Threshold matrix generation for digital halftoning by genetic algorithm optimization

Jarmo T. Alander, Timo Mantere and Tero Pyylampi

Department of Information Technology and Production Economics
University of Vaasa, PO Box 700, FIN-65101 Vaasa, Finland

ABSTRACT

Digital halftoning is used both in low and high resolution high quality printing technologies. Our method is designed to be mainly used for low resolution ink jet marking machines to produce both gray tone and color images. The main problem with digital halftoning is pink noise caused by the human eye's visual transfer function. To compensate for this the random dot patterns used are optimized to contain more blue than pink noise. Several such dot pattern generator threshold matrices have been created automatically by using genetic algorithm optimization, a non-deterministic global optimization method imitating natural evolution and genetics. A hybrid of genetic algorithm with a search method based on local backtracking was developed together with several fitness functions evaluating dot patterns for rectangular grids. By modifying the fitness function, a family of dot generators results, each with its particular statistical features. Several versions of genetic algorithms, backtracking and fitness functions were tested to find a reasonable combination. The generated threshold matrices have been tested by simulating a set of test images using the Khoros image processing system. Even though the work was focused on developing low resolution marking technology, the resulting family of dot generators can be applied also in other halftoning application areas including high resolution printing technology.

Keywords: backtracking, blue noise, combinatorial optimization, dithering /threshold matrices, FM screening, genetic algorithms, halftoning, ink jet printers

1. INTRODUCTION

Digital halftoning, also called spatial dithering, is a rendering method used to display continuous tone pictures on displays capable of producing a very limited number, usually only two, different tone picture elements, here called dots. Tones are thus realized by average local dot densities. The main problem in halftoning is how to place dots so that the picture does not contain artifacts such as moiré or dot clusters caused by dot placement. Moiré can be eliminated by aperiodic dot placement, but the price to be paid for this is the tendency of aperiodic dot patterns to contain visually annoying dot clusters (pink spatial noise).

Our application is designed mainly for low resolution ink jet marking machines[1] to produce both gray tone and color images. The marking machine is used as a "label in demand" device partly replacing higher quality but less flexible printed labels on shipping cartons used in just-in-time production or addressing direct-mail pieces.

1.1. Halftoning Methods

The simpliest technique of producing aperiodic dot patterns is to use (pseudo) random dithering i.e. uncorrelated white noise patterns. The problem with this, usually totally unsatisfactory, method is the clustering of dots at all gray levels causing a disturbing grainy appearance. To overcome this several methods called order dithering have been developed. The dots should be distributed as evenly as possible. These methods are attractive because of their speed and simplicity. One generally used realization called recursive tessalation arrays suffers from strong periodic structure. In this paper an optimization based method is developed to much reduce the magnitude of this periodicity. This paper presents the algorithm for generating recursive tessalation arrays and discusses possibilities to use optimization techniques in the design of dithering methods in general.

Other author information: (Send correspondence to J. T. Alander)
E -mail: *FirstName.LastName*@uwasa.fi

The average distance between neighboring pixels λ_g, called the principal wavelength, can be defined as

$$\lambda_g = \begin{cases} |v|/\sqrt{g}, & 0 \leq g \leq 1/2 \\ |v|/\sqrt{1-g}, & 1/2 < g \leq 1 \end{cases}$$

where $g \in [0,1]$ is the gray level and $|v|$ is the output grid constant. The object of optimization is now to place dots so that the spatial frequency, that is inversely proportional to the distance between dots, contains minimum contribution at low frequences i.e. at the long distance end. The currently most popular method to realize this so called blue noised dithering is called error diffusion or minimum average error, in which negative feedback is used as a low frequency inhibitor. Error diffusion was first introduced by Floyd and Steinberg in 1975. It requires neighborhood operations and is thus more computationally intensive than single pixel based methods.

