Mixed-Integer Optimization of Coronary Vessel Image Analysis using Evolution Strategies

Rui Li
Natural Computing Group
LIACS
Leiden University
Leiden, The Netherlands
ruili@liacs.nl

Michael T.M. Emmerich
Natural Computing Group
LIACS
Leiden University
Leiden, The Netherlands
emmerich@liacs.nl

Jeroen Eggermont
Division of Image Processing
Department of Radiology C2S
LUMC
Leiden, The Netherlands
j.eggermont@lumc.nl

Ernst G.P. Bovenkamp
Division of Image Processing
Department of Radiology C2S
LUMC
Leiden, The Netherlands
e.g.p.bovenkamp@lumc.nl

ABSTRACT

In this paper we compare Mixed-Integer Evolution Strategies (MI-ES) and standard Evolution Strategies (ES) when applied to find optimal solutions for artificial test problems and medical image processing problems. MI-ES are special instantiations of standard ES that can solve optimization problems with different objective variable types (continuous, integer, and nominal discrete). Artificial test problems are generated with a mixed-integer test generator. The practical image processing problem iss the detection of the lumen boundary in IntraVascular UltraSound (IVUS) images. Based on the experimental results, it is shown that MI-ES generally perform better than standard ES on both artifical and practical image processing problems. Moreover it is shown that MI-ES can effectively improve the parameters settings for the IVUS lumen detection algorithm.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search; G.1.6 [**Numerical Analysis**]: Optimization

General Terms

Algorithms, Design, Experimentation

Keywords

Evolution Strategies (ES), Mixed-Integer Evolution Strategies (MI-ES),IntraVascular UltraSound (IVUS)

GECCO'06, July 8–12, 2006, Seattle, Washington, USA.
Copyright 2006 ACM 1-59593-186-4/06/0007 ...$5.00.

1. INTRODUCTION

We study Mixed-Integer Evolution Strategies (MI-ES) [6], a variant of Evolution Strategies (ES) [11], as a solution method for difficult (high-dimensional, non-linear) optimization problems where different types of decision variables have to be considered simultaneously. In particular we look at problems where the decision vector consists of continuous, ordinal and nominal discrete variables. Standard Evolution Strategies only take into account homogeneously typed variable vectors (typically continuous). MI-ES are extensions of Evolution Strategies that, instead of simply truncating continuous variables, use specific operators for different types of discrete variables. The MI-ES approach, adopted in this paper, was proposed in [6] and combines variation operators for different variable types as introduced in [2] (binary variables),[10] (integer variables) and [11] (continuous variables). For similar approaches in the domain of genetic algorithms we refer to [5]. The results of our study will underpin the benefit of the new approach. For studying the MI-ES an artificial test problem with scalable ruggedness and a challenging real-world problem will be considered. As real-world problem we will tackle a problem related to medical image analysis, namely the optimization of algorithms for the detection of features in intravascular ultrasound (IVUS) image sequences.

IVUS images show the inside of coronary or other arteries and are acquired with an ultrasound catheter positioned inside the vessel. An example of an IVUS image with several detected features can be seen in Figure 1. IVUS images are difficult to interpret which causes manual segmentation to be highly sensitive to intra- and inter-observer variability[8].

In addition, manual segmentation of the large number of IVUS images per patient is very time consuming. Therefore an automatic system is needed. However, feature detectors consist of a large number of parameters that are hard to optimize manually and may differ for different interpretations. Moreover, these parameters are subject to change when something changes in the image acquisition process, and there are continuous as well as different types of discrete parameters involved. Encouraged by previous work [4, 3] on optimization of image segmentation algorithms in the medical domain, we consider Evolutionary Algorithms (EA) as a solution method. However, unlike these previous approaches we will use Evolution Strategies rather then Genetic Algorithms, since they seem to be more suitable for dealing with continuous parameters.

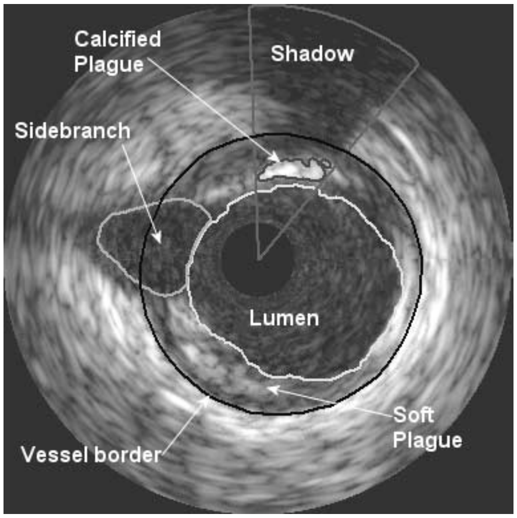

Figure 1: An IntraVascular UltraSound (IVUS) image with detected features. The black circle in the middle is where the ultrasound imaging device (catheter) was located. The dark area surrounding the catheter is called the *lumen*, which is the part of the artery where the blood flows. Above the catheter *calcified plague* is detected which blocks the ultrasound signal causing a dark *shadow*. Between the inside border of the vessel and the lumen there is some soft plague, which does not block the ultrasound signal. The dark area left of the catheter is a sidebranch.

