ISSN: 1803-3814

THREE-DIMENSIONAL GENETIC ALGORITHM FOR THE KIRKMAN’S SCHOOLGIRL PROBLEM

Jan Merta, Tomáš Brandejský

University of Pardubice
Faculty of Electrical Engineering and Informatics
Department of Software Technologies
Studentská 95, 532 10 Pardubice
Czech Republic
jan.merta@student.upce.cz
tomas.brandejsky@upce.cz

Abstract: *This paper focuses on the possibilities of multidimensional genetic algorithms and relevant genetic operators. After the literature overview we used a three-dimensional genetic algorithm to solve a combinatorial task called Kirkman’s Schoolgirl Problem. The first results were not good, but we improved convergence of an algorithm by adding more genetic operators. We also used problem dependent mutation, where we tried to repair the incorrect solution by using the simple heuristic method. Finally, we got a solid number of correct solutions, but we know there is enough room for other improvements.*

Keywords: *multidimensional genetic algorithms, three-dimensional genetic algorithm, multidimensional chromosome, Kirkman’s Schoolgirl Problem*

1 Introduction

The basic version of the genetic algorithm uses a fixed length linear one-dimensional chromosomal encoding. Most problems can be transformed into this representation but sometimes this approach is too complicated for certain problems. In some cases, we lose important information about the problem or linkage of related genes [1]. In other cases, the use of one-dimensional chromosomal encoding leads to the loss of important information about the problem. For example, mapping graphs onto a linear chromosome may lead to the loss of potentially important information about the adjacency of the graph vertices. A special form of a two-dimensional chromosome with real points in a two-dimensional Euclidean space was used to resolve the Travelling Salesman Problem. This led to the preservation of information about the spatial position of individual cities in the article [2].

According to the schema theorem, it is important for the genetic algorithm that the schema has a small length (the related genes are close together). This is because the smaller length of a schema increases the resistance of the schema to the crossover and mutation, and therefore increases the probability of preserving the scheme in the next generation (decreases disruption of a schema) [3]. Multidimensional problems are ideal candidates for the using of a chromosome in the form of a matrix (two-dimensional array) or higher order tensor (n-dimensional array). Multidimensional chromosomes can not only bring more natural and clear interpretation of these problems, but also better performance of the genetic algorithm (due to better linkage of related genes) [1]. For some problems, the linear one-dimensional chromosome coding does not have sufficient performance [4, 5].

Using a multidimensional chromosome may be suitable for solving combinatorial problems in which regular groups of genes occur. They can also be utilized for solving spatial and topological problems (the use of chromosome in the form of an adjacency matrix, incidence matrix etc.). Another interesting area of application is the design of technical systems which we would like to focus on in further research. Good illustrative examples of the use of multidimensional genetic algorithms can be also tasks of recreational mathematics. For the Eight Queens problem we can use the two-dimensional chromosome that naturally represents the structure of the chessboard (for multidimensional variant of problem this problem we can use multidimensional array). Another interesting task of recreational mathematics is Kirkman’s Schoolgirl Problem where we chose to test the genetic algorithm with multidimensional chromosome. Both of these problems have practical applications. N-Queen's problem has interesting applications in many areas [6]. Generalized form of the Kirkman's Schoolgirl Problem is used in the design of experiments [7].

The first part of this article focuses on the possibilities of encoding individuals using the two-dimensional and multidimensional chromosomes and the relevant genetic operators. The second part describes the Kirkman's Schoolgirl Problem. The third part of the article deals with our use of genetic algorithms for the Kirkman’s problem. It also describes our experiments. Finally, we added an evaluation of our experiments and a summary of the achieved results and acquired knowledge.

2 Genetic algorithms with multidimensional chromosome

2.1 Encoding

While a simple linear one-dimensional chromosome can be represented by a one-dimensional array (the vector of its individual genes), in the case of a two-dimensional chromosome, it can be a two-dimensional array (matrix of its individual genes). In the n-dimensional case, it is analogous to the n-dimensional array (tensor of rank n). The chromosome alphabet can be composed of binary numbers, but also real numbers or any characters (as well as a one-dimensional version).