The concept of blue noise was introduced by Ulichney to describe patterns containing only small amounts of energy at low spatial frequences. He also developed efficient halftoning algorithms for dithering with blue noise and metrics for analyzing the frequency content of aperiodic patterns. The algorithms are based on perturbed error diffusion and shown to be superior on rectangular grids.[2]

The dot patterns generated by error diffusion methods are pleasingly isotropic and mostly structureless. However, there are some shortcomings: some correlated artifacts, directional hysteresis and transient behavior near edges and boundaries. The edge sharpening can be most conveniently controlled by first high-pass filtering of the image using e.g. a Laplacian and then halftoning it. Ulichney strongly recommends to keep the halftoning method and sharpening decoupled because they are mutually somewhat compensating and thus interfere with each other.[2]

In psychological terms blue noise is perceived as not to contain any structure i.e. interesting features and it is thus least visually annoying dot pattern.

1.2. Genetic Algorithms

Genetic algorithms (GA) are non-deterministic global optimization methods imitating genetics and natural evolution from the point of view of information processing. Problem parameters are encoded usually as binary vectors called chromosomes after their natural counterparts. A set of these parameter vectors are maintained in an array called population. New trial vectors are generated by selecting random parent vectors and recombining randomly their values in an operation "naturally" called crossover by GA researchers. Some noise is usually added to parameter values every now and then simulating mutations in DNA. Evolution in this highly random process is due to selection of best solution candidates to be held in a usually fixed size population array. Selection is done according to a so called fitness value i.e. the value of the function to be optimized. The most natural problems to be solved by GAs are such that they are difficult to be solved by more traditional methods or for which there does not yet exit any good special algorithm. It should be kept in mind that a hybrid combining GA and some heuristics is usually worth considering. Typically combinatory, integer, and/or discontinuous problems contain many potentially promising problem types to be attacked by GAs. The most prominent good properties of GAs are their flexibility, generality and problem independence, while the corresponding weaknesses include slow processing, nondeterminism and lack of rigorous theory of convergence. For more information on details of GAs see e.g. Refs. 3 and 4. For references to basics of GAs see bibliography (Ref. 5).

1.3. Related Work

In the void-and-cluster method introduced by Ulichney generated dot patterns are postprocessed by searching clusters and voids of dots and swapping corresponding central dot pairs. This simple and sound heuristics leads to very uniform dot patterns.[6] For a review of digital halftoning methods see Refs. 7 or 8. For more references on dithering and halftoning see Hull's bibliography (Ref. 9).

According to our quite complete genetic algorithm bibliography, which currently contains already over 10.000 entries, there are over 600 references to image processing and optics optimization related problems including pattern recognition, image reconstruction, segmentation, and filter design.[10] For a review of GAs in image processing see Ref. 11. Kobayashi and Saito from Keio University seem to be the first to have applied genetic algorithms to halftoning.[12–16] Recently also Newbern and Bove from MIT Media Laboratory have applied genetic algorithms to halftoning.[17]

The dithering problem has some similarity to the ancient magic square problem, which has been solved also by GAs.[18] One of the GA demonstrations of EvoNet Flying Circus is a magic square solver.[19]

```
void  binaryToPermutation(int* CHR, int* P, int n)
//    convert the binary genotype CHR into a permutation index
//    vector P, which together with M forms the phenotype,
//    on which the fitness function is evaluated
{
     int  i=0,j=0;
     for (i=0;i<Xmax*Ymax;i++) P[i] = i;     // initialize permutation array P
     for (i=0; i<n; i+=2) {                   // swap elements of P
        int p=CHR[i],tmp=P[p];
        P[p]=P[i]; P[i]=tmp;
     }
}
```

Figure 1. Routine `binaryToPermutation` to transform an integer (binary) valued chromosome vector `CHR` into a permutation index vector `P` of length `n`.

2. HALFTONING METHOD

Our halftoning method is based on several criteria implemented as a fitness function. The aim is to distribute dots evenly at all gray levels by avoiding clusters. In this respect it resembles Ulichney's void-and-cluster method.[6] Our method is more general, however, because in addition to the dot cluster criteria we can easily add other optimization criteria by modifying the object function. This is facilitated by the flexibility of GA based optimization, which is within reasonable limits much independent of the object function to be evaluated.