Our paper is organized as follows: Section 2 provides details about the application problem. Mixed-Integer Evolution Strategies (MI-ES) are introduced in section 3. Serials of tests on artificial problems for MI-ES are the subject of section 4. Both MI-ES and standard ES are applied to the parameters optimization of lumen feature detector; the experimental results will be shown in section 5. The paper closes with an outlook providing some ideas for the future.

2. IVUS IMAGE ANALYSIS

IntraVascular UltraSound (IVUS) is a technique used to get real-time high resolution tomographic images from the inside of coronary vessels and other arteries. To gain insight into the status of an arterial segment a so-called catheter pullback sequence is carried out. A catheter (∅ ±1mm) with a miniaturized ultrasound transducer at the tip is inserted into a patient's artery and positioned downstream of the segment of interest. The catheter is then pulled back in a controlled manner, using motorized pullback (1 mm/s), during which images are acquired continuously.

In [1] a state-of-the-art multi-agent system is used to detect lumen, vessel, shadows, sidebranches and calcified plagues in IVUS images. The system, shown in Figure 2, is based on the cognitive architecture Soar (States, operators and results) [9]. IVUS image processing agents interact with each other through communication, act on the world by controlling and adapting image processing operations and perceive that same world by accessing image processing results.

Agents thereby dynamically adapt the parameters of low-level image segmentation algorithms based on knowledge of global constraints, contextual knowledge, local image information and personal beliefs. The lumen-agent, for example, encodes and controls an image processing pipeline which includes binary morphological operations, an ellipse-fitter and a dynamic programming module, and it determines all relevant parameters. Generally, agent control allows the underlying segmentation algorithms to be simpler and to be applied to a wider range of problems with a higher reliability.

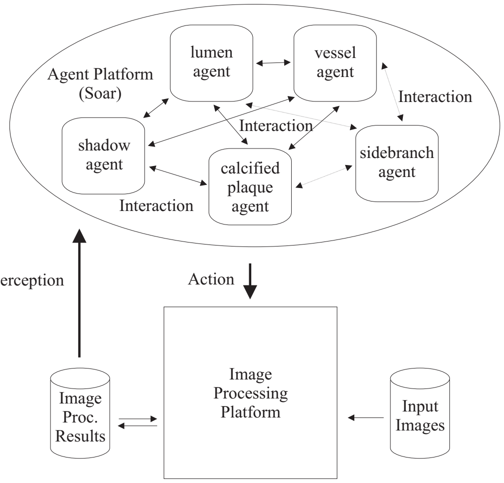

Figure 2: Global view of the multi-agent system architecture.

Although the multi-agent system has shown to offer lumen and vessel detection comparable to human experts [1], it is designed for symbolic reasoning, not numerical optimization. Further it is almost impossible for a human expert to completely specify how an agent should adjust its feature detection parameters in each and every possible interpretation context. As a result an agent has only control knowledge for a limited number of contexts and a limited set of feature detector parameters.

In addition, this knowledge has to be updated whenever something changes in the image acquisition pipeline. Therefore, it would be much better if such knowledge might be acquired by learning the optimal parameters for different interpretation contexts automatically.

3. OPTIMIZATION ALGORITHM

One of the main reasons why standard optimization methods cannot be effectively used in our application, is that different types of parameters are involved in the lumen feature detector. More precisely, those types can be classified as the following:

- **Continuous variables:** These are variables that can change gradually in arbitrarily small steps.

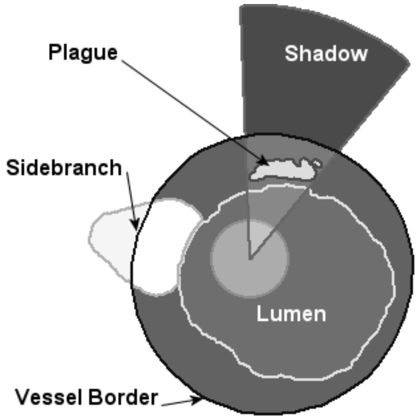

Figure 3: Schematic version of Figure 1 as detected by the multi-agent image segmentation system.

- **Ordinal discrete variables:** These are variables that can be changed gradually but there are smallest steps (e.g. integer quantities).
- **Nominal discrete variables:** These are discrete parameters with no reasonable ordering (e.g. binary decisions, discrete choices from an unordered list/set of alternatives).

Correspondingly, the optimization of lumen feature detector may be formalized in the following way:

$$f(r_1, \dots, r_{n_r}, z_1, \dots, z_{n_z}, d_1, \dots, d_{n_d}) \rightarrow \min \qquad (1)$$

$$\text{subject to:}$$

$$r_i \in [r_i^{min}, r_i^{max}] \subset \mathbb{R},\ i = 1, \dots, n_r$$

$$z_i \in [z_i^{min}, z_i^{max}] \subset \mathbb{Z},\ i = 1, \dots, n_z$$

$$d_i \in D_i = \{d_{i,1}, \dots, d_{i,|D_i|}\}, i = 1, \dots, n_d$$

Here $r_i, i = 1, \dots, n_r$ denote continuous variable, $z_i, i = 1, \dots, n_z$ are integer variable and d_i are discrete variable taken from pre-described sets $D_i, i = 1, \dots, n_d$, respectively. f denotes the objective function to be minimized. In addition, a metric penalty approach [7] is introduced in the objective function that grows proportional with the number of constraint violations and makes sure that infeasible solutions are dominated by all feasible solutions.

