Accelerating Non-Linear Time-Harmonic Problems by a Hybrid Picard-Newton Approach

H. Vande Sande, T. Boonen, H. De Gersem, F. Henrotte, K. Hameyer

***Abstract*— Non-linear magnetic problems are commonly solved by iterative methods. This paper discusses the application of the Picard-method and the Newton-method in the context of solving time-harmonic problems. It is shown that the overall computation time can be decreased by initiating the non-linear solution process using Picard-iterations and by switching to Newton-iterations as soon as an estimator indicates that this is appropriate. The estimator relies on the equivalence between the Newton-method and the gradient based methods for minimizing non-linear multivariate functions. The developed hybrid Picard-Newton method is applied for the simulation of the short-circuit operation of a 400 kW four-pole induction motor. It is pointed out how the estimator can be used for solving multi-harmonic problems as well.**

***Index Terms*— Harmonic analysis, Minimization methods, Newton-Raphson method, Non-linear magnetics**

I. Introduction

NON-LINEAR quasi-static magnetic problems are governed by the equation

$$\nabla \times (\nu \nabla \times A) + \sigma \frac{\partial A}{\partial t} = J \ , \tag{1}$$

with ν the reluctivity tensor [Am/Vs], A the magnetic vector potential [Vs/m], σ the electric conductivity [A/Vm] and J the applied current density vector [A/m^2]. Equation (1) must be complemented with an appropriate gauge and appropriate boundary conditions in order to determine a unique solution [1]. For isotropic and non-hysteretic materials, the reluctivity is a scalar non-linear function of the flux density B [Vs/m^2].

In a *time-domain* finite element (FE) analysis, the non-linearity of magnetic materials can be considered by applying a time-stepping approach. However, this may result in excessive computational efforts if only the steady state behaviour is of interest. In that case, it is more efficient to perform a frequency-domain analysis by applying a *time-harmonic* (TH) finite element method, in which the magnetic vector potential $A(\mathbf{x}, t)$ and the current density vector $J(\mathbf{x}, t)$ are represented by

$$A(\mathbf{x}, t) = \Re\{\tilde{A}(\mathbf{x}) e^{j\omega t}\} \tag{2}$$

and

$$J(\mathbf{x}, t) = \Re\{\tilde{J}(\mathbf{x}) e^{j\omega t}\} \tag{3}$$

H. Vande Sande, F. Henrotte, K. Hameyer are with the Katholieke Universiteit Leuven, Dept. ESAT, Div. ELECTA, Kasteelpark Arenberg 10, B-3001 Heverlee-Leuven, Belgium. E-mail: hans.vandesande@esat.kuleuven.ac.be

T. Boonen is with the Katholieke Universiteit Leuven, Dept. Computer Science, Celestijnenlaan 200A, B-3001 Heverlee-Leuven, Belgium.

H. De Gersem is with the Technische Universität Darmstadt, Computational Electromagnetics Laboratory, Schloßgartenstraße 8, D-64289 Darmstadt, Germany.

respectively. The governing equation (1) becomes

$$\nabla \times \left(\nu \nabla \times \tilde{A}\right) + j\sigma\omega\tilde{A} = \tilde{J} \ , \tag{4}$$

where the variables $\tilde{A}$ and $\tilde{J}$ are phasors [2], [3].

This paper presents a combined Picard-Newton scheme for iteratively solving the non-linear problem (4) by the FE method. The solution process is started by performing Picard-iterations. As soon as an estimator indicates that the expected convergence rate of the Newton-strategy is close to quadratic, Newton-iterations take over the solution process. It will be shown that this combination yields a shorter overall computation time. The method is illustrated for the two-dimensional simulation of the short-circuit operation of a 400 kW induction motor. It is pointed out how this approach can be straightforwardly generalized to the harmonic balance finite element method (HBFEM) [4], [5], [6].