2.1. Genetic Algorithm

The population size of the genetic algorithm varied in the range $[1, 100]$. The smaller values seeming to lead to somewhat faster convergence. GAs are known to be quite insensitive to their parameters such as the population size.[4] The parameter vector or chromosome was encoded to contain permutation information of the generator matrix.

2.1.1. Parameter encoding

The dithering method is based on a threshold matrix, which contains $n \times n$ threshold values. In our case we have used 8×8, 16×16 and 32×32 matrices. The problem is to find such a permutation of the given matrix elements that the object function is optimal. Using conventional immediate encoding is not a good idea, because binary crossover and mutation operators interfere with permutations. In practise this means that some building blocks i.e. threshold values are rapidly lost, if we are not carefully guarding the recombination process, which is both tedious and error prone. However, there is not any fundamental problem in the GA nor in the problem, but in our trivial encoding scheme, which tries to cover too much using a simple and explicit binary encoding. This is the same as if organisms consisted of only DNA i.e. the genotype (= the encoding) and the phenotype (= the creature) were the same, which obviously is not the case.

By clearly dividing the genotype and the phenotype by introducing a proper data structure and encoding we can easily handle also permutations or other similar and actually heavily constrained problems. In GA literature there are given a number of solutions to this problem especially in the context of the famous traveling salesman problem[20] (TSP).[21–35]

We have used the following simple, yet general encoding: Permutations are represented in the chromosome by genes (integers) `P[i]` meaning that elements indexed (linearly) by integers `i` and `P[i]` in the threshold matrix `M` are swapped. The swapping was done by a permutation vector `P` as shown in figure 1.

For more details of the GA used in this work see report,[36] which also includes the source code written in C.

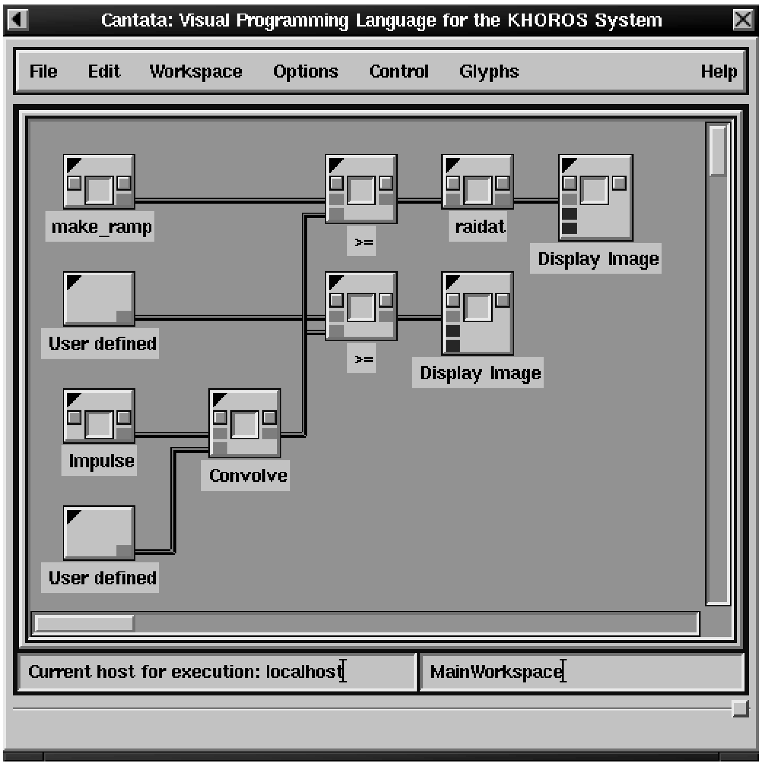

Figure 2. The image halftoning testing setup using Khoros: gray scales (top) and user defined images (bottom).

2.1.2. Fitness function and cluster metric

There are two clearly different goals for the dot pattern generation:

1. the average density of the dot pattern should interpolate as the original pixel values
2. the dot pattern spectrum should be skewed towards high frequences (blue noise)

both goals should be met while keeping the dot pattern randomly distributed in order to avoid artifacts.

