Twin adaptive scheme for solving inverse problems

R. Schaefer[1] and B. Barabasz[2] and M. Paszyński[1]

[1] Department of Computer Science,University of Science and Technology, Cracow, Poland, email: schaefer@agh.edu.pl, paszynsk@agh.edu.pl
[2] Department of Modeling and Information Technology, University of Science and Technology, Cracow, Poland, e-mail: barabasz@metal.agh.edu.pl

Abstract. The paper deals with a class of inverse parametric problems for which the energy function may be defined. The advanced multi-deme strategy that offers an adaptive accuracy is utilized for solving associated optimal control problems. The direct problems necessary for fitness evaluation are computed by the *hp*-adaptive Finite Elements. The new iterative strategy balances the error of solving the direct problem and the error of solving the optimal control problem in order to decrease the total computational cost. The concept of the strategy is partially formally verified (see Lemma 3.1 and 3.2), moreover the advantages in the computational practice are mentioned.

1 Motivation

The inverse parameter problems are frequently formulated as the optimal control ones in which we try to find the vector of parameters of the partial differential equation by minimizing the proper error function. The high computational cost of the error function evaluation and its frequent multimodality make the inverse parameter problems extremely heavy ones.

The heuristic method well suited do solve such problems is the multi-deme Hierarchic Genetic Strategy (HGS) [1], [8]. It offers the reasonable efficiency by using the flexible accuracy of the search performed by various demes.

Each evaluation of the error function that stands for the fitness in the HGS needs the approximate solving of the direct problem (the boundary-value problem for the PDEs). One of the most effective method in this area is the *hp*-adaptive Finite Element Method (FEM) that may offer up to the exponential decrease of the direct problem error with respect to the *hp* approximation parameters (see e.g. [3]).

Authors pretended in [4] the advantages of the common adaptation of the step length of the convex optimization method with the *hp* adaptation in solving direct problem. The paper extends this idea of common adaptation in HGS and FEM in order to decrease the computational cost of solving inverse problems in case of multimodal error function. The class of inverse parametric problems for which the energy function may be defined is selected. The extended HGS/*hp*-adaptive FEM strategy (Algorithm 2) base on two mathematical results (Lemma 2, [4] and Lemma 2, Section 5 and the resulting equation 11) that makes possible to balance the errors coming from the optimization method with the relative error of solving direct problem.

2 Adaptive FEM technique for solving direct problems

The hp adaptive FEM [9], [5] generates a sequence of meshes delivering the exponential convergence of the numerical error with respect to the problem size, expressed in terms of the number of degrees of freedom over the computational mesh. Since the computational cost is the polynomial function of the problem size, then the hp adaptivity provides an exponential convergence of the numerical error with respect to the CPU time. The

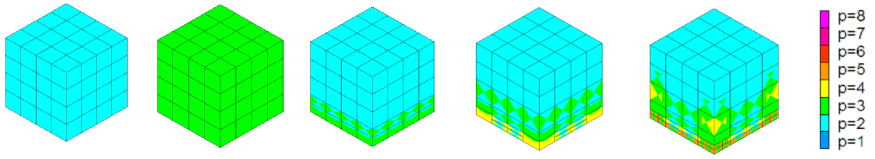

Figure 1. The coarse mesh with $p = 2$ and fine mesh with $p = 3$ on all elements edges, faces, and interiors. The optimal meshes after the first, second and third iterations. Different shading denote different polynomial orders of approximation.

method starts from the arbitrarily selected initial mesh as in the example presented in the most left part of Fig. 1. The polynomial order of approximation in this example was uniformly set to $p = 2$ over all finite elements edges, faces and interiors. The different polynomial orders of approximations are denoted on the pictures by different shading. Note, that the polynomial orders of approximation may vary on elements edges and faces, in particular we may have different orders of approximations in both directions on element faces. The polynomial orders of approximation on element interiors are not denoted in the pictures, but they may be different in each of three possible directions. The minimum rule is enforced on the mesh. The polynomial orders of approximation on element faces is equal to the minimum of orders of adjacent interiors, as well as the polynomial orders on edges are equal to the minimum of orders on adjacent faces.