II. Time-Harmonic FE Formulation

The discretized magnetic vector potential writes

$$\tilde{A}(\mathbf{x}) = \sum_{i=1}^{n} \tilde{A}_i \varphi_i(\mathbf{x}) \ , \tag{5}$$

with n the number of phasor-valued connectors $\tilde{A}_i$ (values at the nodes in 2D problems and circulations along the edges in a 3D problem) and φ_i the corresponding shape functions. Applying Galerkin's method to (4), one ends up with the following system of algebraic equations:

$$\tilde{\mathbf{r}}(\tilde{\mathbf{A}}) = \left(\mathbf{K}(\tilde{\mathbf{A}}) + j\mathbf{L}\right)\tilde{\mathbf{A}} - \tilde{\mathbf{T}} = \mathbf{0} \ , \tag{6}$$

with $\tilde{\mathbf{A}}$ the column vector of the phasor-valued connectors $\tilde{A}_i$. The $(n \times n)$ matrices $\mathbf{K}$ and $\mathbf{L}$ are symmetric with real-valued entries. The column vector $\tilde{\mathbf{T}}$ may have phasor-valued entries. The non-linearity of the problem is due to the dependency of $\mathbf{K}$ on $\tilde{\mathbf{A}}$. In (6), one has additionally defined the residual vector $\tilde{\mathbf{r}}$, which is a non-linear function of $\tilde{\mathbf{A}}$.

III. Residual Approach

In order to solve (6), an iterative strategy is usually applied. The non-linear problem is numerically solved when the norm of the residual has been made smaller than a fixed tolerance. Below, the Picard and Newton-method are discussed.

Given $\tilde{\mathbf{A}}_0$
Evaluate $\mathbf{K}_0$
For $k = 0, 1, 2, ...$

- Solve $(\mathbf{K}_k + j\mathbf{L})\,\tilde{\mathbf{A}}_{k+1} = \tilde{\mathbf{T}}$
- Set $\tilde{\mathbf{d}}_k \longleftarrow \tilde{\mathbf{A}}_{k+1} - \tilde{\mathbf{A}}_k$
- Find $\alpha_k \in [0, 1]$ s.t. $\|\tilde{\mathbf{r}}(\tilde{\mathbf{A}}_k + \alpha_k\tilde{\mathbf{d}}_k)\| < \|\tilde{\mathbf{r}}(\tilde{\mathbf{A}}_k)\|$
- Set $\tilde{\mathbf{s}}_k \longleftarrow \alpha_k\tilde{\mathbf{d}}_k$
- Set $\tilde{\mathbf{A}}_{k+1} \longleftarrow \tilde{\mathbf{A}}_k + \tilde{\mathbf{s}}_k$

Fig. 1. Picard iteration scheme.

Given $\mathbf{A}_0$
Evaluate $\mathbf{r}_0$
For $k = 0, 1, 2, ...$

- Evaluate $\mathbf{J}_k$
- Solve $\mathbf{J}_k\mathbf{d}_k = -\mathbf{r}_k$
- Find $\alpha_k \in [0, 1]$ s.t. $\|\mathbf{r}(\mathbf{A}_k + \alpha_k\mathbf{d}_k)\| < \|\mathbf{r}(\mathbf{A}_k)\|$
- Set $\mathbf{s}_k \longleftarrow \alpha_k\mathbf{d}_k$
- Set $\mathbf{A}_{k+1} \longleftarrow \mathbf{A}_k + \mathbf{s}_k$

Fig. 2. Newton iteration scheme.

A. Picard-iterations

The Picard-method (Fig. 1) is obtained by successively evaluating the stiffness matrix $\mathbf{K}_k = \mathbf{K}(\tilde{\mathbf{A}}_k)$ and solving the complex system of equations $(\mathbf{K}_k + j\mathbf{L})\tilde{\mathbf{A}}_{k+1} = \tilde{\mathbf{T}}$. For this reason, it is often referred to as the method of *successive substitution*. However, as it is not guaranteed that $\|\tilde{\mathbf{r}}(\tilde{\mathbf{A}}_{k+1})\| < \|\tilde{\mathbf{r}}(\tilde{\mathbf{A}}_k)\|$, it is possible that the algorithm diverges. To avoid this, an efficient line search along the direction $\tilde{\mathbf{d}}_k = \tilde{\mathbf{A}}_{k+1} - \tilde{\mathbf{A}}_k$ is performed, in order to determine a relaxation factor $\alpha_k \in [0, 1]$ for which $\|\tilde{\mathbf{r}}(\tilde{\mathbf{A}}_k + \alpha_k\tilde{\mathbf{d}}_k)\| < \|\tilde{\mathbf{r}}(\tilde{\mathbf{A}}_k)\|$ is true. The iterate $\tilde{\mathbf{A}}_{k+1}$ is then set to $\tilde{\mathbf{A}}_k + \alpha_k\tilde{\mathbf{d}}_k$. In the algorithm of Fig. 1, the evaluation of the stiffness matrix is performed during the line search routine [7], [8], [9].