If the spatial properties of individual genes are to be better preserved, an alternate form of the chromosome, which reflects position of points in a space, can be used. In a one-dimensional version, the individual genes reflect the position of the points on a line. In a two-dimensional and three-dimensional version, the individual genes reflect the position of points in real Euclidean space [8].

2.2 Operators

Crossover usually works with two parental chromosomes and divides them into two or more groups of genes. The offspring inherits some groups of genes from the first parental chromosome and remaining groups of genes from the second parental chromosome. Unlike the crossover and mutation of one-dimensional chromosomes, the possibilities of crossover and mutation of the multidimensional chromosomes can be intuitively more diverse. While a one-dimensional chromosome can be divided into two (or more) groups of genes by a non-dimensional cutting point, the two-dimensional chromosome can be divided into groups by a crossover point or one-dimensional cutting line. This line can divide the two-dimensional array horizontally or vertically. The cutting lines need not be perpendicular to the coordinate axes. We can use more complex curves to divide the two-dimensional (or multidimensional) chromosome into groups of genes [2, 4].

One of the first papers dealing with multidimensional representations is an article [9] written by Cohoon and Paris. The authors use a rectangle-style crossover of two-dimensional chromosomes. For the purposes of a crossover, a small rectangle (submatrix of chromosomal matrix) from the parental chromosome is selected. Genes from this rectangle are copied to the offspring and remaining genes are inherited from the other parent.

The crossover of multidimensional chromosomes is also described in the article [10]. Balázs describes a one-point crossover in two and three dimensions. In the two-dimensional variant, the x and y coordinates of the cutting point are generated. The cutting point divides the chromosome into four parts (1-4 in the clockwise direction). The first offspring inherits the genes from the first and third parts of the first parental chromosome, and from the second and fourth parts of the second parental chromosome. The second offspring inherits the genes in the reverse manner. The procedure is best illustrated in the figure. One-point crossover in three dimensions is analogous. The author also adds a mathematical description of his method using schema theorem.

Anderson and his co-workers in their article on the *Ising problem* designed a uniform crossover generalized for two-dimensional chromosomes. They called this special type of crossover the *block-uniform crossover*. This crossover divides the parental chromosome into *i x j* blocks of genes with randomly generated dimensions. Individual blocks of the first parent chromosome are exchanged with the corresponding blocks of the second parent according to the given probability. The authors also tested two-point crossover, uniform crossover, vertical band crossover, horizontal band crossover, block crossover and mixed crossover, but proposed that the block-uniform crossover had the best performance for solving the *Ising problem* of all tested operations.

The *geographic crossover* proposed in the article [4] by Kahng and Moon promises better diversity of genetic operators. If the *r x r* grid is encoded by a linear chromosome of length r^2, there are r^2*-1* ways to divide the chromosome. But if the *r x r* grid is naturally encoded by an *r x r* chromosome, there are only *2*(r-1)* ways to divide the chromosome by horizontal and vertical cutting lines. This is a disadvantage because the two-dimensional has smaller diversity of crossover operators. Despite this disadvantage, the multidimensional chromosome overcomes linear representation in solving some problems, due to better linkage of the genes [5].

Crossing only by vertical or horizontal cutting lines also has the problem of preserving the more complex structures of related genes in the next generation. The proposed framework solves the problem of smaller a diversity of genetic operators. It also the preserves the more complex structures of related genes. The multidimensional chromosome is divided into subspaces by cutting surfaces. The simplified two-dimensional variant randomly selects two sides (they can't be the same), for example, the top and right side. There are six combinations (top-right, top-left, top-bottom, left-right, left-bottom, right-bottom). A random point is chosen on each of the two selected sides and, from these points, random curves are developed simultaneously until the two curves meet and create a final cut. Up to k cutting surfaces are chosen and the whole procedure is a multidimensional generalization of a one-dimensional k-point crossover [4]. Another version of the multidimensional k-point crossover called Z3 was proposed by Bui and Moon [8].

3 The Kirkman's Schoolgirl Problem

The *Kirkman's problem* (The Fifteen Schoolgirls Problem) published by T. P. Kirkman in 1850 (in The Lady's and Gentleman's Diary) is a combinatorial puzzle about fifteen schoolgirls who take a daily walk [11].