The method solves the problem on the initial mesh, called the "coarse" mesh, then the mesh is globally hp refined. Each finite element is broken into 8 child elements, and the polynomial orders of approximation are uniformly raised by one, on each element edges, faces and interiors. The problem is solved again on the obtained "fine" mesh, presented in the second picture in Fig. 1. The relative error estimations for each finite element from the "coarse" mesh are utilized to generate new optimal mesh, presented in the third picture in Fig. 1. Finite element with high relative are either h, p or hp refined. The h refinement consists in breaking a finite element into smaller child elements, in one, two or three possible directions. The p refinement consists in adjusting polynomial orders of approximation on some element edges, faces or interiors. The hp refinement is a mix of both h and p refinements. For more details on the algorithm selecting optimal h, p or hp refinements, please refer to [9]. The optimal mesh becomes the coarse mesh for the next iterations, and the entire procedure is repeated. The example of generated sequence of optimal meshes is presented in Fig. 1. The fourth and fifth pictures present the optimal meshes generated in the second and third steps of the algorithm. Fot the the particular boundary value which was solved, the numerical error is reduced from 25% on the first mesh (presented in the first picture), through 15% on the second mesh (presented in the

third picture), through 8% on the third mesh (presented in the fourth picture), down to 5% on the last mesh.

3 The relation between the objective function error and the Finite Element Method error

Let us consider the direct problem defined by the abstract variational equation

$$\begin{cases} u \in u_0 + V \\ b(u,v) = l(v) \ \forall v \in V \end{cases} \tag{1}$$

obtained from the partial differential equation describing the sample physical phenomena. In the above V is the proper Sobolev space, the u_0 is the shift of the Dirichlet boundary conditions [3]. Functionals b and l depend on the physical phenomena to be modeled. They also depend on the inverse problem parameters d. If b is symmetric and positively defined [7], the variational problem 1 is equivalent with the minimization one 2

$$\begin{cases} u \in u_0 + V \\ E(u) = \frac{1}{2} b(u,u) - l(u) \longrightarrow \min \end{cases} \tag{2}$$

where $E(u) = \frac{1}{2} b(u,u) - l(u)$ is the functional of the total energy of the solution.

The problem 1 may be approximated by using FEM with the finite dimensional subspace $V_{h,p} \subset V$

$$\begin{cases} u_{h,p} \in u_0 + V_{h,p} \\ b(u_{h,p}, v_{h,p}) = l(v_{h,p}) \ \forall v_{h,p} \in V_{h,p}. \end{cases} \tag{3}$$

The coarse mesh solution $u_{h,p}$ is found in the $V_{h,p}$ space. The corresponding fine mesh solution $u_{\frac{h}{2},p+1}$ is found in the $V_{\frac{h}{2},p+1}$ space. The coarse mesh solution space is a subset of the corresponding fine mesh solution space $V_{h,p} \subset V_{\frac{h}{2},p+1}$, and both spaces are included in the subsets of the space V of the solutions of the exact variational problem 1.

The inverse problem can be formulated as

$$\text{Find } \hat{\mathbf{d}} \in \mathcal{D}: \ \left| J_{h,p}\left(\hat{\mathbf{d}}\right) - J(\mathbf{d}^*) \right| = \lim_{h \to +\infty, p \to +\infty} \min_{\mathbf{d} \in \mathcal{D}} \left| J_{h,p}(\mathbf{d}) - J(\mathbf{d}^*) \right| \tag{4}$$

where $\mathbf{d}^*$ denotes exact parameters of the inverse problem (exact solution of the variational formulation for these parameters is well comparable with experiment data), $\mathbf{d}$ denotes approximated parameters of the inverse problem, $\mathcal{D}$ is a set of all admissible parameters $\mathbf{d}$, $J(\mathbf{d}^*) = E(u(\mathbf{d}^*))$ is the energy of the exact solution $u(\mathbf{d}^*)$ of the variational problem 1 for exact parameters $\mathbf{d}^*$, $J_{h,p}(\mathbf{d}) = E(u_{h,p}(\mathbf{d}))$ is the energy of the solution $u_{h,p}(\mathbf{d})$ of the approximated problem 3 for approximated parameters $\mathbf{d}$.