3.1 Representation and Selection

Mixed-integer evolution strategies (MI-ES) were first proposed in Emmerich et al. for chemical engineering plant optimization with process simulators from industry [6]. Unlike other Evolution Strategies, MI-ES can deal simultaneously with continuous, integer, and nominal discrete variables and therefore it is especially well suited for solving the given image analysis problem.

In the MI-ES, individuals are represented as tuples taken from the search space $\mathbb{S}$

$$\vec{a} = (r_1, \dots, r_{n_r}, z_1, \dots, z_{n_z}, d_1, \dots, d_{n_d}, \\ \sigma_1, \dots, \sigma_{n_\sigma}, \zeta_1, \dots, \zeta_{n_\zeta}, p_1, \dots, p_{n_p})$$

$(r_1, \dots, r_{n_r}, z_1, \dots, z_{n_z}, d_1, \dots, d_{n_d})$ are called *object parameters.* $(\sigma_1, \dots, \sigma_{n_\sigma}, \zeta_1, \dots, \zeta_{n_\zeta}, p_1, \dots, p_{n_p})$ denote the *strategy parameters* or *step-size*, and their explicit explanations are that $\sigma_1, \dots, \sigma_{n_\sigma}$ are average step-size for the continuous variables, $\zeta_1, \dots, \zeta_{n_\zeta}$ are average step-size for the ordinal discrete variables, and $p_1, \dots, p_{n_p}$ are mutation probabilities for the nominal discrete variables.

The generational loop of the MI-ES reads as follows: After random initialization and evaluation of μ individuals [1], λ offspring individuals are generated through a recombination and a mutation operator. Then the fitness function is used to evaluate these λ offspring. Next, the selection operator chooses the μ best individuals among those λ offspring individuals and μ parental individuals that do not exceed the maximal life-span (age). Usually, life-span, or age is expressed using κ, and $\kappa = 1$ corresponds to a (μ, λ)-selection and $\kappa = \infty$ to a $(\mu+\lambda)$ selection. As long as the termination criterion[2] is not fulfilled, the μ selected individuals form the parental generation for the next iteration loop. The detailed description of the main loop of the MI-ES can be found in [6]. To allow for an automatic step-size adaptation it is recommended in [11] to set $\lambda/\mu \gtrsim 7$ and use a *comma*-strategy.

3.2 Mutation and recombination for different variable types

An ad-hoc approach for using the continuous ES for integer optimization is to just truncate the continuous values after their mutation to the next integer value. However, a conceptual drawback of this approach would be that the step-size might reduce to a value that is too small to generate any improvement. Even worse, in case of nominal discrete values the implicit assumption of a neighborhood could cause the ES to converge to an artificial local optimum.

Thus, unlike the selection operator, the mutation operator is different in the MI-ES than in the standard ES, because it takes into account the different types of decision variables. Noting, that there are similar approaches in the realm of genetic algorithms [5], we here adopt and further investigate an approach suggested by Emmerich et al. [6]. Algorithm 1 describes the mutation procedure in detail: For the local and global step-size learning rates τ_l and τ_g we use the recommended parameter settings $\tau_l = \frac{1}{\sqrt{2\sqrt{n_r}}}$ and $\tau_g = 1/\sqrt{2n_r}$

Algorithm 1 Mutation procedure in MI-ES

```
$N_g = N(0, 1)$: normal distributed random variable
for $i = 1, \dots, n_r$ do
  $\sigma_i' \leftarrow \sigma_i \exp(\tau_g \mathrm{N}_g + \tau_l \mathrm{N}(0, 1))$
  $r_i' = r_i + N(0, \sigma_i')$
end for
for $i = 1, \dots, n_z$ do
  $\zeta_i' \leftarrow \zeta_i \exp(\tau_g \mathrm{N}_g + \tau_l \mathrm{N}(0, 1))$
  $z_i' \leftarrow z_i + \mathrm{G}(0, \zeta_i')$
end for
$p_i' := 1/[1 + \frac{1-p_i}{p_i} * \exp(-\tau_l * \mathrm{N}(0, 1))]$
for $i \in \{1, \dots, n_d\}$ do
  if $\mathrm{U}(0, 1) < p_i'$ then
    $d_i' \leftarrow$ uniformly randomly value from $D_i$
  end if
end for
```

[1] uniform distribution within the corresponding ranges

[2] typically a maximal number of generations

[6]. Note, that in case a single step-size is chosen, $\tau_l = 0$. In the algorithm $\mathrm{N}(0,1)$ denotes a function that results in a standard normal distributed random number. Accordingly, $\mathrm{U}(0,1)$ denotes a function returning a uniformly distributed number in $[0,1] \subset \mathbb{R}$ and $\mathrm{G}(0,q)$ returns a geometrically distributed random value.

Among these distributions, the geometrical distribution deserves further attention, as it is rarely referred to in literature. Rudolph [10] proposed the geometrical distribution for the mutation of integer vectors within the ES. He provided various reasons for choosing this distribution: Geometrical distributed random variables are random variables whose values are in $\mathbb{Z}$. They have properties similar to normal distributed random variables in $\mathbb{R}$. In particular, they have infinite support in $\mathbb{Z}$, are unimodal with a peak in 0, and their probability function is symmetric in the origin. Moreover, as pointed out by [10], multivariate extensions are characterized by a rotational symmetry with regard to the ℓ_1 norm[3] and they belong to a family of maximal entropy distributions. Finally, by increasing the value q the standard deviation of the random variable can be gradually increased. All these characteristics make the geometric distribution well suited for application within mixed-integer evolution strategies.