3. RESULTS

Only preliminary simulation testing of the resulting dot patterns on a set of test images has been done. The simulations were done using the Khoros image processing system.[37] The test setup is shown in figure 2.

In the test runs the number of fitness function evaluations was ranging from appr. 40.000 to about 10^6.

4. DISCUSSION

4.1. Halftoning

The results of our preliminary test and those by other groups seem to indicate that genetic algorithm optimization can be used to solve such combinatorial problems as generation of threshold matrices. But is the method adequate for low resolution printing at all? The good property of the method is its simplicity and thus fast processing by the output device. The situation totally changes if the image is rendered dot by dot well before the printing process by a computer. E.g. an ink jet printer has only a few large dots and we can place every single dot by using GAs and/or other optimization methods. In that case the fitness function should measure the error between the inputted and outputted images, or their filtered versions.

Figure 3. Two original 512×512 pixel and 256 tone test images.

Figure 4. Test images halftoned by GA optimized threshold matrix.

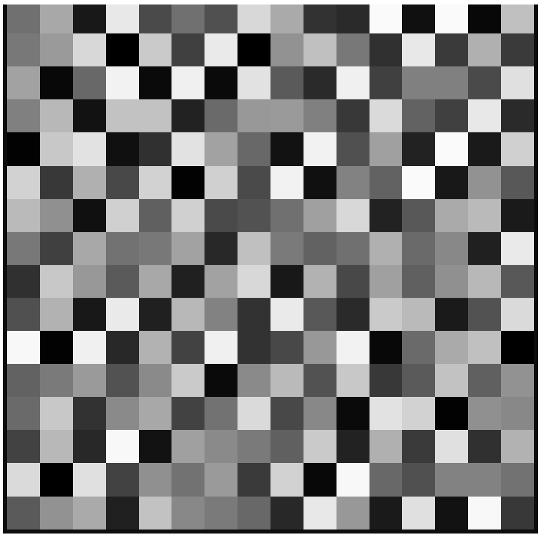

Figure 5. A close up view of the threshold matrix used to generate the images shown in figure 4.

We should not smooth sharp edges too much and we should not create artificial edges where such do not exist in reality. It is apparent that the fitness function should somehow be designed so that both these contradictory views are included. Obviously this situation can be seen as a classification problem, where pixels are grouped into non-edge and a set of pixels having more or less a clear edge character.

Yet another approach to online halftoning is a fast routine or hardware function that decides dot placement using a window of input pixels.[38] In that approach genetic programming[39] could be used to automatically produce needed function or hardware structure.

4.2. Optimization by Genetic Algorithm

It turned out that the threshold matrix design is a difficult optimization problem. The problem has a lot of parameters, the values of which can be arranged in very many ways i.e. the search space is incredibly large. The autocorrelation length of the fitness functions tend to be very short. After much testing using different fitness functions, optimization algorithms and their combinations, it was evident that a hybrid approach combining a kind of backtracking strategy and genetic algorithm using a quite small population size was performing usually the fastest. A similar combinatorial problem having many parameters, but solved best by a small population size GA can be found also in Ref. 40. Backtracking itself is a good and much used deterministic search strategy in combinatorial problems. In its general (global) form its main drawback is that the number of evaluations grows rapidly with the problem size. In our case we have limited the steps by allowing only a few random steps before returning to the best (or one of the best) encountered solutions up to that time. This might be similar to protein evolution in nature. Very seldom seems totally new protein structures appear. What is happening instead is evolution of proteins by point mutations: an amino acid every now and then is replaced by another, while the well performing overall structure is conserved over millions of generations. Protein fitness landscape is also known to be very rugged and difficult to search.[41] This analogy suggests considering population initialization: the better structures to start with the better results can be expected. In our case magic squares were used as the starting point of further optimization.