The relative FEM error is defined by means of the energy norm difference between the coarse and fine mesh solutions

$$err_{FEM}(\mathbf{d}) = \left\| u_{h,p}(\mathbf{d}) - u_{\frac{h}{2},p+1}(\mathbf{d}) \right\|_E. \tag{5}$$

The energy norm involves the L^2 norm of the function and its first derivatives.

Objective function error is defined as the energy difference between the solution of the approximated problem 3 for approximated parameter $\mathbf{d}$ and the exact solution of the problem 1 for exact parameter $\mathbf{d}^*$, assumed to be equal to the energy measured during the experiment

$$e_{h,p}\left(\mathbf{d}\right) = \left|J_{h,p}\left(\mathbf{d}\right) - J\left(\mathbf{d}^*\right)\right|. \tag{6}$$

In other words, the approximated parameter $\mathbf{d}$ is placed into the approximated formulation 3, the solution of the problem $u_{h,p}\left(\mathbf{d}\right)$ (which depends on $\mathbf{d}$) is computed by FEM, and the energy of the solution $E\left(u_{h,p}\left(\mathbf{d}\right)\right)$ is computed.

Lemma 3.1. *(see Lemma 2 in [4])*

$$e_{\frac{h}{2},p+1}\left(\mathbf{d}\right) \leq \frac{1}{2}\left\|u_{\frac{h}{2},p+1}\left(\mathbf{d}\right) - u_{h,p}\left(\mathbf{d}\right)\right\|_E^2 + \left|J_{h,p}\left(\mathbf{d}\right) - J\left(\mathbf{d}^*\right)\right|,$$

where

$$e_{\frac{h}{2},p+1}\left(\mathbf{d}\right) = \left|J_{\frac{h}{2},p+1}\left(\mathbf{d}\right) - J\left(\mathbf{d}^*\right)\right|.$$

The objective function error over the fine mesh is limited by the relative FEM error of the coarse mesh with respect to the fine mesh, plus the objective function error over the coarse mesh.

We may evaluate

$$\left|J_{h,p}(\mathbf{d}) - J(\mathbf{d}^*)\right| \;\leq\; \left|J_{h,p}(\mathbf{d}) - J(\mathbf{d})\right| + \left|J(\mathbf{d}) - J(\mathbf{d}^*)\right| \tag{7}$$

Moreover, if we asume that the functional J is Lipschitz continuous with respect to the parameters $\mathbf{d}$, then $\left|J(\mathbf{d}) - J(\mathbf{d}^*)\right| \leq \alpha\left|\mathbf{d} - \mathbf{d}^*\right|$. The next Lemma is necessary for estimation of the first term in the above formula.

Lemma 3.2.

$$2\left|J_{h,p}(\mathbf{d}) - J(\mathbf{d})\right| = \left\|u(\mathbf{d}) - u_{h,p}(\mathbf{d})\right\|_E^2$$

Proof:

$$2\;\left|J_{h,p}(\mathbf{d}) - J(\mathbf{d})\right| = 2\;\left|Eu_{h,p}(\mathbf{d}) - Eu(\mathbf{d})\right| =$$
$$\left|b(u_{h,p}(\mathbf{d}), u_{h,p}(\mathbf{d})) - 2\,l(u_{h,p}(\mathbf{d})) - b(u(\mathbf{d}), u(\mathbf{d})) + 2\,l(u(\mathbf{d}))\right| =$$
$$\left|b(u_{h,p}(\mathbf{d}), u_{h,p}(\mathbf{d})) - b(u(\mathbf{d}), u(\mathbf{d})) + 2\,l(u(\mathbf{d}) - u_{h,p}(\mathbf{d}))\right| =$$
$$\left|b(u_{h,p}(\mathbf{d}), u_{h,p}(\mathbf{d})) - b(u(\mathbf{d}), u(\mathbf{d})) + 2\,b(u(\mathbf{d}), u(\mathbf{d}) - u_{h,p}(\mathbf{d}))\right| =$$
$$\left|b(u_{h,p}(\mathbf{d}), u_{h,p}(\mathbf{d})) - b(u(\mathbf{d}), u(\mathbf{d})) + 2\,b(u(\mathbf{d}), u(\mathbf{d})) - 2\,b(u(\mathbf{d}), u_{h,p}(\mathbf{d}))\right| =$$
$$\left|b(u_{h,p}(\mathbf{d}), u_{h,p}(\mathbf{d})) + b(u(\mathbf{d}), u(\mathbf{d})) - 2\,b(u(\mathbf{d}), u_{h,p}(\mathbf{d}))\right| =$$
$$\left|b(u_{h,p}(\mathbf{d}), u_{h,p}(\mathbf{d})) - b(u(\mathbf{d}), u_{h,p}(\mathbf{d})) + b(u(\mathbf{d}), u(\mathbf{d})) - b(u(\mathbf{d}), u_{h,p}(\mathbf{d}))\right| =$$
$$\left|-b(u(\mathbf{d}) - u_{h,p}(\mathbf{d}), u_{h,p}(\mathbf{d})) + b(u(\mathbf{d}) - u_{h,p}(\mathbf{d}), u(\mathbf{d}))\right| =$$
$$\left|b(u(\mathbf{d}) - u_{h,p}(\mathbf{d}), u(\mathbf{d}) - u_{h,p}(\mathbf{d}))\right| = \left\|u(\mathbf{d}) - u_{h,p}(\mathbf{d})\right\|_E^2$$

