# The Parameterized Complexity of Multiparent Recombination

Carlos Cotta* Pablo Moscato†

*LCC-ETSII-UMA, Campus de Teatinos (3.2.49)
University of Málaga, 29071, Málaga, Spain
ccottap@lcc.uma.es

†School of Electrical Engineering and Computer Science
The University of Newcastle, Callaghan 2308, NSW, Australia
moscato@cs.newcastle.edu.au

## 1 Introduction

We introduce the first model for the computational complexity analysis of decision problems that arise in population-based metaheuristic design. In particular our hardness results could be linked to the practical difficulties of designing multiparent recombination metaheuristics in Evolutionary Algorithms and the path relinking recombination mechanisms that use multiple "elite" solutions in Scatter Search and other memetic algorithms. We expect that the new formalization will provide insights that will help to create more mathematically well founded exact or heuristic recombination algorithms aimed to solve a variety of associated combinatorial optimization problems. This has the similar spirit than the addition of *behaviors* had for parameterizing recombination operators [1], which we can now see as a instantiation of a more systematic and generic pattern for recombination design based on this new formulation of the problem. An $NP$-hard combinatorial optimization problem known as Min Feature Set (MFS) problem perfectly casts the issues involved in multiparent recombination algorithm design. In this paper we discuss it within its *parameterized complexity* membership.

Recombination is undoubtedly the major component of population-based metaheuristics. While its intuitive rôle has been always clear (to combine the "information" present in a set of solutions to create new solutions), the guidelines for designing practical recombination operators have experienced a remarkable evolution. First of all, nowadays it is increasingly more accepted that instead of directly manipulating the syntactic units used to encode solutions, the operator must extract *relevant* information from these solutions and recombine it (with independence of whether solutions are encoded on the basis of these particular information pieces or not). The Min Traveling Salesman (Min TSP) is a good example of this situation (the relevant information pieces would be "edges"). We will refer to these relevant "pieces of information" as *features*. These features of the solutions are also known as "attributes" in the Tabu Search and Scatter Search literature. We note, however, that in most of the cases where the original problem is intractable, these features of the solutions generally correspond to *predicates computable in polynomial-time* on the size of the instance of the original problem.

After having identified the relevant features (let us suppose we managed to find all features of a set of parent solutions in polynomial-time), the next and obviously important step is deciding how we can use this information. While *blind* recombination operators that randomly shuffle the set of features were more typical in the past, the addition of problem-domain knowledge to guide the process is becoming increasingly popular. The terms *hybrid* GAs and more generally *memetic algorithms* (MAs) [8] have been coined to denote these methods that use *smarter* reproductive operators and periods of single-agent optimization.

## 2 Smart Multiparent Recombination

There exist a plethora of mechanisms to create *smart* recombination operators, e.g., [12] but, up to this paper and to the best of our knowledge, no formalization and complexity results for some of the decision problems involved in multi-parent recombination have been reported. For instance, suppose we have a number of $k_{par} \geq 3$ tours from a relatively large population of size $Pop \gg k_{par}$. Let us also suppose that $k_{par} - 1$ of them have lengths values which are below the population's average length value, but one has a value well above average (to strengthen the argument we can even suppose that it is actually the longest tour in the population). While the preservation of edges/features present in all $m$ parents can still make some sense, we notice that the preservation of edges/features present in the best $k_{par} - 1$ parents *and not* present in the worst tour, seems also a valuable heuristic. Analogously, the *avoidance* of a feature present in the worst tour *and not* present in the other $k_{par} - 1$ tours is certainly another appealing heuristic for recombination. Associated *behaviors* can be usefully inferred to guide the recombination. Behaviors can be understood as a description of a preferred search direction within solution space or a model of what a good solution should have.

The previous example clearly depicts the existence of a more general problem: given a set of parent solutions, find the selection of an optimal set of features *to avoid* and *to preserve*. This problem already appears when we have parents that can be categorized in two different classes. A natural measure of optimality is the cardinality of the set, since we expect that $k_{par}$ is already a small number in comparison with the size of the instance, then we only expect to make a valid inference if the number of chosen features is also small.