The recombination operator we used is similar to the standard recombination in ES. For the object variables we used uniform crossover (discrete recombination) and for the step-size variables intermediate recombination. Since uniform crossover is well defined for discrete variable types, no adaptations have been considered here.

A geometrically distributed random variable G can be generated from two uniformly distributed random variables $u_1 := U(0,1); u_2 := U(0,1)$ via:

$$\mathrm{G} = G_1 - G_2,\ p = 1 - \frac{q/n_z}{1+ \ \overline{1+(\frac{q}{n_z})^2}},$$

$$G_i = \left\lfloor \frac{\ln(1-u_i)}{\ln(1-p)} \right\rfloor, i = 1,2 \tag{2}$$

The mutation of the mutation probabilities is done by means of a logistic distribution as described in [6].

To make sure that variables stay within their respective boundaries we have added some routines for interval treatment to the MI-ES. While for the continuous variables we used reflection at the boundary, for the integer variables we set the value to the bound, whenever the bound is exceeded. The latter method is also used to keep the mutation probabilities within bounds.

4. TESTS ON ARTIFICIAL PROBLEMS

4.1 Test functions

In order to select a favorable variant of the MI-ES for the time-consuming runs on the IVUS image analysis problem, we study the behavior of the MI-ES and standard ES (continuous variables are truncated to integer values) on barrier problems using a new problem generator. We designed a multimodal barrier problem generator that produces integer optimization problems with a scalable degree of ruggedness (determined by parameter C) by generating an integer array A using the following algorithm:

```
$\mathtt{A}[i] = i, i = 0, \dots, 20$
for $k \in \{1, \dots, C\}$ do
    $j \leftarrow$ random number out of $\{0, \dots, 19\}$
    swap values of $\mathtt{A}[j]$ and $\mathtt{A}[j+1]$
end for
```

Then a barrier function is computed:

$$f_{barrier}(\mathbf{r}, \mathbf{z}, \mathbf{d}) = \sum_{i=1}^{n_r} \mathtt{A}[\lfloor r_i \rfloor]^2 + \sum_{i=1}^{n_z} \mathtt{A}[z_i]^2 + \sum_{i=1}^{n_d} \mathtt{B_i}[d_i]^2 \rightarrow \min \tag{3}$$

$$n_r = n_z = n_d = 5, \mathbf{r} \in [0,20]^{n_r} \subset \mathbb{R}^{n_r},$$
$$\mathbf{z} \in [0,19]^{n_z}, \mathbf{d} \in \{0, \dots, 19\}^{n_d} \tag{4}$$

Here, $B_i[0], \dots, B_i[19]$ denotes a set of i permutations of the sequence $0, \dots, 19$, each of which being randomly chosen before the run. This construction prevents that the value of the nominal value d_i is quantitatively (anti-)correlated with the value of the objective function f. Such a correlation would contradict with the assumption that d_i are nominal values. Whenever a correlation between neighboring values can be assumed it is wiser to assign them to the ordinal type and treat them accordingly.

The parameter C controls the ruggedness of the resulting function with regard to the integer space. High values of C result in rugged landscapes with many barriers. To get an intuition about the influence of C on the geometry of the function we included plots for a two-variable instantiation of the barrier function in Figure 4 for $C = 20$ and $C = 100$.

4.2 Experimental Results

We used the following MI-ES settings for the experiments on the barrier problems: $\{(\mu = 3, \lambda = 10), (\mu = 4, \lambda = 28), (\mu = 15, \lambda = 100)\}$ for the population and offspring sizes and $\{(n_\sigma = n_\zeta = n_p = 1), (n_\sigma = n_r, n_\zeta = n_z, n_p = n_d)\}$ for the step-size mode.

From Figures 5 and 6, it turns out that the $(\mu = 4, \lambda = 28)$ setting performs best on the more difficult problems. However, the plots on the barrier problem show that for long runs with $t \gg 2000$ a strategy with a larger population size might be favorable. We also observed that it is less risky with regard to the performance of the strategy to use a single step-size per parameter type $(n_\sigma = n_\zeta = n_p = 1)$, instead of individual step-sizes $(n_\sigma = n_r, n_\zeta = n_z, n_p = n_d)\}$ for all of the variables. This corresponds to the findings in [6].

We also compared MI-ES to standard ES. The results displayed in Figure 7 ($C = 20$) and Figure 8 ($C = 100$) show that the $(\mu = 4, \lambda = 28)$ MI-ES perform better than standard $(\mu = 4, \lambda = 28)$ ES for the multimodal barrier problem.

5. EXPERIMENTAL RESULTS ON IMAGE ANALYSIS PROBLEM

MI-ES are now used to find optimal parameter settings for the IVUS lumen detection algorithm. We focused on the lumen detector of the IVUS system, because it can produce good results in isolation without additional information about sidebranches, shadows, plagues and vessels. The settings used for the MI-ES and ES algorithms were $(\mu = 4, \lambda = 28)$ in combination with a single step-size per parameter type as found with the artificial test problem in the previous section.

