Analysing Metaheuristic Components

Helena Stegherr and Jörg Hähner

University of Augsburg, 86159 Augsburg, Germany
helena.stegherr@informatik.uni-augsburg.de

Abstract

The vast number of nature-inspired metaheuristics makes it increasingly difficult to keep an overview of efficient and innovative developments. Especially novel approaches based on obscure metaphors and lacking rigorous evaluation are often—and usually rightfully—disregarded in research and application. However, even among established approaches, new developments are hard to detect and integrating them into the current set of methods is not an easy task. Altogether, finding a suitable metaheuristic for the problem at hand is aggravated in all application domains, including Lifelike Computing Systems. In this paper, we present ways that can facilitate extracting relevant information on metaheuristics. The approach is centred around a unified view on metaheuristics, with a focus on their components as the relevant parts determining the performance and the behaviour of metaheuristic frameworks and algorithms. We furthermore describe strategies for the conceptual and empirical analysis of those components. This procedure can be applied in different levels of detail and is therefore adaptable to the respective goals of the investigation of different metaheuristics. Its advantages and problems are discussed and we conclude that this is one possible and useful way to gain a better understanding of existing metaheuristics and to deal with new approaches.

Introduction

Metaheuristics are capable of successfully approximating solutions of black-box optimisation problems where exact optimisers are not applicable. This makes them suitable for a number of tasks, from engineering to biology/medicine, but also within more complex computing systems (Hussain et al., 2018). In Lifelike Systems, metaheuristics are often utilised to optimise the parameters of other components, especially machine learning components, enabling the self-improvement mechanism of these systems. These parameter optimisation problems can differ in their often unknown fitness landscapes and the task is complicated by the dynamically changing environment in which lifelike systems are deployed. Additionally, there can be several areas in the system that require an optimiser, e. g. learning components, other optimisers or the environment itself.

For all application areas of metaheuristics, there arises the same initial question: Which metaheuristic is the most suitable for the given optimisation problem? This question results from the *No free lunch* theorem, which states that no metaheuristic performs best on all problems (Wolpert and Macready, 1997). To some extent, this also led to an increasing amount of different metaheuristics, hybrids and variants, with more than 300 approaches by 2020, summarised in a presumably non-exhaustive list by Molina et al. (2020). As most of those are strongly metaphor-based, it is hard to detect innovative and efficient strategies that could be advantageous for the given problem. However, falling back to well known approaches, e. g. evolutionary algorithms, might restrict performance as more suitable strategies exist. Altogether, this results in a necessity to facilitate the assessment of metaheuristics in terms of their functionality, performance and behaviour.

In this paper, we argue on the importance of conceptual and empirical analysis of metaheuristic components, based on a unified framework, and present our research agenda on this behalf. We first specify how such a unified framework can be described and utilised. The next section provides insights into conceptual ways to analyse metaheuristics based on their components and how this can be complemented by empirical studies. The advantages and problems of the approach itself and in relation to Lifelike Systems are discussed and we end on a short conclusion and illustrate options for future work.

A Unified View on Metaheuristics

The development of a unified concept for metaheuristics ultimately results from the demand of more standardisation, reusability, knowledge on components and consistency in descriptions (Swan et al., 2015; Sörensen, 2015). In recent years, some detailed unification strategies were presented, each of them with a different goal in mind: from providing a basis for describing metaheuristics (Bandaru and Deb, 2016) to finding inherent strategies in metaheuristics (Chicco and Mazza, 2020), but also to construct new algorithms (Song and Fong, 2016), to compare (de Armas et al., 2021) and evaluate (Cruz-Duarte et al., 2020) metaheuristics in terms of their components. Ultimately, the unification facilitates deriving differences and similarities of metaheuristics, enables the transfer of features and the construction of combinations of metaheuristics (Bandaru and Deb, 2016).