In multi-parent recombination, we would wish to extract the behavioral pattern from the features of the other parent solutions in the current population. Subsequently, the recombination of information is "guided" by this behavioral pattern. For example, *rebel*, *obsequent*, and *conciliator* behaviors for two-parent recombination have been introduced in [1]. If we consider two parents $A = \{0,0,1,0,1\}$ and $B = \{1,0,1,1,0\}$ (assume that $A$ is a better solution than $B$), the recombination using these parents as input and each type of behavior is as follows. The 'x' stands for a value that will be decided by a repair-based algorithm aimed at recovering feasibility:

| | | | | | |
|---|---|---|---|---|---|
| rebel | {1 | x | x | 1 | 0} |
| conciliator | {x | 0 | 1 | x | x} |
| obsequent | {0 | x | x | 0 | 1} |

The names have been chosen with reference to the semantics of their preservation of information (allele values). The *conciliator* behavior is an example of a recombination procedure that *respects* features present in both parents (every child it produces contains all the gene values common to its two parents, i.e., those in $A \cap B$). It shares the property of being a *respectful* recombination as it is also the case of *uniform crossover*. In this case, since all alleles not in $A \cap B$ have either the value '0' or '1', the recombination with conciliator behavior is said also to be *transmitting* [11] (each gene value in the offspring is present in at least one of its parents). Note that the first behavior (rebel) bias the construction of descendants *away* from the best of the parents. The opposite is true for the second behavior (obsequent). Finally, the third behavior (conciliator) implies looking for descendants in the region "in between" the parents. The combined use of these behavior-based recombination operators has provided strong results for the MIN NUMBER PARTITIONING and seems suitable to be used in connection with MAs.

In some sense, behavior-based recombination can sometimes be seen as a particular case of multiparent recombination. The reason in some circumstances the behavioral pattern can specify a complete feasible solution or set of solutions. In addition to the potential use of these "behavioral" solutions as a source of "preferred" material, the main utility is in providing meta-information on how to select actual information from the parents. To illustrate the issue, given parents $A$ and $B$ as above, the *rebel* behavior can be understood from a multi-parent recombination process. First, construct all the parents whose Hamming distance with respect to $A$ (which is assumed to be the best parent) is 1, respecting genes in $A \cap B$. Assign to each of them an *attractor* status. The best parent $A$ is given a *repeller* status, while the worst parent $B$ is given an *attractor* status. Then, identify the *minimum cardinality feature set* that can explain the attractor and repeller nature of these parents.

| | | | | | |
|---|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 1 | repeller |
| 1 | 0 | 1 | 0 | 1 | attractor |
| 0 | 0 | 1 | 1 | 1 | attractor |
| 0 | 0 | 1 | 0 | 0 | attractor |
| 1 | 0 | 1 | 1 | 0 | attractor |

In this example, a minimum feature set requires columns 1, 4, and 5. Having identified these features, their values can for instance be copied from $B$ (a more precise definition of the FEATURE SET problem will be provided in next section).

No matter this connection, the relevance of finding an optimal feature subset to be regarded in multi-parent recombination will be more explicit in the next section. The process will be formalized as a combinatorial optimization problem. Then its complexity will be studied within the paradigm of parameterized complexity [7]. The main result of this paper is to show that while the general version of the multiparent recombination problem is not only "classically" intractable but is *parametrically* intractable in general, there is some hope that restricted versions of the problem might be *fixed-parameter tractable* (FPT). Some implications of this fact for the applicability of behavior-based recombination as well as for a more rather general form of multiparent recombination will be discussed.

## 3 Formalization of the Problem

Let us assume that the recombination operator is given a set of solutions to be recombined and a set of behavioral patterns. As mentioned above, the main objective of this latter set is to provide hints on which information should be taken from each parent as well as to indicate desirable/undesirable regions of the search space. This latter aspect can be addressed by assigning a tag to each solution in either set indicating whether it constitutes an *attractor* (1) or a *repeller* (0). The recombination process thus consists of determining which features are responsible for the attractor/repeller status of each solution. Subsequently, the values of these features (i.e., whether any of these features should be present or absent from the descendants) is chosen either matching any of the attracting parents, or avoiding values in any repellent parent. The first part of this definition of the recombination process perfectly matches the combinatorial problem known as $k$-FEATURE SET, and the decision version is here reformulated as follows:

- **Instance**: A set of $m$ examples $X = \{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$, such that for all $i$, $x^{(i)} = \{x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)}, t^{(i)}\} \in \{0,1\}^{n+1}$, and an integer $k > 0$.