5. CONCLUSIONS AND FUTURE

The preliminary results and the corresponding results in the literature seem to indicate that the genetic algorithm optimization approach is one alternative to be considered when developing a digital halftoning method. Its most prominent feature is its flexibility: by modifying the object function we can generate tailored halftoning patterns to meet special requirements.

Because the main constraint on the dot pattern is the number of dots in each pixel location, the generator should operate so that the number of dots is invariant i.e. all the operations applied keep the total number of dots fixed. The main operation fulfilling this constraint is swapping of areas having the same shape. In our case special permutation encoding of chromosomes was used instead of immediate encoding to meet this constraint.

Potential further research topics include a more general genetic programming[39] (GP) type approach to automatically generate procedures that generate dot patterns according to input image data. One alternative is to use GP to generate a fast routine or even a digital circuit to do halftoning for applications requiring utmost processing speed such as motion video rendering.

More material on this work can be found at our ftp site `ftp.uwasa.fi` in directory `cs/report98-1`.[36]

Acknowledgement

This project is a part of a larger project applying GAs in industrial optimization problems including statistical testing of embedded software by GAs.[42] and has been funded by the Finnish Technology Development Centre (Tekes) and FBM Limited Oy. Prof. Matti Linna and prof. Caj Södergård are acknowledged for their helpful comments during the work and Lilian Grahn for her help with the proofreading of this paper.

REFERENCES

1. Trident, "Ultrajet (TM) printhead," 1998. `http://www.tridentintl.com/htm/Product`.
2. R. A. Ulichney, "Dithering with blue noise," *Proceedings of the IEEE* **76**, pp. 56–79, Jan. 1988.
3. J. T. Alander, "On finding the optimal genetic algorithms for robot control problems," in *Proceedings IROS '91 IEEE/RSJ International Workshop on Intelligent Robots and Systems '91*, vol. 3, pp. 1313–1318, IEEE Cat. No. 91TH0375-6, (Osaka), 3.-5. Nov. 1991.
4. J. T. Alander, "On optimal population size of genetic algorithms," in *CompEuro 1992 Proceedings, Computer Systems and Software Engineering, 6th Annual European Computer Conference*, P. Dewilde and J. Vandewalle, eds., pp. 65–70, IEEE Computer Society, IEEE Computer Society Press, (The Hague), 4.-8. May 1992.
5. J. T. Alander, "Indexed bibliography of genetic algorithms basics, reviews, and tutorials," Report 94-1-BASICS, University of Vaasa, Department of Information Technology and Production Economics, 1995. (`ftp.uwasa.fi:cs/report94-1/gaBASICSbib.ps.Z`)
6. R. A. Ulichney, "The void-and-cluster method for dither array generation," in *Human Vision, Visual Processing, and Digital Display IV*, J. P. Allebach and B. E. Rogowitz, eds., vol. SPIE-1913, pp. 332–343, The International Society for Optical Engineering, Bellingham, WA, (San Diego, CA), Feb. 1993.
7. R. A. Ulichney, *Digital Halftoning*, MIT Press, Cambridge, MA, 1987.
8. W. H. Banks, ed., *Advances in Printing Science and Technology*, Pentech Press Publishers, London, 1993.
9. D. Hull, "Dithering bibliography," 1997. (`http://pertsserver.cs.uiuc.edu/~ hull/halftone/dither.html`).
10. J. T. Alander, "Indexed bibliography of genetic algorithms in optics and image processing," Report 94-1-OPTICS, University of Vaasa, Department of Information Technology and Production Economics, 1995. (`ftp.uwasa.fi:cs/report94-1/gaOPTICSbib.ps.Z`)
11. C. Bounsaythip and J. T. Alander, "Genetic algorithms in image processing - a review," in Alander,[44] pp. 173–192, (`ftp.uwasa.fi:cs/3NWGA/Bounsaythip.ps.Z`)
12. N. Kobayashi and H. Saito, "Halftone algorithm using genetic algorithm," in *Proceedings of the Fourth International Conference on Signal Processing Applications and Technology*, vol. 1, pp. 727–731, DSP Associates, Newton, MA, (Santa Clara, CA), 28. Sept.- 1. Oct. 1993.
13. N. Kobayashi and H. Saito, "Half-toning technique using genetic algorithm," in *Proceedings of the 1994 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-94)*, vol. 5-I, pp. 105–108, IEEE, New York, (Adelaide (Australia)), 19.-22. Apr. 1994.
14. H. Saito and N. Kobayashi, "Evolutionary computation approaches to halftoning algorithm," in *Proceedings of the First IEEE Conference on Evolutionary Computation*, vol. 2, pp. 787–791, IEEE, New York, NY, (Orlando, FL), 27.-29. June 1994.
15. N. Kobayashi and H. Saito, "Halftoning technique using genetic algorithm," *Transactions of the Institute of Electronics, Information and Communication Engineers A (Japan)* **J78D-2**(10), pp. 1450–1459, 1995. (in Japanese; also in English as[16])