B. Newton-iterations

In order to determine a search direction, the Picard-method does not use any information about the differential reluctivity of the non-linear materials. Obviously, this restricts the achievable convergence rate of this iterative method. To improve on that, the Jacobian $\mathbf{J}$ of $\tilde{\mathbf{r}}$ is involved in the iteration process.

The basic idea behind the Newton-method is to set the first order Taylor series expansion of the residual $\tilde{\mathbf{r}}$ to zero. However, when working with complex variables, the Taylor series expansion is only defined if the residual is an analytic function of $\tilde{\mathbf{A}}$. Unfortunately, in magnetodynamic problems, this is generally not the case because the reluctivity tensor depends on the modulus of the flux density $\tilde{B}$, which is not an analytical function of $\tilde{\mathbf{A}}$. Mathematically, this is expressed by the fact that the Cauchy-Riemann condition

$$\frac{\partial\tilde{\mathbf{r}}}{\partial\tilde{\mathbf{A}}_k^{\mathrm{re}}} = \frac{1}{j}\frac{\partial\tilde{\mathbf{r}}}{\partial\tilde{\mathbf{A}}_k^{\mathrm{im}}} \tag{7}$$

is not fulfilled [10]. Consequently, in order to use a Newton-scheme, one has to derive the Jacobian from the equivalent real representation of $\tilde{\mathbf{r}}$, defined by

$$\mathbf{r} = \begin{pmatrix} \tilde{\mathbf{r}}^{\mathrm{re}} \\ \tilde{\mathbf{r}}^{\mathrm{im}} \end{pmatrix} . \tag{8}$$

By setting

$$\mathbf{M} = \begin{pmatrix} \mathbf{K} & -\mathbf{L} \\ \mathbf{L} & \mathbf{K} \end{pmatrix} , \tag{9}$$

$$\mathbf{A} = \begin{pmatrix} \tilde{\mathbf{A}}^{\mathrm{re}} \\ \tilde{\mathbf{A}}^{\mathrm{im}} \end{pmatrix} \text{ and } \mathbf{T} = \begin{pmatrix} \tilde{\mathbf{T}}^{\mathrm{re}} \\ \tilde{\mathbf{T}}^{\mathrm{im}} \end{pmatrix} , \tag{10}$$

it follows that

$$\mathbf{r}(\mathbf{A}) = \mathbf{M}(\mathbf{A})\,\mathbf{A} - \mathbf{T} . \tag{11}$$

The $(2n \times 2n)$ matrix $\mathbf{M}$ has real-valued entries but it is non-symmetric. The $(2n \times 1)$ vectors $\mathbf{r}$, $\mathbf{A}$ and $\mathbf{T}$ have real-valued entries.

The Newton iteration scheme is directly obtained by setting the first-order Taylor expansion of $\mathbf{r}$

$$\mathbf{r}(\mathbf{A} + \mathbf{d}) \approx \mathbf{r}(\mathbf{A}) + \mathbf{J}(\mathbf{A})\,\mathbf{d} \tag{12}$$

to zero, with

$$\mathbf{J}_{ij} = \frac{\partial\mathbf{r}_i}{\partial\mathbf{A}_j} = \mathbf{M}_{ij} + \sum_{k=1}^{2n}\frac{\partial\mathbf{M}_{ik}}{\partial\mathbf{A}_j}\mathbf{A}_k . \tag{13}$$

This reveals that the $(2n \times 2n)$ Jacobian $\mathbf{J}$ is the sum of the non-symmetric matrix $\mathbf{M}$ and a symmetric matrix $\mathbf{N}$ [10]:

$$\mathbf{J}(\mathbf{A}) = \mathbf{M}(\mathbf{A}) + \mathbf{N}(\mathbf{A}) . \tag{14}$$

The Jacobian $\mathbf{J}$ is non-symmetric but it has real-valued entries. Given the iterate $\mathbf{A}_k$, one can evaluate $\mathbf{r}_k$ and $\mathbf{J}_k$ and solve the system $\mathbf{J}_k\mathbf{d}_k = -\mathbf{r}_k$. In order to avoid the divergence of the iterative process, $\mathbf{A}_{k+1}$ is updated by $\mathbf{A}_k + \alpha_k\mathbf{d}_k$, with α_k chosen in such a way that the norm of the residual strictly decreases at each step.