[3] sum of absolute values

name	type	range	dependencies	default
maxgray	integer	[2, 150]	> mingray	35
mingray	integer	[1, 149]	< maxgray	1
connectivity	nominal	{4,6,8}		6
relativeopenings	boolean	{false,true}		true
nrofcloses	integer	[0, 100]	used if not relativeopenings	5
nrofopenings	integer	[0, 100]	used if not relativeopenings	45
scanlinedir	nominal	{0,1,2}		1
scanindexleft	integer	[-100, 100]	< scanindexright	-55
scanindexright	integer	[-100, 100]	> scanindexleft	7
centermethod	nominal	{0,1}		1
fitmodel	nominal	{ellipse, circel}		ellipse
sigma	continuous	[0.5 10.0]		0.8
scantype	nominal	{0,1,2}		0
sidestep	integer	[0, 20]		3
sidecost	continuous	[0.0, 100]		5
nroflines	integer	[32, 256]		128

Table 1: Parameters for the lumen feature detector.

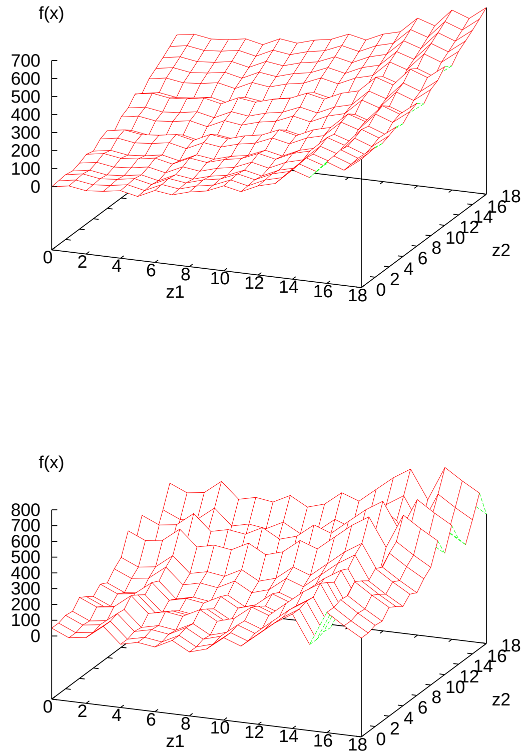

Figure 4: Surface plots of the barrier test function for two variables and $C = 20$ (upper) and $C = 100$ (lower). All other variables were kept constant at a value of zero, two integer values were varied in the range from 0 to 20.

Table 1 contains the parameters for the lumen feature detector together with their type, range, dependencies and the default settings determined by an expert. As can be seen the parameters are a mix of continuous, ordinal discrete (integer) and nominal discrete(including boolean) variables.

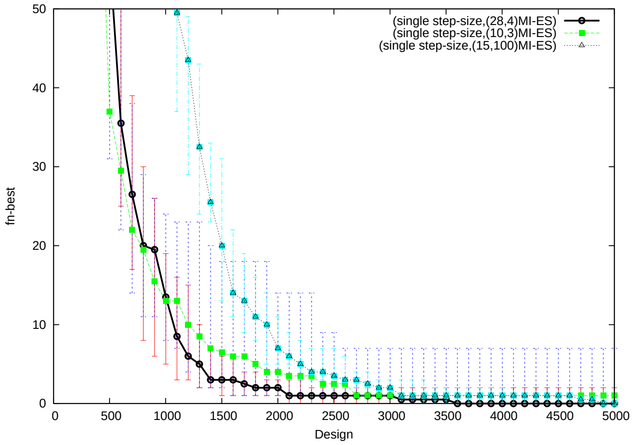

Figure 5: Median and quartiles for the best found function value of the barrier function ($C = 20$, 20 repeated runs) for different settings of μ and λ.

For the experiments we use five disjoint sets of 40 images which had been used previously in[1]. The images were acquired with a 20 Mhz Endosonics Five64 catheter (see Figure 9) using motorized pullback (1 mm/s). Image size is 384 × 384 pixels (8 bit greyscale) with a pixel-size of 0.02602 mm^2.

The fitness function used in the experiments is based on the difference between the contour c found by the lumen feature detector and the desired lumen contour C drawn by a human expert. The difference measure is defined as the sum of the distances of the points of contour c that are more than a *threshold* distance away from contour C. The reason to allow for a small difference between the two contours is that even an expert will not draw the exact same contour twice in a single IVUS image. The fitness function itself is the calculated average difference over the 40 images in the dataset.

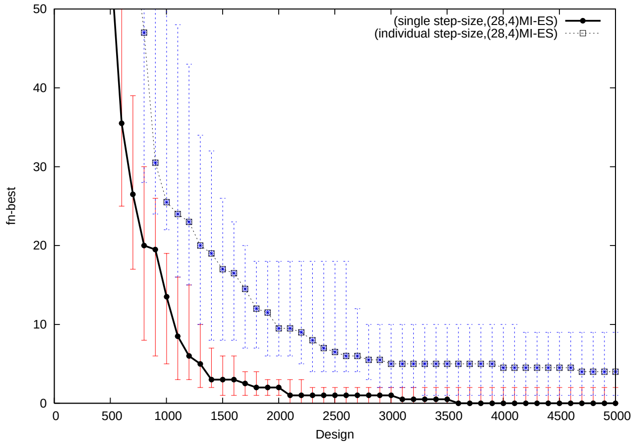

Figure 6: Median and quartiles for the best found function value: Single and multiple step-size MI-ES on the barrier function ($C = 20$, 20 repeated runs).