Definition. The problem is to arrange them into five triplets for seven days in a week, so that no girl will walk with any of her school-fellows in any triplet more than once. The problem can be generalized for *n* girls (where n is divisible by three) into triplets to walk out for *d* days (where *d = (n-1)/2*) [11].

Kirkman's problem is part of *combinatorial design*, and it is a special example of the *Steiner triple system* with additional restrictive conditions. When working on the school problem, Kirkman laid the foundations of *block design*, which became the basis of *design theory* and *design of experiments* [7].

Individual solutions of Kirkman's problem can be easily described using a three-dimensional array (*DAY x GROUP NUMBER x GIRL NUMBER*). Thanks to its natural structure, the problem becomes an ideal candidate for testing a genetic algorithm based on a multidimensional chromosome with restrictive conditions.

4 Methodology

The aim of the experiment was to use a multidimensional genetic algorithm to generate one solution for *Kirkman's schoolgirl problem* (15 schoolgirls, 5 triplets, 7 days). For the experiments we used three-dimensional chromosomal encoding in the form of 7x5x3 where values of genes were numbers of individual girls.

For the experiment, we have implemented some methods of crossover and mutation working with the three-dimensional chromosome.

We tested the crossing methods:

- **Corner block crossover:**
  In 3D chromosomal space the point with coordinates *i, j* and *k* is generated. The offspring inherits the genes that are at the coordinates $x < i$, $y < j$ and $z < k$ from one parent, the remaining genes are from the other parent.
- **One-point three-dimensional crossover:**
  The same crossover, as proposed by Balázs, which we described in the chapter on multidimensional crossover methods. The generated point with the coordinates *i, j* and *k* divides the chromosome into 8 parts.
- **Classic uniform crossover:**
  The three-dimensional chromosome is linearly iterated across the individual genes and a classical uniform crossover occurs.
- **Band crossover:**
  Two parents swap the whole day (matrix) between each other.

We also tested the mutation methods like:

- **Smart intergroup mutation:**
  On a random day, girls are swapped between two random groups.
- **Inversion:**
  Inverts the order of days within the entire week or inverts the order of the groups of girls within the day.
- **Random shuffle:**
  Randomly mixes days within the week or groups of girls within the day.
- **Random repair:**
  Finds duplicate pairs of girls. It tries to fix the solution by replacing random duplicate pairs with pairs that are not present in the chromosome.

The mutation takes place directly on a concrete individual, and another individual is not needed.

Individuals were randomly generated, so that every girl would appear in every day. Complete randomness was also tested. The fitness function has been defined as a number of errors consisting of:

- number of **duplicate pairs**,
- the sum of the **number of missing girls** in individual days.

The goal was to minimize this number of errors to **zero**.

The Stochastic Universal Sampling was selected as the selection mechanism. Population size was tested in sizes of 5, 10 and 50. We used the correction mechanism, which corrected the chromosome so that in each day, each of the girls would appear once. We also set elitism value to 1 individual. For illustrative purposes, the genetic algorithm with a one-dimensional chromosome was tested by linearizing the three-dimensional chromosome into a vector of 105 values.

5 Results

The performance of the individual configurations was measured as a percentage of the converged individuals until the given generation limit (50,000, 100,000). In every experiment we ran the genetic algorithm 1000 times. We used the Stochastic universal sampling as a selection mechanism.

The convergence of a one-dimensional genetic algorithm, with a basic one-point (or k-point) crossover and simple algorithm mutation, was zero. We also tried a basic three-dimensional genetic algorithm with a corner block crossover and smart intergroup mutation. This variant had zero convergence too. Changing the crossover to a one-point three-dimensional crossover gave us similar results. We added inversion to the list of genetic operators and the first interesting results appeared. Configuration with a one-point three-dimensional crossover, smart intergroup mutation, and inversion, gave us 1% convergence (10 correct solution on 1000 runs of algorithm until 100,000th generation). The size of this population was 10. We also tried a population size of 50, but it didn't give us any good results.

We tried the configuration with all crossovers (corner block crossover, one-point 3D crossover, classic uniform crossover and band crossover), smart intergroup mutation, inversion, and the population size of 5. It gave us 2% convergence (until 100,000 generations). When we tried to add random repair mutation, convergence increased to 4 %. We also tried to add the last genetic operator called random shuffle. This change of configuration increased the convergence of the genetic algorithm to 6 %. We also tried only a generation limit 50,000 for this configuration, and convergence was 1.5 %. Finally, we decreased the population limit to 30,000. This change gave us 1.5% convergence. The results of our experiments with three-dimensional genetic algorithm are shown at Table 1 in more readable form.