The Newton-scheme is summarized in Fig. 2. As in the Picard-scheme, the evaluation of the residual is performed in the line search routine. Although the variables in both methods are of a different type (real versus complex-valued), one can observe similarity between the methods: the Newton-scheme reduces to the Picard-scheme of Fig. 1 if the non-linear term $\mathbf{N}$ in (14) is omitted. The Picard-method can therefore be regarded as a quasi-Newton method, using an approximate jacobian that is much faster to compute and that yields a system which is easier to solve. This fact is exploited to increase the overall speed of the solution process.

C. Computational aspects

The system that has to be solved in the Picard-strategy is complex symmetric and of size $(n \times n)$. For this purpose, either the Conjugate Orthogonal Conjugate Gradient method (COCG) [11] or a variant of the Quasi-Minimal Residual method (QMR) for complex symmetric matrices [12] can be used. For the

Newton-strategy, a real positive definite, but non-symmetric system of size $(2n \times 2n)$ has to be solved. As a consequence, system solvers such as the Bi-Conjugate Gradient method (BiCG) or the Generalized Minimal Residual method (GMRES) can be used [13]. Unfortunately, the computational cost for applying BiCG on the real equivalent system is approximately twice the one for applying COCG on the complex symmetric system. This favours the Picard-approach.

On the other hand, the Picard-strategy makes no use of any information about the differential reluctivities in the elements. For this reason, it features a lower asymptotic convergence rate of the non-linear residual when compared to the Newton-strategy. However, it is observed in practice that the initial convergence rate of both strategies is more or less the same. Moreover, if both strategies start from the zero solution, the first iterate is identical. Therefore, it is suggested to initiate the solution process by Picard-iterations and to switch to Newton-iterations as soon as a significantly better convergence rate can be expected. This approach is applied here and is called the *hybrid Picard-Newton approach*. Below, it is outlined in which way the analogy between the presented strategies and the gradient based methods for minimizing multivariate functions is used for developing a suitable estimator for the switching moment.

IV. Function Minimization Approach

Efficient gradient based methods for minimizing multivariate functions $F(\mathbf{A})$ rely on the second order Taylor series expansion of that function. At the iterate $\mathbf{A}_k$, the second order Taylor series expansion is a quadratic surface $F_k^{\mathrm{qm}}(\mathbf{s})$ in a multidimensional parameter space:

$$F_k^{\mathrm{qm}}(\mathbf{s}) = F(\mathbf{A}_k) + \mathbf{s}^{\mathrm{T}}\nabla F(\mathbf{A}_k) + \frac{1}{2}\mathbf{s}^{\mathrm{T}}\nabla^2 F(\mathbf{A}_k)\mathbf{s} \; . \quad (15)$$

This surface is called the *quadratic model* here. A step towards the minimum of the function can be found by applying an efficient line search along the Newton-direction, i.e. the solution of the system

$$\nabla^2 F(\mathbf{A}_k)\mathbf{d}_k = -\nabla F(\mathbf{A}_k) \; . \quad (16)$$

The Newton-direction is a vector which points from the actual iterate to the minimum of the quadratic model [7], [9].