Based on these approaches towards a unified metaheuristic framework, a component-based concept is depicted in Figure 1. It combines the ideas of Bandaru and Deb (2016) and de Armas et al. (2021) and is applicable to both, a conceptual comparison of metaheuristics based on their components, and the empirical analysis of the specific implementations. The main components identified are those for initialisation, selection, solution generation, replacement and update of solution, archiving and termination. Their structure and their common operators delineate the different metaheuristic frameworks as defined by Sörensen and Glover (2013). A specific metaheuristic algorithm is then determined by using individual operators for each component, usually depending on the problem. Additionally, more than one operator can be inherent to one component, especially for the generation and archiving mechanisms.

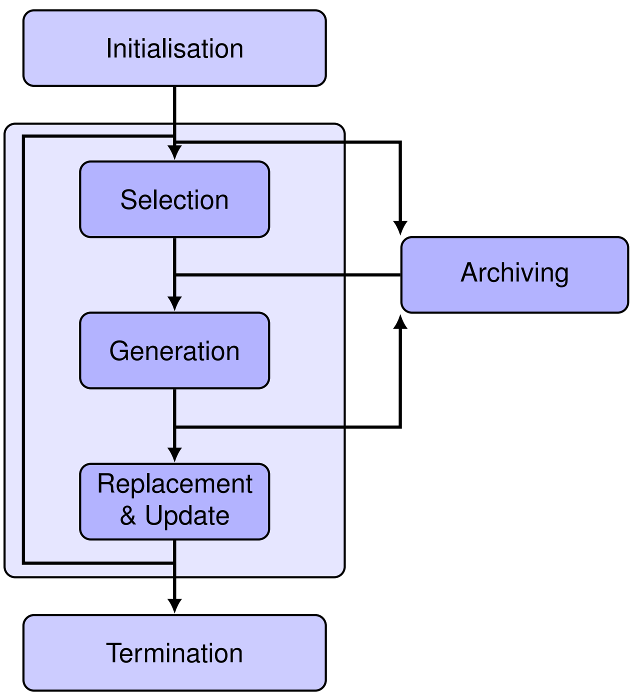

Figure 1: Unified component view on metaheuristics (adapted from Bandaru and Deb (2016); de Armas et al. (2021)).

The advantages of such a unified view on metaheuristics are manifold. On the one hand, it facilitates the analysis of existing algorithms independent of their metaphor. On the other hand, the descriptions of new approaches can be oriented towards a unified framework to allow an easier grasp of their concepts. Implementation-wise, it enables a modular approach for the construction of metaheuristic algorithms and an unproblematic exchange of components and operators, thus assisting in the assembly of algorithms for the given problem, but also in the development of hybrid algorithms and hyperheuristics.

Analysis of Metaheuristic Components

With a unified framework as a basis for dissecting metaheuristics, the resulting components can be further analysed to determine their respective capabilities. This can be by a purely conceptual analysis, enabling a rough assessment of the functioning and the features of the components and their operators, or by empirical studies based on an component-based implementation with operator exchange facilities.

Conceptual Analysis

A conceptual analysis aims at finding common features of components, operators and their different possible combinations. General features inherent to metaheuristic components and their operators are often included in classification approaches as they order metaheuristics according to their capabilities which are related to their structure (Molina et al., 2020; Stork et al., 2020; Fausto et al., 2019; Stegherr et al., 2020). Furthermore, they are intended to facilitate the selection of an appropriate algorithm for a given problem, for example by matching the fitness function and the corresponding fitness landscape to algorithm classes (Stork et al., 2020). Ultimately, the algorithms should be classified by their performance on different optimisation problems (Woodward and Swan, 2010). This, however, requires detailed experiments to determine the respective algorithm performances.

Among those criteria used for the classification of metaheuristics, some directly relate to metaheuristic components and the resulting algorithmic features. These features can be related to the specific search procedures utilised, e. g. neighbourhood search, hill climbing or population-based search (Lones, 2014, 2019). They are, however, not necessarily determined by one single component or operator but can result from a specific combination of those. Additionally, one component and even one operator can utilise several search procedures. Other algorithmic features depending on these functional parts of a metaheuristic and describing the overall capabilities of the approach are presented by Chicco and Mazza (2020). They include, for example, the use of elitism, selection and self-adaptation strategies. Again, these features can result from a combination of operators and one operator can contribute to several features.