Finally, we have the desired dependency between the inverse error and the FEM error

$$e_{\frac{h}{2},p+1} \leq \left\|u_{\frac{h}{2},p+1}(\mathbf{d}) - u_{h,p}(\mathbf{d})\right\|_E^2 + \left\|u(\mathbf{d}) - u_{h,p}(\mathbf{d})\right\|_E^2 + \alpha\left|\mathbf{d} - \mathbf{d}^*\right|. \tag{8}$$

The last term $\alpha |\mathbf{d} - \mathbf{d}^*|$ is of the same degree as the Lipschitz constant α times the accuracy of the HGS algorithm, which will be shown in the following sections. The objective function error with respect to the fine mesh is limited by the relative FEM error of the coarse mesh solution with respect to the fine mesh solution plus absolute FEM error of the coarse mesh solutions plus the accuracy of the HGS algorithm.

4 The main idea of HGS

Hierarchic Genetic Strategy was introduced by Kołodziej and Schaefer (see [1], [8]). Its main idea is running a set of dependent evolutionary processes in parallel. The dependency relation has a tree structure with a restricted number of levels m. The processes of lower order (close to the root of the structure) represent a chaotic search with low accuracy. They detect the promising regions of the optimization landscape, in which more accurate processes of higher order are activated. Populations evolving in different processes can contain individuals which represent the solution (the phenotype) with different precision achieved by binary genotypes of different length.

The strategy starts with the process of the lowest order 1 called the root. After a fixed number of evolution epochs the best adapted individual is selected. We call this procedure a *metaepoch* of the fixed period K. After every metaepoch a new process of the order 2 can be activated. This procedure is called *sprouting operation*. Sprouting can be generalized in some way to branches of a population's tree of a higher order up to $m-1$. Sprouting is performed conditionally, according to the outcome of the *branch comparison operation*.

The HGS genetic process is of the order $j \in \{1, \ldots, m\}$ if the individuals from the evolving population have genotypes of the length $s_j \in \mathbb{N}$. The lengths of binary strings used in various order processes satisfy the inequality $1 < s_1 <, \ldots, < s_m < +\infty$. The initial population for the new sprouted branch of the order $j+1 \leq m$ contains individuals with prefixes identical to the genotype of the best adapted individual in the process of the order j. Suffixes of the length $s_{j+1} - s_j$ of these individuals are initialized randomly (according to the uniform distribution).

The branch comparison operation in HGS is based on the *prefix comparison*. The operator acts on populations evolving in the processes of two consecutive orders j and $j+1$. Let us assume that we distinguish the best fitted individual x from the branch of the j^{th} order after some metaepochs. If there is at least one individual with the prefix of the length s_j identical to x among $j+1$ order branches, then a new process of the order $j+1$ is not activated.