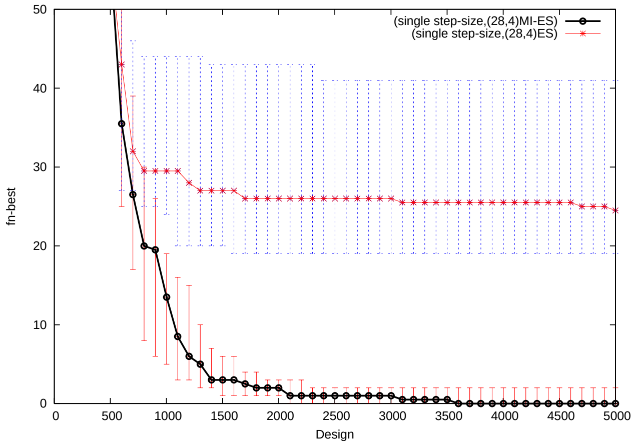

Figure 7: Median and quartiles for 20 repeated runs with single step-size MI-ES and standard ES on the barrier function ($C = 20$, 20 repeated runs).

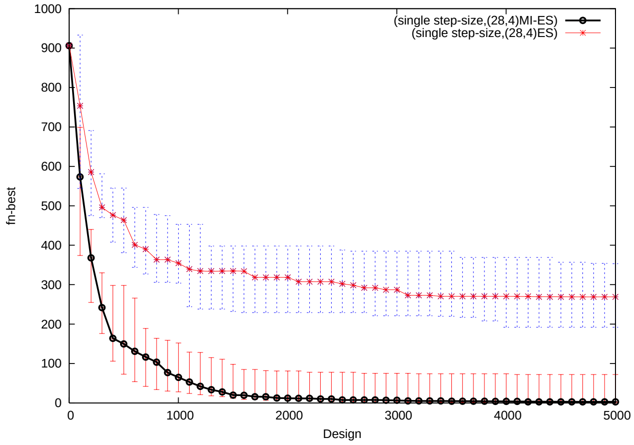

Figure 8: Median and quartiles for 20 repeated runs with single step-size MI-ES and standard ES on the barrier function ($C = 100$, 20 repeated runs).

Figure 9: A 20 Mhz Endosonics Five64 catheter such as was used to produce the IVUS images used in the experiments. Courtesy of Jomed/Endosonics.

Let $\#points$ denote the total number of points of contour C, then the contour difference is defined as:

$$difference(c, C) = \sum_{p=1}^{\#points} d(c_p, C), \;\; if \;\; d(c_p, C) > threshold$$

On each of the 5 datasets we trained our $(4, 28)$ MI-ES and ES algorithms. We limited our number of iterations to 25 resulting in 704 fitness evaluations because fitness evaluations are very time consuming. The results are displayed in Table 2. (MI-)ES solution 1 was trained on dataset 1 by the (MI-)ES algorithm, (MI-)ES solution 2 was trained on dataset 2 by the (MI-)ES algorithm, etc ....

Table 2 shows that for most cases the MI-ES parameter solutions result in lower average contour differences when applied to both test- and training data than the default parameters. Only parameter solution 3 applied to dataset 5 has a higher average contour difference (444.2 vs 446.4). To determine if the best results obtained by the MI-ES algorithm are also significantly better than the default parameter results, a paired two-tailed t-test was performed on the (40) difference measurements for each image dataset and each solution using a 95% confidence interval ($\text{p} = 0.05$). The t-test shows that all differences are significant except for the difference between MI-ES solution 3 applied to dataset 5 and the default parameters and the difference between MI-ES solution 5 applied to dataset 3 and the default parameters. Therefore we conclude that the MI-ES solutions are significantly better than the default parameter solution in 92% of the cases (23 out of 25) and equal in the other two cases.

When we look at the results of the ES parameter solutions compared to the default parameter solution we see that all the differences are statistically significant meaning that the ES solutions are significantly better than the default parameter solution in 23 out of 25 cases but worse in the other 2 cases (ES solutions 3 and 4 applied to dataset 5).

If we look at the performance of the MI-ES and ES algorithms when trained on a dataset we see that on Dataset 1 the ES solution is a little better, but the difference is not statistically significant. On all other datasets the MI-ES

	Dataset 1		Dataset 2		Dataset 3		Dataset 4		Dataset 5	
	Fitness	S.D.	Fitness	S.D.	Fitness	S.D.	Fitness	S.D.	Fitness	S.D.
Default Parameters	395.2	86.2	400.2	109.2	344.8	66.4	483.1	110.6	444.2	90.6
MI-ES Solution 1	151.3	39.2	183.6	59.0	201.0	67.1	280.9	91.9	365.5	105.9
MI-ES Solution 2	160.3	45.9	181.4	58.7	206.7	70.3	273.6	74.5	372.5	99.2
MI-ES Solution 3	173.8	42.1	202.9	69.1	165.6	47.2	250.7	80.2	446.4	372.8
MI-ES Solution 4	154.0	51.7	243.7	67.7	198.8	80.1	186.4	59.0	171.3	57.8
MI-ES Solution 5	275.7	75.6	358.4	76.9	327.7	56.7	329.1	82.0	171.8	54.5
ES Solution 1	149.8	46.0	185.1	60.1	203.1	66.6	270.2	69.8	358.1	100.1
ES Solution 2	157.8	45.2	183.0	57.3	202.8	68.1	270.7	69.4	364.2	98.5
ES Solution 3	212.6	46.3	243.3	72.3	182.3	49.7	252.9	58.4	756.8	734.4
ES Solution 4	247.2	143.0	255.7	85.4	176.4	49.2	232.9	62.4	941.0	795.2
ES Solution 5	217.0	51.6	225.3	68.5	292.8	73.6	330.7	78.0	324.7	96.0