Within the context of function minimization, the Newton-scheme of Fig. 2 is obtained as well when minimizing half the square of the residual norm:

$$F = \frac{1}{2}\|\mathbf{r}\|^2 \; . \quad (17)$$

This can be explained by taking its gradient

$$\nabla F = \mathbf{J}^{\mathrm{T}}\mathbf{r} \quad (18)$$

and its Hessian

$$\nabla^2 F = \mathbf{J}^{\mathrm{T}}\mathbf{J} + \sum_{i=1}^{n} \mathbf{r}_i \nabla^2 \mathbf{r}_i \; . \quad (19)$$

$$\nabla^2 F \approx \mathbf{J}^{\mathrm{T}}\mathbf{J} \quad (\text{if } \|\mathbf{r}\| \text{ small}) \quad (20)$$

If (18) and (20) are introduced in (16), $\mathbf{d}_k$ equals the line search direction of the Newton-strategy. Hence, applying the Newton-method for solving the TH problem (4) is equivalent to minimizing the multivariate function (17) with an approximate Hessian. This equivalence forms the basis of a truncation error estimator for determining the switching moment between the Picard and the Newton-method. It also reveals some interesting features [9]:

- The approximate Hessian (20) of F is positive definite. As a consequence, the Picard and Newton-strategy both compute a direction, along which a line search algorithm can find a lower value of (17).
- From (19), it follows that the smaller the residual is, the better the Hessian approximation is. Moreover, the approximation equals the exact Hessian at the minimum of (17). Hence, if a minimum is found, it is the only minimum in its neighbourhood. It is the global minimum if the function is convex. This is not guaranteed for the general case which is treated here. However, simulations reveal that, in practice, the computed minimum is the global minimum.
- Due to the approximation of the Hessian which is applied to compute the line search direction, the Picard and Newton-strategies are equivalent to quasi-Newton minimization methods, featuring superlinear convergence. However, the asymptotic convergence rate of the Newton-strategy is close to quadratic because the Hessian is asymptotically exact. The latter statement is not true for the Picard-strategy.

V. Basic Truncation Error Estimator

A. Principle

During the non-linear iteration process, the difference between two successive approximate solutions gradually decreases. As a consequence, the quadratic model (15) steadily becomes a better estimator of the function $F(\mathbf{A})$, because the truncation error of the second order Taylor series expansion is of order $O(h^3)$. A measure of the quality of the quadratic approximation is found in the theory of trust region methods for minimizing multivariate functions [9]. The ratios

$$\rho_{k1} = \frac{F(\mathbf{A}_k) - F_k^{\mathrm{qm}}(\mathbf{s}_k)}{F(\mathbf{A}_k) - F(\mathbf{A}_{k+1})} \quad (21)$$

and

$$\rho_{k2} = \frac{F_{k+1}^{\mathrm{qm}}(-\mathbf{s}_k) - F(\mathbf{A}_{k+1})}{F(\mathbf{A}_k) - F(\mathbf{A}_{k+1})} \quad (22)$$

express how close the actual reduction of the function approximates the predicted reduction by the quadratic model. Therefore, ratio (21) uses the quadratic model at iterate k, while (22) uses the quadratic model at iterate $k+1$. Fig. 3 illustrates their meaning, for a univariate function. The closer these ratios are to unity, the more accurate the quadratic approximation is. Therefore, their relative difference κ from one is a measure for the validity of the quadratic model:

$$\kappa = 100 \times \frac{|1-\rho_{k1}| + |1-\rho_{k2}|}{2} \; . \quad (23)$$

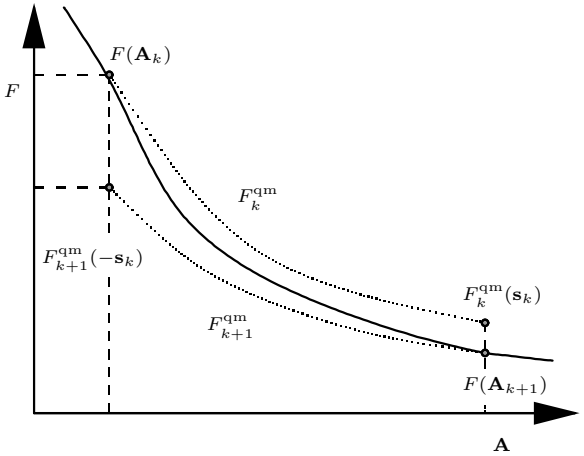

Fig. 3. The meaning of ratios (21) and (22), for a univariate function $F(\mathbf{A})$ (solid line). The quadratic models are represented by dotted lines.

This quantity is called the *truncation error estimator*. If κ approaches zero, it is expected that the quadratic model is an accurate approximation of the function in the neighbourhood of the actual iterate.