- **Question:** Does there exist a *feature set* $S$, $S \subseteq \{1, \cdots, n\}$, with $|S| = k$ and such that for all pairs of examples $i \neq j$, if $t^{(i)} \neq t^{(j)}$ then exists $l \in S$ such that $x_l^{(i)} \neq x_l^{(j)}$ ?

This problem can be shown to be $NP$-complete by a reduction from VERTEX COVER [5]. This result would be commonly taken as a synonym of general intractability. However, such a statement should be regarded very carefully for two related reasons. On one hand, it is well-known that $NP$-hardness is often a *worst-case* scenario loosely related to what could be called a *typical* or *average* situation of solving instances of interest. Secondly, and more important, it is usually the case that these combinatorial problems have some kind of structural parameter. A complexity analysis based on this parameter can show that the problem is indeed perfectly tractable for a certain range of values for this parameter. The paradigm of parameterized complexity [7] was precisely created for this kind of analysis. This paradigm establishes a hierarchy of parameterized complexity classes $FPT \subseteq W[1] \subseteq W[2] \subseteq \cdots \subseteq W[t] \subseteq W[SAT] \subseteq W[P]$ that allows discriminating problems of different complexity according to the chosen parameter. For example, problems in the $FPT$ class (acronym that stands for *fixed-parameter tractable*) have algorithms whose worst-case complexity is $O(f(k)n^c)$, where $k$ is the parameter, $f(k)$ and arbitrary function of $k$ only, and $c$ is a constant. In contrast, the complexity of solving problems in $W[1]$ is $O(f(k)n^{g(k)})$, substantially harder in general.

A prototypical example of the $FPT$ class is the case of VERTEX COVER: if the size of the vertex cover is taken as a parameter, this problem can be shown to be in $FPT$ [6], existing algorithms for solving it in $O(1.2852^k + kn)$, i.e., linear in $n$ for fixed $k$, and polynomial in $n$ for $k \in O(\log n)$. This is achieved by combining the results of [3] and the speed-up method of [9]. Clearly, the $k$-FEATURE SET deserves a similar analysis. The next section is devoted to this. We have recently proved [4] the following claim:

**Theorem:** *The parameterized version of* FEATURE SET *in which the number of features is taken as a parameter is* $W[2]$*-complete.*

This is bad news for who may wish to implement an exact algorithm to optimally solve the MFS problem that naturally arises in multiparent recombination. Not only the problem is $NP$-hard, under the strong conjecture that $W[2] \neq FPT$, we can not assume that we can find a fixed-parameter tractable algorithm for this problem. As a consequence *the parameterized complexity of multiparent recombination is W[2]-hard*.

# 4 FPT Subclasses of Multiparent Recombination Algorithms

The One-Out Feature Set problem (OOFSP) is a special case of the Feature Set problem in which the Boolean column vector $T = \{t^{(1)}, t^{(2)}, \cdots, t^{(m)}\}$ (i.e., the rightmost column of $X$) has all identical values in all positions but one. Thus, the One-Out Feature Set problem models situations in which we have $m-1$ solutions that can be "attractors" and one "repeller" solution or *vice versa*. We have also been able to prove that:

**Proposition 1:** *The* One-Out Feature Set *problem is $NP$-complete and the* One-Out $k$-Feature Set *is $W[2]$-complete.*

We do have some good news, however. To identify amenable subclasses of OOFSP, let us consider the partitioning of OOFSP instances according to the equivalence relation *MaxRowWeight*. This equivalence relation groups into the same equivalence class those problem instances whose canonical form contains the same maximum number of 1s per row. Now, the following result hold:

**Proposition 2:** *The subclasses $d$*-MaxRowWeight OOFSP *are fixed parameter tractable if* $d$ is bounded by a constant.