16. N. Kobayashi and H. Saito, "Halftoning technique using genetic algorithms," *Systems and Computers in Japan* **27**, pp. 89–97, Sept. 1996. (English translation of[15])
17. J. Newbern and V. M. Bove, Jr., "Generation of blue noise arrays by genetic algorithm," in *Human Vision and Electronic Imaging II*, B. E. Rogowitz and T. N. Pappas, eds., vol. SPIE-3016, pp. 441–450, SPIE- Int. Soc. Optical Engineering, Bellingham, (San Jose, CA), 10.-13. Feb. 1997.
18. D. H. Ardell, "TOPE and magic squares: A simple GA approach to combinatorial optimization," in *Genetic Algorithms at Stanford 1994*, J. R. Koza, ed., Stanford Bookstore, (Stanford, CA), Fall 1994.
19. T. Fogarty, "Evonet," brochure, Napier University, 1996.
20. E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy, and D. B. Shmoys, eds., *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, John Wiley & Sons, New York, 1985.
21. I. M. Oliver, D. J. Smith, and J. R. C. Holland, "A study of permutation crossover operators on the traveling salesman problem," in *Genetic Algorithms and their Applications: Proceedings of the Second International Conference on Genetic Algorithms and Their Applications*, J. J. Grefenstette, ed., pp. 224–230, Lawrence Erlbaum Associates: Hillsdale, New Jersey, (MIT, Cambridge, MA), 28. - 31. July 1987.
22. B. P. Buckles, F. E. Petry, and R. L. Kuester, "Schema survival rates and heuristic search in genetic algorithms," in *Proceedings of the 1990 IEEE International Conference on Tools for Artificial Intelligence TAI'90*, A. Dollas and N. G. Bourbakis, eds., pp. 322–327, IEEE Computer Society Press, Los Alamitos, CA, (Herndon, VA), 6.-9. Nov. 1990.
23. F. Q. Bac and V. L. Perov, "New evolutionary genetic algorithms for NP-complete combinatorial optimization problems," *Biological Cybernetics* **69**(3), pp. 229–234, 1993.
24. C. C. Klimasauskas, "Genetic algorithm optimizes 100-city route in 21 minutes on a PC!," *Advanced Technology for Developers* **2**, pp. 9–17, Feb. 1993.
25. K. Shahookar, W. Khamisani, P. Mazumder, and S. M. Reddy, "Genetic beam search for gate matrix layout," *IEE Proceedings, Computers and Digital Techniques* **141**, pp. 123–128, Mar. 1994.
26. P. W. Poon and J. N. Carter, "Genetic algorithm crossover operators for ordering applications," *Computers & Operations Research* **22**(1), pp. 135–148, 1995.
27. S. Chatterjee, C. Carrera, and L. A. Lynch, "Genetic algorithms and traveling salesman problems," *European Journal of Operational Research* **93**, pp. 490–510, 20. Sept. 1996.
28. C. Bierwirth, "A generalized permutation approach to job shop scheduling with genetic algorithms," *OR Spektrum* **17**(2-3), pp. 87–92, 1995.
29. A. L. Patton, . William F. Punch, and E. D. Goodman, "A standard GA approach to native protein conformation prediction," in *Proceedings of the Sixth International Conference on Genetic Algorithms*, L. J. Eshelman, ed., pp. 574–581, (Pittsburgh, PA), 15.-19. July 1995.
30. P. Robbins, "The use of a variable length chromosome for permutation manipulation in genetic algorithms," in Pearson *et al.*,[43] pp. 144–147.
31. N. Sangalli, Q. Semeraro, and T. Tolio, "Performance of genetic algorithms in the solution of permutation flowshop problems," in Pearson *et al.*,[43] pp. 495–498.
32. Y. K. Kim, C. J. Hyun, and Y. Kim, "Sequencing in mixed model assembly lines: a genetic algorithm approach," *Computers & Operations Research* **23**(12), pp. 1131–1145, 1996.
33. C. Bierwirth, D. C. Mattfeld, and H. Kopfer, "On permutation representations for scheduling problems," in *Parallel Problem Solving from Nature – PPSN IV*, H.-M. Voigt, W. Ebeling, I. Rechenberg, and H.-P. Schwefel, eds., vol. 1141 of *Lecture Notes in Computer Science*, pp. 310–318, Springer-Verlag, Berlin, (Berlin (Germany)), 22.-26. Sept. 1996.
34. M. Yagiura and T. Ibaraki, "Use of dynamic programming in genetic algorithms for permutation problems," *Eur. J. Oper. Res.* **92**(2), pp. 387–401, 1996.
35. G. Ucoluk, "A method for chromosome handling of r-permutations of n-element set in genetic algorithms," in *Proceedings of 1997 IEEE International Conference on Evolutionary Computation*, pp. 55–58, IEEE, New York, NY, (Indianapolis, IN), 13.-16. Apr. 1997.
36. J. T. Alander, T. Mantere, and T. Pyylampi, "Threshold matrix generation for digital halftoning by genetic algorithm optimization," Research Report 98-1, University of Vaasa, Department of Information Technology and Production Economics, 1998. (to appear partly in SPIE-3522; `ftp.uwasa.fi:cs/report98-1/Halftoning.ps.Z`)