B. Computational aspects

The computation of ρ_{k1} and ρ_{k2} requires the knowledge of the Picard-step $\mathbf{s}_k$, the residual and the Jacobian, both in the current and previous iterate. The individual terms in these ratios are then computed by:

$$F(\mathbf{A}_k) = \frac{1}{2}\mathbf{r}_k^{\mathrm{T}}\mathbf{r}_k \ , \tag{24}$$

$$F(\mathbf{A}_{k+1}) = \frac{1}{2}\mathbf{r}_{k+1}^{\mathrm{T}}\mathbf{r}_{k+1} \ , \tag{25}$$

$$F_k^{\mathrm{qm}}(\mathbf{s}_k) = \frac{1}{2}\mathbf{r}_k^{\mathrm{T}}\mathbf{r}_k + \left(\mathbf{J}_k^{\mathrm{T}}\mathbf{r}_k\right)^{\mathrm{T}}\mathbf{s}_k + \frac{1}{2}\mathbf{s}_k^{\mathrm{T}}\mathbf{J}_k^{\mathrm{T}}\mathbf{J}_k\mathbf{s}_k \ , \tag{26}$$

$$F_{k+1}^{\mathrm{qm}}(-\mathbf{s}_k) = \frac{1}{2}\mathbf{r}_{k+1}^{\mathrm{T}}\mathbf{r}_{k+1} - \left(\mathbf{J}_{k+1}^{\mathrm{T}}\mathbf{r}_{k+1}\right)^{\mathrm{T}}\mathbf{s}_k + \frac{1}{2}\mathbf{s}_k^{\mathrm{T}}\mathbf{J}_{k+1}^{\mathrm{T}}\mathbf{J}_{k+1}\mathbf{s}_k \ . \tag{27}$$

However, when organized efficiently, the evaluation of (24) - (27) requires only

- the computation and storage of the non-linear contributions to the Jacobian,
- the storage of the previous residual vector,
- two matrix-vector multiplications,
- four dot-products.

In the example, it will be shown that the evaluation time of κ is negligible when compared to the time required for solving the linear system of equations.

VI. Improved Truncation Error Estimator

The line search algorithm yields a reduction of the step length, when compared to the computed direction $\mathbf{d}_k$. As a result, it is possible that the ratios (21) and (22) indicate a good agreement between the actual function and its approximation, even when this is not true for the full step. Therefore, it is suggested to alter the definition of the truncation error estimator κ in such a way that it estimates the quality of the quadratic model if the non-relaxed step would be applied, given only the data for the actual underrelaxed step.

Since the truncation error for the second order Taylor series expansion is of order $O(h^3)$, one can define two scalar constants c_k^+ and c_k^- from

$$F(\mathbf{A}_{k+1}) = F_k^{\mathrm{qm}}(\mathbf{s}_k) + c_k^+\|\mathbf{s}_k\|^3 \tag{28}$$

and

$$F(\mathbf{A}_k) = F_{k+1}^{\mathrm{qm}}(-\mathbf{s}_k) + c_k^-\|\mathbf{s}_k\|^3 \tag{29}$$

respectively. Elaborating ratio (21), using (15), (18), (20) and (28), yields

$$\frac{1}{\rho_{k1}} = 1 + \frac{c_k^+\|\mathbf{s}_k\|^3}{\mathbf{r}_k^{\mathrm{T}}\mathbf{J}_k\mathbf{s}_k + \frac{1}{2}\|\mathbf{J}_k\mathbf{s}_k\|^2} \ . \tag{30}$$

In a next step, it is assumed that c_k^+ is the same for the full step. The expected ratio $\rho_{k1}^{\mathbf{d}}$ for the full step can then be computed by transforming $\mathbf{s}_k$ into $\mathbf{d}_k$. From (30), it follows that

$$\frac{1}{\rho_{k1}^{\mathbf{d}}} = 1 + \frac{c_k^+\|\mathbf{d}_k\|^3}{\mathbf{r}_k^{\mathrm{T}}\mathbf{J}_k\mathbf{d}_k + \frac{1}{2}\|\mathbf{J}_k\mathbf{d}_k\|^2} \ . \tag{31}$$