Table 1: Results of experiments with 3D GA

Crossovers	Mutations	Convergence (%)
corner block	smart intergroup	0
one-point 3D	smart intergroup	0
one-point 3D	smart intergroup inversion	1
corner block one-point 3D classic uniform band	smart intergroup inversion	2
corner block one-point 3D classic uniform band	smart intergroup inversion random repair	4
corner block one-point 3D classic uniform band	smart intergroup inversion random repair random shuffle	6

Next lines represent the example of correct solution generated by genetic algorithm after 66414 generations (stochastic universal sampling, population size of 5, all operators, 1 elite individual).

(13 03 02) (10 14 00) (05 09 01) (06 11 08) (04 07 12)
(07 11 02) (10 13 08) (04 05 00) (01 06 14) (12 03 09)
(01 12 10) (03 14 04) (00 11 13) (07 06 05) (09 08 02)
(04 06 02) (12 08 00) (10 09 07) (01 11 03) (13 14 05)
(14 08 07) (06 10 03) (05 11 12) (13 04 09) (01 00 02)
(14 02 12) (03 08 05) (00 06 09) (07 01 13) (04 11 10)
(06 12 13) (11 09 14) (00 07 03) (08 01 04) (10 02 05)

This individual was generated at 13th run of genetic algorithm with best configuration and its convergence is shown at Fig. 1.

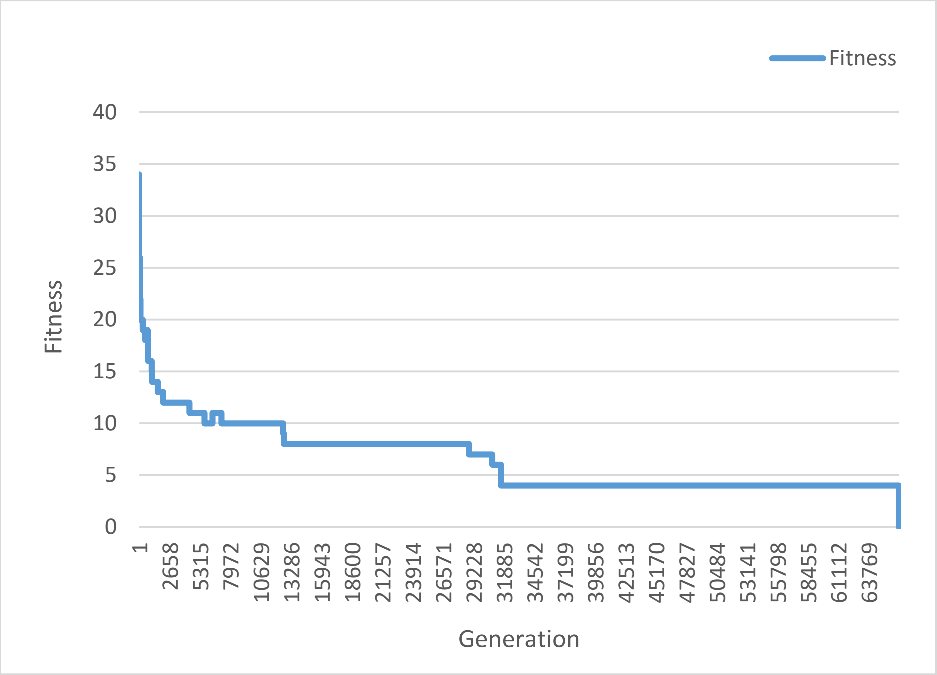

Figure 1 - Convergence of three-dimensional algorithm for the Kirkman's problem

6 Discussion

The basic configuration of the algorithms (1D and 3D) with simple a one-point crossover and simple mutation gave no results. Three-dimensional genetic algorithms with basic operations were a bit better, but it's hard to compare, because the difference is too small, and the experiments were too few. It was necessary to add additional genetic operators.