Another way to analyse metaheuristic components is provided by Blum and Roli (2003). They classify operators by their intensification and diversification behaviour in terms of its dependency on the objective function, any other function or randomness. In this case, operators are analysed individually, but can be compared to other operators for the same component.

A conceptual analysis of metaheuristic components and algorithm-specific operators according to features such as those presented here does not only allow for a better overview of metaheuristic strategies but also facilitates the selection of appropriate algorithms. It enables the construction of profiles for operators and their combinations within a component structure, which can be used for comparing different metaheuristics but also readily present the features that might be required for the problem at hand. Furthermore, in combination with problem-specific information, it directly aids in choosing a suitable algorithm.

Empirical Analysis

The goal of an empirical component analysis is to provide problem-specific and generalised information on the performance and behaviour of metaheuristics. It enables a more application-oriented comparison of algorithms than the conceptual analysis and complements it by presenting important knowledge for matching algorithms to optimisation problems as described by Woodward and Swan (2010).

To this end, the empirical analysis of metaheuristic components has to focus on measuring performance, as well as the search behaviour of the algorithms. Performance measures include the quality of the found solution and the budget (Halim et al., 2020), while behavioural measures are, e. g. the solution similarity and the intensification and diversification rates (Scheibenpflug et al., 2012). Furthermore, the operators of the respective components have to be evaluated in different combinations, as these combinations can exhibit mutually reinforcing effects on performance and behaviour. This is facilitated by the use of a unified framework instead of the individual metaheuristics, as well as the transfer of operators for one component from one algorithm to another. The overall analysis has to be performed according to benchmarking guidelines to provide valid results (LaTorre et al., 2020; Bartz-Beielstein et al., 2020).

A first evaluation in a unified framework with a focus on the performance of the algorithms is provided by Cruz-Duarte et al. (2020), showing that some operators can be responsible for the overall performance on a specific problem. Another empirical analysis of performance and intensification and diversification behaviour, focussed on different operator combinations of *Genetic Algorithms*, was performed by Scheibenpflug and Wagner (2013). They showed that the combination of operators can result in different behaviour than the individual operators would suggest. These studies show how much information and understanding on metaheuristic algorithms can be gained by analysing their components. Increasing the number of operators and combinations and extending the performance and behavioural measurements will provide further comprehensive insights.

Next to the general gain of knowledge, an empirical component analysis bridges the gap until theoretical proofs are presented for the behaviour and applicability of metaheuristics on specific problems. Furthermore, it can give hints as to which theoretical analyses are important to perform first. In terms of the overall analysis of metaheuristic components, it allows specifying and quantifying the conceptual analysis. Especially the utilisation of a unified structure for the evaluation of different component and operator combinations facilitates the analysis and comparison of metaheuristics independent of their frameworks and metaphors.

Advantages and Problems of a Component-based View on Metaheuristics

The analysis of metaheuristic components based on a unified framework provides several possibilities. First, it helps to bring structure to the vast field of metaheuristics by determining common features of the different approaches. This allows to systematically analyse metaheuristics according to the capabilities of their components and respective operators. Furthermore, standardising the description of metaheuristics by focussing on their components' features reduces the dependence on metaphors and provides a basis for the presentation of novel approaches. The empirical analysis adds further advantages. Problem-specific knowledge can be gained on the performance and behaviour of metaheuristics depending on their components, and in some cases operators may even prove to be generally well suited or not applicable at all. These analyses make it easier for Lifelike System engineers to evaluate whether the metaheuristics are a good fit for the given problem. Additionally, components and operators within a unified framework can provide an easy way to configure and change (e. g. hybridise) metaheuristic algorithms depending on the problem at hand, without having to construct each algorithm individually.