The special kind of *hierarchical nested encoding* is used in order to obtain the search coherency for branches of various degrees. Let us denote by Ω_s the genetic universum composed of binary codes of the length $s > 0$, so $\Omega_{s_1}, \ldots, \Omega_{s_m}$ stand for the binary genetic universa of branches of degrees $1, \ldots, m$. Each universum is linearly ordered by the relation induced by the natural order among integers represented by binary strings. Moreover, for $j = 2, \ldots, m$ we can represent genetic spaces Ω_{s_j} in the following way:

$$\Omega_{s_j} = \left\{ (\omega, \xi),\, \omega \in \Omega_{s_{j-1}},\, \xi \in \Omega_{s_j - s_{j-1}} \right\}. \tag{9}$$

We describe the hierarchical nested encoding for $\mathcal{D} \subset \mathbb{R}$ only. The natural generalization of this construction to $\mathcal{D} \subset \mathbb{R}^N$, $N > 1$ may be found in [8].

We intend to define a sequence of meshes $\mathcal{D}_{r_1}, \ldots, \mathcal{D}_{r_m} \subset \mathcal{D} \subset \mathbb{R}$ so that $\#\Omega_{s_j} = \#\mathcal{D}_{r_j}$, $j = 1, \ldots, m$ and a sequence of one-to-one encoding mappings $\mathrm{code}_j : \Omega_{s_j} \to \mathcal{D}_{r_j}$. First, we arbitrarily define the densest mesh $\mathcal{D}_{r_m}$ in $\mathcal{D} \subset \mathbb{R}$ and the encoding $\mathrm{code}_m : \Omega_{s_m} \to \mathcal{D}_{r_m}$ as a strictly increasing function. Next, we arbitrarily define the set of selections $\phi_j : \Omega_{s_j} \to \Omega_{s_{j+1}-s_j}$, $j = 1, \ldots, m-1$ necessary for the construction of meshes $\mathcal{D}_{r_j}$, $j = 1, \ldots, m-1$. Finally, we put

$$\mathcal{D}_{r_j} = \left\{ \mathrm{code}_{j+1}(\omega, \phi_j(\omega)),\ \omega \in \Omega_{s_j} \right\}. \tag{10}$$

The meshes and encoding of the lower order are defined recursively. Figure 2 below shows the sample meshes D_{r_1}, D_{r_2} and $D_{r_3} \subset \mathbb{R}$ in the case of $s_1 = 2$, $s_2 = 3$, $s_3 = 5$, $\phi_1(00) = \phi_1(01) = 1$, $\phi_1(10) = \phi_1(11) = 0$, $\phi_2 \equiv 01$. j = 1, . . . , m-1.

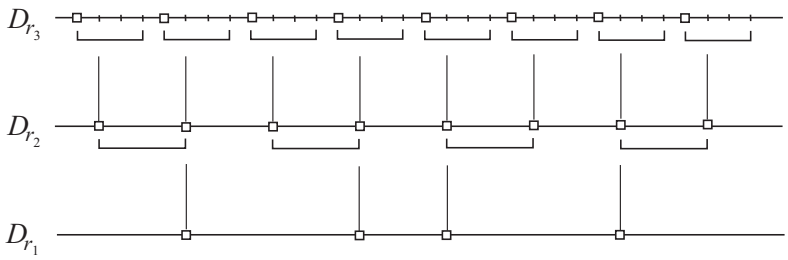

Figure 2. One dimensional nested meshes ($\mathcal{D} \subset \mathbb{R}$) for Hierarchical Genetic Strategy in the case $s_1 = 2, s_2 = 3, s_3 = 5$.

The Simple Genetic Algorithm defined (see Vose [6]) with the constant size μ_j populations that depends only on the branch order $j = 1, \ldots, m$ is used for each branch processing during metaepoch. Usually, larger population cardinality is set for lower order branches (close to the root) and much smaller cardinality is set for higher order branches and leafs. Additionally, the mutation rate p_m is set higher for the root and main branches in order to strengthen the wide exploration of the admissible domain.

Two kinds of stop condition are applied in the HGS strategy. The first called *branch stop condition* detects the lack of progress in the evolution process. It bases on the information gathered by the branch process only and is utilized to stop and remove the non-promising branches of the strategy. The second one, called *global stop condition* try to encounter the situation in which the HGS could not find more local extremes. The global information about the HGS searching progress is necessary in this case.