Different types of crossovers are good, but without mutation operators they can't reach the correct solution of the problem. When we added another genetic operator, convergence of the algorithm improved. The combination of all crossover types and all mutations gave best results. It confirms the positive impact of greater diversity of operators. The disadvantage of using multiple operators is that the setting of all the parameters of individual genetic operators is highly complicated. We also tried to decrease the population limit for best configuration, and we got good results. It gave us some correct solutions until a generation limit of 50,000 and even until a generation limit of 30,000.

We found that small populations have better performance for this problem. A size of 5 (or 10) is ideal for different configurations. Larger populations (especially size of 50) with a combination of multiple operators brought a worsening of convergence (10-20 on average). This can be explained by a positive effect of higher evolutionary pressure.

We found that an algorithm often gets stuck at a local minimum (fitness with value 3-5) and it remains there until the generation limit. This small fitness value means that there are only few errors (sometimes the only one error) in the solution, but the algorithm cannot resolve these final steps in order to find a correct solution. It would be good to think of a new heuristic method to solve this problem. It is also possible that we make some mistakes in the choice of software or algorithm implementation.

7 Conclusion

In this article, we have firstly done an overview on existing research in the field of genetic algorithms with multidimensional chromosomes. Then we tried the possibilities of a three-dimensional genetic algorithm with a set of simple genetic operators to find one solution of the Kirkman's schoolgirl problem. Gradually, our tuning setup succeeded in increasing the convergence of the genetic algorithm. Finally, after a solid number of generations, we have found several correct solutions to a complex combinatorial problem with restrictive conditions. It should be noted that at this time, the experiments only served to improve the configuration of the genetic algorithm and its convergence. So, at this stage we cannot fully compare particular configurations against each other. For a more accurate comparison of individual genetic operators and configurations, multiple experiments with more runs of the genetic algorithm are needed. The repeatability of the results and the reasonable percentage of converged individuals were also important. The use of more tested genetic operators finally solved the problem of too little convergence of the algorithm.

In the future, we need to think about improvements that lead to more stable results. We would like to use the findings from the experiments to solve other interesting problems. We would like to use the knowledge about multidimensional genetic algorithms to create an algorithm for designing technical systems.

Acknowledgement: The work has been supported by the Funds of University of Pardubice, Czech Republic. This support is very gratefully acknowledged.

References

[1] Bui, T. N., and Moon, B. R.: On Multi-Dimensional Encoding/Crossover. ICGA, pp. 49–56. (2002)

[2] Jung, S., and Moon, B. R.: The Natural Crossover for the 2D Euclidean TSP. In Genetic and Evolutionary Computation Conference, pp. 1003–1010. (2000)

[3] Langdon, W. B., and Poli, R.: Foundations of Genetic Programming. Springer (2002). ISBN 3540424512.

[4] Kahng, A. B., and Moon, B. R: Toward More Powerful Recombinations. In International Conference on Genetic Algorithms, pp. 96-103 (1995)

[5] Anderson, C., Jones, K., and Ryan, J.: A two-dimensional genetic algorithm for the Ising problem. Complex Systems, pp. 327-333 (1991)

[6] Sosic, R., and Gu, J.: Efficient local search with conflict minimization: a case study of the n-queens problem. In IEEE Transactions on Knowledge and Data Engineering, vol. 6, no. 5, pp. 661-668 (1994)

[7] Colbourn, C. J. and Dinitz, J. H.: Handbook of combinatorial designs. Second edition. Boca Raton: Taylor & Francis Group, Discrete mathematics and its applications (2007) ISBN 978-1-58488-506-1.

[8] Seo, D. I., and Moon, B. R.: A survey on chromosomal structures and operators for exploiting topological linkages of genes. In Genetic and Evolutionary Computation Conference (2003)

[9] J. Cohoon and D. Paris. Genetic placement. In IEEE International Conference on Computer-Aided Design, pp. 422-425 (1986)

[10] Balázs, M.-E.: Multidimesional Crossover in Genetic Algorithms. Artificial Neural Networks and Genetic Algorithms, Springer (1998)

[11] Ball Rouse, W. W. and, Coxeter, H. S. M.: Mathematical recreations and essays. 13th ed. Dover Publications, Dover books on mathematical and word recreations (1987) ISBN 0-486-25357-0.