However, there still are some problems. It remains to be determined if a component analysis based on these concepts is feasible. This concerns the unification approach, which might not be viable for all metaheuristics, as well as the empirical analysis, which is extensive when aiming at analysing all or even most existing component structures and their respective operators. In addition, it might not be worth the effort if the often criticised strongly metaphor-related metaheuristics do not provide any new insights or useful features. For the conceptual analysis, it is still questionable how relevant the gathered information on features is for applications but also for comparisons. Furthermore, empirical studies that aim at providing problem-specific information require knowledge on the problems and their characteristics as well, which is still a research area needing attention. Altogether, the information gathered by this approach might not be sufficient to effectively facilitate the selection of appropriate algorithms, neither in Lifelike Systems nor for any other optimisation problem. Last but not least, no conceptual or empirical analysis is as good as a formal theoretical approach.

Conclusion

The field of metaheuristic research becomes less and less transparent in terms of new approaches and extensions to existing algorithms or frameworks. This makes it increasingly difficult to assess the features and capabilities of the respective algorithms and therefore the selection of a suitable approach. While this problem could be alleviated by extensive theoretical analyses of metaheuristics, this again is a difficult task requiring time and expertise.

Another approach to analyse metaheuristics more comprehensively while extensive theory is not yet available is described in this paper. It is based on a unified understanding of metaheuristics revealing common components. These components and their respective operators, which define individual algorithms, can be analysed conceptually as well as empirically. The analyses provide different levels of detail and, when combined, can offer a comprehensive view on metaheuristics. Furthermore, utilising a unified structure for metaheuristics presents an efficient basis for implementation, hybridisation and even the development of hyperheuristics, as operators can be exchanged easily before and during the optimisation process. Altogether, it facilitates the application of suitable metaheuristics in all domains, including Lifelike Systems.

The execution of metaheuristic component analyses in a unified framework can follow different approaches, as shown by Cruz-Duarte et al. (2020) or de Armas et al. (2021). However, none of them include extensive conceptual comparisons and empirical evaluations of performance and behaviour yet. We want to align our approach with the structure presented in this paper. To this end, we already examined classification systems to utilise their criteria for a conceptual analysis of metaheuristic components (Stegherr et al., 2020). In terms of empirical analysis, we established our basic unified structure and determined the experimental design necessary for empirical component analyses (Stegherr et al., 2021). The next steps include the examination of metaheuristics and their components and their incorporation into a unified framework. Furthermore, the respective operators will be reassembled to provide different combinations and to determine their influences in these combinations. Then, the combinations will be analysed conceptually to detect important common features, as well as in large experiments to evaluate their performance and behaviour on different optimisation problems. Though this approach can be extensive in terms of the number of included components and its feasibility for all metaheuristics still has to be shown, it will ultimately provide comprehensive information on metaheuristics and their applicability.

References

Bandaru, S. and Deb, K. (2016). Metaheuristic Techniques. In *Decision Sciences*, pages 693–750. CRC Press.

Bartz-Beielstein, T., Doerr, C., van den Berg, D., Bossek, J., Chandrasekaran, S., Eftimov, T., Fischbach, A., Kerschke, P., Cava, W. L., Lopez-Ibanez, M., Malan, K. M., Moore, J. H., Naujoks, B., Orzechowski, P., Volz, V., Wagner, M., and Weise, T. (2020). Benchmarking in Optimization: Best Practice and Open Issues. *arXiv:2007.03488*.

Blum, C. and Roli, A. (2003). Metaheuristics in combinatorial optimization. *ACM Computing Surveys*, 35(3):268–308.

Chicco, G. and Mazza, A. (2020). Metaheuristic Optimization of Power and Energy Systems: Underlying Principles and Main Issues of the 'Rush to Heuristics'. *Energies*, 13(19):5097.

Cruz-Duarte, J. M., Ortiz-Bayliss, J. C., Amaya, I., Shi, Y., Terashima-Marín, H., and Pillay, N. (2020). Towards a Generalised Metaheuristic Model for Continuous Optimisation Problems. *Mathematics*, 8(11):2046.

de Armas, J., Lalla-Ruiz, E., Tilahun, S. L., and Voß, S. (2021). Similarity in metaheuristics: a gentle step towards a comparison methodology. *Natural Computing*.