37. Khoral Research, Inc., "Home page of Khoral Research, inc., creator of Khoros technology," 1998.
38. P. W. Wong, "Entropy-constrained halftoning using multipath tree coding," *IEEE Transactions on Image Processing* **6**, pp. 1567–1579, Nov. 1997.
39. J. R. Koza, *Genetic Programming: On Programming Computers by Means of Natural Selection and Genetics*, The MIT Press, Cambridge, MA, 1992.
40. J. T. Alander and M. Ryynänen, "Magnetic field refinement: Genetic algorithms vs. hill-climbing?," in Alander,[44] pp. 333–340. (`ftp.uwasa.fi:cs/3NWGA/Alander4.ps.Z`)
41. S. A. Kauffman, *The Origins of Order, Self-Organization and Selection in Evolution*, Oxford University Press, New York, 1993.
42. J. T. Alander, T. Mantere, G. Moghadampour, and J. Matila, "Searching protection relay response time extremes using genetic algorithm – software quality by optimization," in *Proceedings of the Fourth International Conference on Advances in Power System Control, Operation & Management (APSCOM-97)*, vol. 1, pp. 95–99, IEE (Hong Kong), (Hong Kong), 11.-14. Nov. 1997. (`ftp.uwasa.fi:cs/report97-5/HongKong.ps.Z`)
43. D. W. Pearson, N. C. Steele, and R. F. Albrecht, eds., *Artificial Neural Nets and Genetic Algorithms*, (Alès (France)), Springer-Verlag, Wien New York, 19.-21. Apr. 1995.
44. J. T. Alander, ed., *Proceedings of the Third Nordic Workshop on Genetic Algorithms and their Applications (3NWGA)*, (Helsinki (Finland)), Finnish Artificial Intelligence Society (FAIS), 18.-22. Aug. 1997. (`ftp.uwasa.fi:cs/3NWGA/*.ps.Z`)