Let us assume for the sake of simplicity, that all HGS branches are processed in the common shared memory, so that the information about the searching progress is globally available. We assume moreover that the operations on the branch populations during each metaepoch are performed exclusively by the branch processes. The separate module continuously checks the global stop condition and sends the proper signal to the branch processes if satisfied. Now we are ready to formulate the draft of the HGS branch algorithm of the j^{th} order $j = 1, \ldots, m$ (see the Algorithm 1).

1: **if** $j = 1$ **then**
2: initialize the root population P^0;
3: **end if**

4: **repeat**
5: $t \leftarrow 0$;
6: **if** *global_stop_condition* received **then**
7: STOP;
8: **end if**
9: **for all** $i \in P^t$ **do**
10: compute $f_j(i)$;
11: **end for**
12: **if** *branch_stop_condition* **then**
13: STOP;
14: **end if**
15: perform selection with the fitness f_j;
16: perform genetic operations;
17: **if** NOT ($t \mod K$) **then**
18: distinguish the best fitted individual x^* from P^t;
19: **if** (NOT $prefix\_comparision(x^*)$) AND ($j < m$) **then**
20: *sproute*;
21: **end if**
22: **end if**
23: $t \leftarrow t + 1$;
24: **until** ($false$)

Algorithm 1: Draft of the HGS branch algorithm of j^{th} order

5 The coupled direct-inverse adaptive algorithm

The HGS may be utilized to solve the parameter inverse problem 4 under consideration. The searching domain $\mathcal{D} \subset \mathbb{R}^N$ is the N-dimensional brick of admissible parameters where N stands for the number of independent parameters to be identified. The m-level nested affine encoding is utilized. Populations of higher order represent the solution with the higher accuracy, while the lower order ones deliver the worse approximation of solutions. Note, that the distance δ_j between the nodes of the j^{th} order mesh $\mathcal{D}_{r_j}$ represent the upper bound of the accuracy that is available for the search at this HGS degree. The individual $i \in \Omega_{s_j}$ represents the admissible solution $\mathbf{d} = \text{code}_j(i)$ to the inverse problem in the j^{th} order HGS process. The right-hand-side of the formula 8 informs us about the components, that affects the FEM error degree if the fitness in the j^{th} order process is computed.

$$\left\| u_{\frac{h}{2},p+1}(\mathbf{d}) - u_{h,p}(\mathbf{d}) \right\|_E^2 + \| u(\mathbf{d}) - u_{h,p}(\mathbf{d}) \|_E^2 + \alpha \delta_j \tag{11}$$

This information allows us to propose the extended version of the HGS strategy dedicated for solving inverse problems, in which the fitness value is computed with the accuracy adopted to the maximum accuracy offered by the HGS process of a particular order j. The idea of this strategy is to balance the components of the FEM error given by the formula 11 in order to decrease the computational cost of fitness evaluation, keeping the accuracy of inverse problem solving on the assumed level. This approach is stressed in the Algorithm 2. In particular we perform the *hp* adaptation of the FEM solution of the direct problem while the ratio $\frac{err_{FEM}}{\delta_j}$ is greater then the assumed *Ratio* which stands

for the parameter of this strategy. The above procedure is executed independently for each individual $i \in P^t$ at the evaluation step (see lines 10 - 17 in the Algorithm 2).

1: **if** $j = 1$ **then**
2: initialize the root population P^0;
3: **end if**
4: **repeat**
5: $t \leftarrow 0$;
6: **if** *global_stop_condition* received **then**
7: STOP;
8: **end if**
9: **for all** $i \in P^t$ **do**
10: solve the direct problem for $\mathbf{d} = \text{code}_j(i)$ on the coarse and fine FEM meshes;
11: compute the relative FEM error $err_{FEM}(\mathbf{d})$ according to the formula 5;
12: **while** $err_{FEM}(\mathbf{d}) > Ratio\ \delta_j$ **do**
13: execute one step of *hp* adaptivity;
14: solve the problem on the new coarse and fine FEM meshes;
15: compute the relative FEM error $err_{FEM}(\mathbf{d})$ according to the formula 5;
16: **end while**;
17: compute $f_j(i) = |J_{h,p}(\mathbf{d}) - J(\mathbf{d}^*)|$ using the FEM mesh finally established;
18: **end for**
19: **if** *branch_stop_condition* **then**
20: STOP;
21: **end if**
22: perform selection with the fitness f_j;
23: perform genetic operations;
24: **if** NOT ($t \mod K$) **then**
25: distinguish the best fitted individual x^* from P^t;
26: **if** (NOT $prefix\_comparison(x^*)$) AND ($j < m$) **then**
27: *sprout*;
28: **end if**
29: **end if**
30: $t \leftarrow t + 1$;
31: **until** ($false$)