Fausto, F., Reyna-Orta, A., Cuevas, E., Andrade, Á. G., and Perez-Cisneros, M. (2019). From ants to whales: metaheuristics for all tastes. *Artificial Intelligence Review*, 53(1):753–810.

Halim, A. H., Ismail, I., and Das, S. (2020). Performance assessment of the metaheuristic optimization algorithms: an exhaustive review. *Artificial Intelligence Review*, 54(3):2323–2409.

Hussain, K., Salleh, M. N. M., Cheng, S., and Shi, Y. (2018). Metaheuristic research: a comprehensive survey. *Artificial Intelligence Review*, 52(4):2191–2233.

LaTorre, A., Molina, D., Osaba, E., Ser, J. D., and Herrera, F. (2020). Fairness in Bio-inspired Optimization Research: A Prescription of Methodological Guidelines for Comparing Meta-heuristics. *arXiv:2004.09969*.

Lones, M. A. (2014). Metaheuristics in nature-inspired algorithms. In *Proceedings of the 2014 conference companion on Genetic and evolutionary computation companion - GECCO Comp '14*. ACM Press.

Lones, M. A. (2019). Mitigating Metaphors: A Comprehensible Guide to Recent Nature-Inspired Algorithms. *SN Computer Science*, 1(49).

Molina, D., Poyatos, J., Del Ser, J., García, S., Hussain, A., and Herrera, F. (2020). Comprehensive Taxonomies of Nature- and Bio-inspired Optimization: Inspiration versus Algorithmic Behavior, Critical Analysis and Recommendations. *Cognitive Computation*.

Scheibenpflug, A. and Wagner, S. (2013). An Analysis of the Intensification and Diversification Behavior of Different Operators for Genetic Algorithms. In *Computer Aided Systems Theory - EUROCAST 2013*, pages 364–371. Springer Berlin Heidelberg.

Scheibenpflug, A., Wagner, S., Pitzer, E., Burlacu, B., and Affenzeller, M. (2012). On the analysis, classification and prediction of metaheuristic algorithm behavior for combinatorial optimization problems. *24th European Modeling and Simulation Symposium, EMSS 2012*, pages 368–372.

Song, Q. and Fong, S. (2016). Brick-Up Metaheuristic Algorithms. In *2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*. IEEE.

Sörensen, K. (2015). Metaheuristics-the metaphor exposed. *International Transactions in Operational Research*, 22(1):3–18.

Stegherr, H., Heider, M., and Hähner, J. (2020). Classifying Metaheuristics: Towards a unified multi-level classification system. *Natural Computing*.

Stegherr, H., Heider, M., Luley, L., and Hähner, J. (2021). Design of Large-Scale Metaheuristic Component Studies. To be published in the 2021 Genetic and Evolutionary Computation Conference Companion (GECCO '21 Companion).

Stork, J., Eiben, A. E., and Bartz-Beielstein, T. (2020). A new taxonomy of global optimization algorithms. *Natural Computing*.

Swan, J., Adriaensen, S., Bishr, M., Burke, E. K., Clark, J. A., De Causmaecker, P., Durillo, J., Hammond, K., Hart, E., Johnson, C. G., Kocsis, Z. A., Kovitz, B., Krawiec, K., Martin, S., Merelo, J. J., Minku, L. L., Özcan, E., Pappa, G. L., Pesch, E., Garcia-Sànchez, P., Schaerf, A., Sim, K., Smith, J., Stützle, T., Stefan, V., Wagner, S., and Yao, X. (2015). A Research Agenda for Metaheuristic Standardization. pages 1–3. Paper presented at MIC 2015: the XI Metaheuristics International Conference, Agadir, United Kingdom.

Sörensen, K. and Glover, F. W. (2013). Metaheuristics. In *Encyclopedia of Operations Research and Management Science*, pages 960–970. Springer US.

Wolpert, D. H. and Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82.

Woodward, J. R. and Swan, J. (2010). Why classifying search algorithms is essential. In *2010 IEEE International Conference on Progress in Informatics and Computing*. IEEE.