The proposition above provides us the description of a tractable subclass of OOFSP, via reduction to $d$-Hitting Set. At present the best $FPT$ algorithm for the 3-Hitting Set problem is a linear time algorithm in the total number of features of complexity $O(2.311^k + n)$ proposed by Niedermeier and Rossmanith [10].

# 5 Conclusions

We have recently shown that one formalization of the multi-parent recombination problem is not fixed-parameter tractable unless a very unlikely condition holds. While this result severely limits the conditions under which we expect to efficiently find an optimal selection of features to be preserved or avoided from many parent solutions, our results and formalization encourage more research to find fixed-parameter tractable special cases.

While we have proved that minimum feature-based multiparent recombination is $W[2]$-hard in the general case, finding the smallest feature set for behavior-based and/or multiparent recombination is more amenable whenever the set comprising the parents and the behavioral solutions contains just one attractor or repeller, and some constraints regarding the number of 1s per row are respected. Notice also that this $FPT$ result is applicable to standard multiparent recombination (i.e., without any additional behavioral pattern or partial solution) if, for instance, we intend to generate solutions away from the worst parent (or, analogously, towards the best parent). It must also be noted that by computing feature sets, macro-units of information with context-free meaning are identified, and can be exchanged as a whole. This is very important when the problem/representation exhibits high epistasis.

Future work will be directed to study the complexity of other parameterized versions of multi-parent recombination. Other $FPT$ results may allow the use of an arbitrary set of attractors and repellers. These results may arise from the application of the notion of *bounded treewidth* [2] to this problem.

**Acknowledgements:** The authors would like to thank M. Fellows and V. Raman for their comments on an earlier version of this manuscript. C.C.'s work is partially supported by MCyT under grant TIC2002-04498-C05-02.

## References

[1] R. Berretta and P. Moscato. The number partitioning problem: An open challenge for evolutionary computation ? In D. Corne, M. Dorigo, and F. Glover, editors, *New Ideas in Optimization*, pages 261–278. McGraw-Hill, 1999.

[2] H.L. Bodlaender. Treewidth: Algorithmic techniques and results. In I. Privara and P. Ruzicka, editors, *Proceedings 22nd International Symposium on Mathematical Foundations of Computer Science*, volume 1295 of *Lecture Notes in Computer Science*, pages 29–36. Springer-Verlag, Berlin, 1997.

[3] J. Chen, I.A. Kanj, and W. Jia. Vertex cover: further observations and further improvements. *Journal of Algorithms*, 41:280–301, 2001.

[4] C. Cotta and P. Moscato. The $k$-FEATURE SET problem is $W[2]$-complete. *Journal of Computer and System Sciences*, 67(4):686–690, 2003.

[5] S. Davies and S. Russell. $NP$-completeness of searches for smallest possible feature sets. In R. Greiner and D. Subramanian, editors, *AAAI Symposium on Intelligent Relevance*, pages 41–43, New Orleans, 1994. AAAI Press.

[6] R. Downey and M. Fellows. Fixed parameter tractability and completeness I: Basic theory. *SIAM Journal of Computing*, 24:873–921, 1995.

[7] R. Downey and M. Fellows. *Parameterized Complexity.* Springer-Verlag, 1998.

[8] P. Moscato and C. Cotta. A gentle introduction to memetic algorithms. In F. Glover and G. Kochenberger, editors, *Handbook of Metaheuristics*, pages 105–144. Kluwer Academic Publishers, Boston MA, 2003.

[9] R Niedermeier and P. Rossmanith. A general method to speed up fixed-parameter-tractable algorithms. *Information Processing Letters*, 73:125–129, 2000.

[10] R. Niedermeier and P. Rossmanith. An efficient fixed parameter algorithm for 3-HITTING SET. *Journal of Discrete Algorithms*, 1(1):89–102, 2003.

[11] N.J. Radcliffe. The algebra of genetic algorithms. *Annals of Mathematics and Artificial Intelligence*, 10:339–384, 1994.

[12] N.J. Radcliffe and P.D. Surry. Fitness variance of formae and performance prediction. In L.D. Whitley and M.D. Vose, editors, *Foundations of Genetic Algorithms III*, pages 51–72, San Mateo CA, 1994. Morgan Kauffman.