, C.G.E. and Rinnooy Kan, A.H., 1992. "Interactive optimization of bulk sugar deliveries," Interfaces, 22(3), pp.4-14.

[12] Meignan, D., 2014, July. "A heuristic approach to schedule reoptimization in the context of interactive optimization," In Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation (pp. 461-468).

[13] Meignan, D., 2015. "An experimental investigation of reoptimization for shift scheduling," In Proceedings of the 11th Metaheuristics International Conference (pp. 1-10).

[14] Deb, K. and Sundar, J., 2006, July. "Reference point based multi-objective optimization using evolutionary algorithms," In Proceedings of the 8th annual conference on Genetic and evolutionary computation (pp. 635-642).

[15] Jaszkiewicz, A. and SÅĆowiÅĎski, R., 1999. "The âĂŸLight Beam SearchâĂŹapproachâĂŞan overview of methodology applications," European Journal of Operational Research (EJOR), 113(2), pp.300-314.

[16] Miettinen, K. and MÃď'kelÃď', M.M., 2000. "Interactive multiobjective optimization system WWW-NIMBUS on the Internet," Computers & Operations Research, 27(7-8), pp.709-723.

[17] Greco, S., Matarazzo, B. and SÅĆowiÅĎski, R., 2008. "Dominance-based rough set approach to interactive multiobjective optimization," Multiobjective optimization: Interactive and evolutionary approaches, pp.121-155.

[18] Miettinen, K., Eskelinen, P., Ruiz, F. and Luque, M., 2010. "NAUTILUS method: An interactive technique in multiobjective optimization based on the nadir point," EJOR, 206(2), pp.426-434.

[19] Laukkanen, T., Tveit, T.M., Ojalehto, V., Miettinen, K. and Fogelholm, C.J., 2012. "Bilevel heat exchanger network synthesis with an interactive multi-objective optimization method," Applied Thermal Engineering, 48, pp.301-316.

[20] Miettinen, K., 2007. "Using interactive multiobjective optimization in continuous casting of steel," Materials and Manufacturing Processes, 22(5), pp.585-593.

[21] Ruotsalainen, H., Miettinen, K. and Palmgren, J.E., 2010. "Interactive multiobjective optimization for 3D HDR brachytherapy applying IND-NIMBUS," In New Developments in Multiple Objective and Goal Programming (pp. 117-131). Springer Berlin Heidelberg.

[22] Tveit, T.M., Laukkanen, T.P., Ojalehto, V., Miettinen, K. and Fogelholm, C.J., 2012. "Interactive Multi-objective Optimisation of Configurations for an Oxyfuel Power Plant Process for CO 2 Capture," Chemical Engineering Transactions, 29, pp.433-438.

[23] Ahani, N., Andersson, T., Martinello, A., Teytelboym, A. and Trapp, A.C., 2021. "Placement optimization in refugee resettlement," Operations Research, 69(5), pp.1468-1486.

[24] Trachanatzi, D., Rigakis, M., Marinaki, M. and Marinakis, Y., 2020. "An interactive preference-guided firefly algorithm for personalized tourist itineraries," Expert Systems with Applications, 159, p.113563.

[25] He, L., Gilbert, M., Johnson, T. and Pritchard, T., 2019. "Conceptual design of AM components using layout and geometry optimization," Computers & Mathematics with Applications, 78(7), pp.2308-2324.

[26] Ye, Q., Li, F. and Hu, Y., 2022. "Interactive portfolio optimization with cognition-limited human decision making assisted by auxiliary factors."

[27] Feit, A.M., Nancel, M., John, M., Karrenbauer, A., Weir, D. and Oulasvirta, A., 2021. "AZERTY amÃľliorÃľ: computational design on a national scale," Communications of the ACM, 64(2), pp.48-58.

[28] Piemonti, A.D., BabbarâĂŘSebens, M., Mukhopadhyay, S. and Kleinberg, A., 2017. "Interactive genetic algorithm for userâĂŘcentered design of distributed conservation practices in a watershed: an examination of user preferences in objective space and user behavior," Water Resources Research, 53(5), pp.4303-4326.

[29] Ruiz, A.B., Ruiz, F., Miettinen, K., Delgado-Antequera, L. and Ojalehto, V., 2019. "NAUTILUS Navigator: free search interactive multiobjective optimization without trading-off," Journal of Global Optimization, 74(2), pp.213-231.

[30] Hu, S., Li, D., Jia, J. and Liu, Y., 2021. "A Self-Learning Based Preference Model for Portfolio Optimization," Mathematics, 9(20), p.2621.

[31] Jatschka, T., Raidl, G.R. and Rodemann, T., 2021. "A general cooperative optimization approach for distributing service points in mobility applications," Algorithms, 14(8), p.232.

[32] Hanine, Y., Lamrani Alaoui, Y., Tkiouat, M. and Lahrichi, Y., 2021. "Socially responsible portfolio selection: an interactive intuitionistic fuzzy approach," Mathematics, 9(23), p.3023.

[33] Jatschka, T., Rodemann, T. and Raidl, G.R., 2019. A cooperative optimization approach for distributing service points in mobility applications. In Evolutionary Computation in Combinatorial Optimization: 19th European Conference, Proceedings 19 (pp. 1-16). Springer International Publishing.

[34] Jatschka, T., Rodemann, T. and R. Raidl, G., 2019. Exploiting similar behavior of users in a cooperative optimization approach for distributing service points in mobility applications. In Machine Learning, Optimization, and Data Science: 5th International Conference, Proceedings 5 (pp. 738-750). Springer International Publishing.

[35] Jatschka, T., Rodemann, T. and Raidl, G.R., 2022. A Large Neighborhood Search for a Cooperative Optimization Approach to Distribute Service Points in Mobility Applications. In Metaheuristics and Nature Inspired Computing: 8th International Conference (pp. 3-17). Springer International Publishing.

[36] Figueiredo, M.V., Silva, A.F. and Marins, F.A.S., 2022. Multi-Objective Optimization and Design of Experiments Applied on Orthopedic Assets Distribution Process. Available at SSRN 4178121.

[37] Meignan, D., 2018. A user experiment on interactive reoptimization using iterated local search. Recent Developments in Metaheuristics, pp.399-413.