Algorithm 2: Draft of the HGS branch algorithm of j^{th} order with varying accuracy of the fitness computation

6 Concluding remarks

- The proposed strategy allows for optimal balancing of the direct and inverse problem solution accuracy. Maintaining the optimal balance may significantly reduce the computational cost of the inverse problem solution, keeping the optimal total accuracy. The paper [4] showed the first test results in this direction. It contains the optimal balance of errors: FEM and Hook-Jeeves procedures for searching the CTE parameter in the SFIL process. The proposed strategy lead to significant speedup of the solution process.

- Kołodziej and Schaefer proved, that under some assumptions concerning the heuristics (see Vose [6] for the SGA heuristic definition) of the SGA processes that define the HGS strategy, it has the same ability of finding local extreme as the SGA process used in the highest order branches (see e.g. Theorem 4.1 in [8]). They also try to evaluate (see e.g. Hypothesis 1 in [8]) the efficiency of the HGS strategy with respect to the single SGA process searching on the densest mesh $\mathcal{D}_{r_m}$. It seems that the similar results may be obtained for the coupled HGS/*hp*-adaptive FEM defined in the section. The main condition that have to be checked is the proper tunning of the SGA processes of the consecutive orders $1, \ldots, m$ in case of the fitness computed with the varying accuracy.
- The presented version of the HGS/*hp*-adaptive FEM (see Algorithm 2) needs the common RAM (shared memory computer environment) for implementation. It is easy to re-designed this strategy to the message-passing environments. The agent-oriented version of HGS is already designed and tested [2].
- The Algorithm 2 base on the assumed maximum accuracy forcing by the highest order mesh $\mathcal{D}_{r_m}$. It is possible to relax this condition by using the conditional sprouting operation with respect to the ratio of the inverse error in the total computational error shown in 8.

Acknowledgments. The financial assistance of KBN, project no. 10.10.110.642. The work reported in this paper was partially supported by Polish MNiSW grant no. 3 TO 8B 055 29. The work of the third author was supported by Foundation for Polish Science under program Homing.

Bibliography

[1] Kołodziej J. *Hierarchical Strategies of the Genetic Global Optimization.* PhD thesis, Jagiellonian University, Faculty of Mathematics and Informatics, 2003.

[2] Momot J, Kosacki K, Grochowski M, Uhruski P, and Schaefer R. Multi-agent system for irregular parallel genetic computations. *LNCS*, 3038:623–630, 2004.

[3] Demkowicz L. *Computing with hp-Adaptive Finite Elements.* Chapman & Hall/Crc Applied Mathematics & Nonlinear Science, 2006.

[4] Paszyński M, Barabasz B, and Schaefer R. Efficient adaptive strategy for solving inverse problems. *Lecture Notes in Computer Science*, 4487 (2007):342–349, 2007.

[5] Paszyński M and Demkowicz L. Parallel fully automatic hp-adaptive 3d finite element package. *Engineering with Computers*, 22:255–276, 2006.

[6] Vose MD. *The Simple Genetic Algorithm.* MIT Press, 1999.

[7] Ciarlet P. *The Finite Element Method for Elliptic Problems.* Society for Industrial & Applied, 2002.

[8] Schaefer R and Kołodziej J. Genetic search reinforced by the population hierarchy. *Foundations of Genetic Algorithms 7*, pages 383–399, 2003.

[9] Rachowicz W, Pardo D, and Demkowicz L. Fully automatic hp-adaptivity in three dimensions. Technical report, ICES Report 04-22, 2004.