Table 2: Performance of the best found MI-ES and ES parameter solutions when trained on one of the five datasets ((MI-)ES solution i was trained on dataset i). All parameter solutions and the (default) expert parameters are applied to all datasets. Average difference (fitness) and standard deviation w.r.t. expert drawn contours are given.

solution trained on that dataset is significantly better than the ES solution trained on the same dataset. On Dataset 5 MI-ES solution 4 has a slightly lower fitness than MI-ES solution 5 that was trained on the dataset but the difference is not statistically significant. On Dataset 3, ES solution 4 has a lower fitness than ES solution 3 but again the difference is not significant.

Visual inspection of the results of the application of MI-ES parameter solution 4 to the other datasets shows that this solution is a good *approximator* of the lumen contours in the other datasets, but that the particular solutions trained with those datasets follow the expert contours more closely. Perhaps dataset 4 contains features of the other datasets (1,2,3 and 5) which may explain this behavior. However, visual inspection of the the image datasets does not show any apparent differences. When we compare the contours detected by the MI-ES and ES algorithms to the expert drawn contours we see that they are very similar and in some cases the (MI-)ES contours actually seem to follow the lumen boundary more precisely. Besides being closer to the expert drawn contours, another major difference between the (MI-)ES found contours and the default parameter settings is that the ES solutions are more smooth.

Apart from looking at the average contour difference (or fitness) of the different parameter solutions we can also compare the performance between the MI-ES and ES algorithms by looking at their ability to "learn" the dependencies between the variables as displayed in Table 1. In Figure 10 the total number of illegal solutions evolved by both the MI-ES and ES algorithms are displayed. As can be seen the MI-ES algorithm manages to "learn" the dependencies much faster than the ES algorithm.

In Figures 11 and 12 we have plotted the fitness and best fitness for both the MI-ES and ES algorithms on Dataset 2. Invalid solutions were given a very high fitness penalty and are omitted from the plots to improve readability. In the case of the MI-ES algorithm the spread of the entire population decreases as the population reaches the best solution, which indicates that the step-size adaptation works properly.

6. CONCLUSIONS

In this paper we studied Mixed-Integer Evolution Strate-

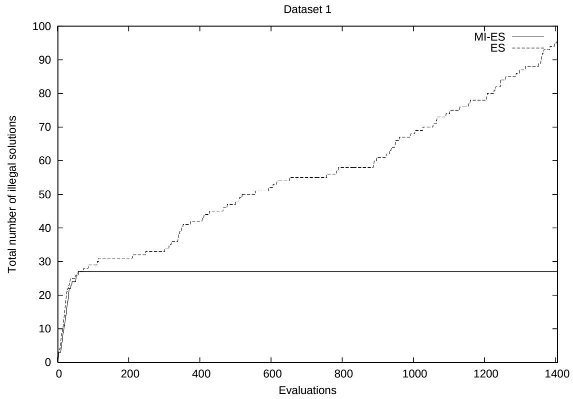

Figure 10: The accumulated number of illegal solutions on Dataset 2 evolved by both the MI-ES and ES (dotted line) algorithms. As can be seen the MI-ES algorithm manages to "learn" the dependencies quite fast while the ES algorithm keeps evolving invalid solutions even after 1400 fitness evaluations.

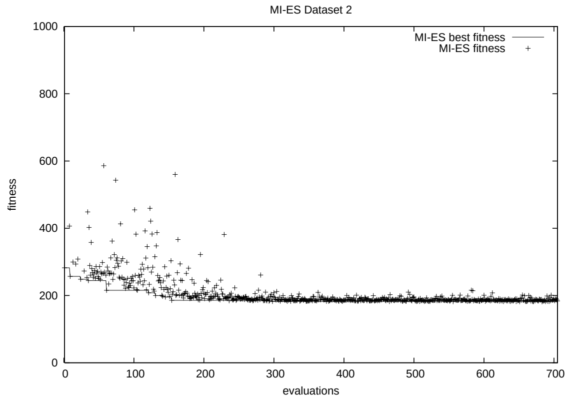

Figure 11: The fitness and best fitness values during the run of the MI-ES algorithm. As can be seen the entire population of the MI-ES algorithm quickly converges on the best found fitness.

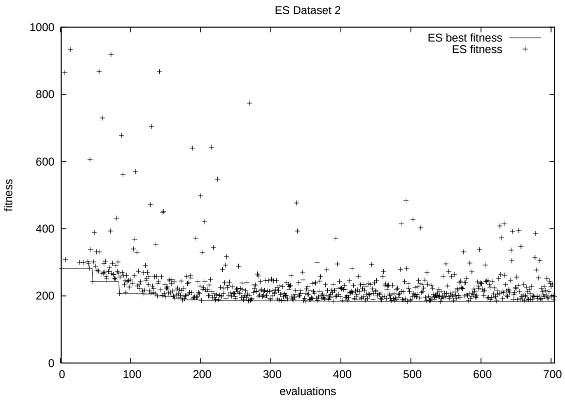

Figure 12: The fitness and best values fitness during the run of the ES algorithm. As can be seen the population does not converge on the best fitnes.

gies for the optimization of medical-image analysis problems. The Mixed-Integer Evolution Strategy uses specific variation operators for different types of decision variables (continuous, integer, and nominal discrete). All operators support automatic adaptation of the mutation strength and avoid biased sampling. Different instantiations of the MI-ES are tested on artificial test problems to determine favorable default settings. Moreover, on these problems we compare the MI-ES to the standard (continuous) ES using simple truncation of continuous variables. It turns out that the MI-ES approach has a higher convergence reliability than the standard ES.

A similar result is obtained for the medical image analysis. Here the MI-ES always produced better or equal results than the default settings chosen by an expert. The standard ES failed to do so in some of the cases. Moreover, on all five data sets the results of the MI-ES were significantly better (four times) or equal (one time) than those obtained with the standard ES, trained on the same data set.

In summary, the results show that the MI-ES is a valuable technique for improving the parameter settings of the lumen detector. The results encourage further studies on extended image sets and for other feature detectors. The results of this study suggest also its use in other mixed-integer optimization problems.

However, we do not expect to find one optimal solution for each feature detector to work in all possible contexts and for all possible patients. Therefore we are going to apply the methods outlined in this paper to different image interpretation contexts which should result in a set of optimal image feature detector solutions rather than in a single solution. The aim is then to let an agent in the multi-agent system decide which particular solution to use based on its current knowledge of the situation.

We also intend to further study the mixed-integer evolution strategy algorithm both in theory and practice to get more insight into its strengths and weaknesses.

7. ACKNOWLEDGMENTS

This research is supported by the Netherlands Organisation for Scientific Research (NWO) and the Technology Foundation STW.

8. ADDITIONAL AUTHORS

Thomas Bäck
Natural Computing Group, Leiden Institute of Advanced Computer Science (LIACS), Leiden University, The Netherlands, baeck@liacs.nl.
Jouke Dijkstra
Division of Image Processing (LKEB), Department of Radiology C2S, Leiden University Medical Center (LUMC), The Netherlands, J.Dijkstra@lumc.nl.
Johan H.C. Reiber
Division of Image Processing (LKEB), Department of Radiology C2S, Leiden University Medical Center (LUMC), The Netherlands, J.H.C.Reiber@lumc.nl.

9. REFERENCES

[1] E.G.P. Bovenkamp, J. Dijkstra, J.G. Bosch, and J.H.C. Reiber. Multi-agent segmentation of IVUS images. *Pattern Recognition*, 37(4):647–663, April 2004.

[2] Thomas Bäck and Martin Schütz. Evolution Strategies for Mixed-Integer Optimization of Optical Multilayer Systems. Evolutionary Programming, 33 - 51, 1995

[3] L. Ballerini and L. Franzén. Genetic Optimization of Morphological Filters with Applications in Breast Cancer Detection, In Raidl et al. eds: Applications of Evolutionary Computing, pp. 250–259, LNCS 3005, 2004, Springer.

[4] S. Cagnoni, A.B. Dobrzenieki, R. Poli and J.C. Yanch: Genetic algorithm-based interactive segmentation of 3D medical images. *Image and Vision Computing*,17(12):881–895, October 1999.

[5] K. Chen, I.C. Parmee and C.R. Gane. Dual mutation strategies for mixed-integer optimisation in power-station design: Evolutionary Computation, 1997., IEEE International Conference (1997), IEEE Press

[6] M. Emmerich, M. Schütz, B. Gross and M. Grötzner. Mixed-Integer Evolution Strategy for Chemical Plant Optimization. In I. C. Parmee, editor, Evolutionary Design and Manufacture (ACDM 2000), 2000, pp. 55-67, Springer NY

[7] F. Hoffmeister and J. Sprave. Problem independent handling of constraints by use of metric penalty functions. In L. J. Fogel, P. J. Angeline, and Th. Bäck, eds.: Evolutionary Programming V - Proc. Fifth Annual Conf. Evolutionary Programming (EP'96), pp. 289–294. The MIT Press, 1996.

[8] G. Koning, J. Dijkstra, C. von Birgelen, J.C. Tuinenburg, J. Brunette, J-C. Tardif, P.W. Oemrawsingh, C. Sieling, and S. Melsa. Advanced contour detection for three-dimensional intracoronary ultrasound: validation – in vitro and in vivo. *The International Journal of Cardiovascular Imaging*,(18):235–248, 2002.

[9] A. Newell. *Unified Theories of Cognition.* Number ISBN 0-674-92101-1. Harvard University Press, Cambridge, Massachusetts, 1990.

[10] G. Rudolph, An Evolutionary Algorithm for Integer Programming, In Davidor et al.: Proc. PPSN III, Jerusalem, Springer, Berlin, 1994, pp. 139–148.

[11] H.-P. Schwefel: Evolution and Optimum Seeking, Sixth Generation Computing Series, John Wiley, NY, 1995