In order to recompute ρ_{k1} for the actual step, $\mathbf{d}_k$ is now determined by $\mathbf{s}_k/\alpha_k$, and the nominator in the right hand side is divided by α_k^3. After reordering, this finally yields

$$\frac{1}{\rho_{k1}} = 1 + \frac{c_k^+\|\mathbf{s}_k\|^3}{\alpha_k\left[\alpha_k\mathbf{r}_k^{\mathrm{T}}\mathbf{J}_k\mathbf{s}_k + \frac{1}{2}\|\mathbf{J}_k\mathbf{s}_k\|^2\right]} \ . \tag{32}$$

Comparing (30) and (32) reveals the effect of the proposed modification on ρ_{k1}. An analogous argumentation can be followed for ρ_{k2}, yielding

$$\frac{1}{\rho_{k2}} = 1 + \frac{c_k^-\|\mathbf{s}_k\|^3}{\alpha_k\left[-\alpha_k\mathbf{r}_{k+1}^{\mathrm{T}}\mathbf{J}_{k+1}\mathbf{s}_k + \frac{1}{2}\|\mathbf{J}_{k+1}\mathbf{s}_k\|^2\right]} \ . \tag{33}$$

Using (32) and (33), the improved truncation error estimator κ is computed in the same way as in (23). There is no significant difference in the computational effort for evaluating this estimator, when compared to the basic truncation error estimator.

VII. Implementation

For the analysis, the mathematical software libraries PETSc (Portable Extensible Toolkit for Scientific Computing) and TAO (Toolkit for Advanced Optimization) have been used [14], [15]. These packages are freeware and written in C/C++. They provide a rich environment for developing scientific applications in a single or multiprocessor environment.

VIII. Example

A four-pole 400 kW induction motor is simulated under short-circuit operation by applying three different methods:

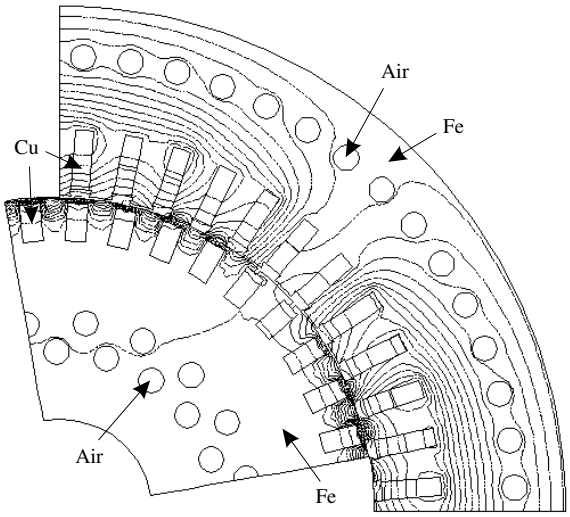

Fig. 4. Magnetic field of a 400 kW induction motor in short-circuit operation.

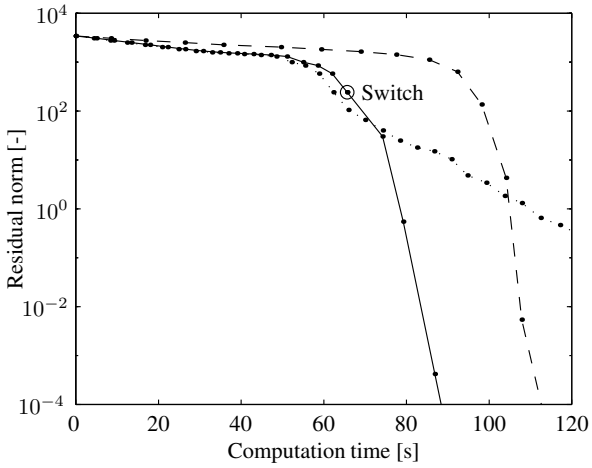

Fig. 5. The residual norm as a function of computation time, for the Newton-method (dashed), the Picard-method (dotted) and the hybrid Picard-Newton method (solid).

1) The Picard-method, using an ILU-preconditioned COCG-algorithm for solving the associated complex symmetric system of equations;
2) The Newton-method, using an ILU-preconditioned BiCG-algorithm for solving the associated real, positive definite, but non-symmetric system of equations;
3) The proposed hybrid Picard-Newton method, combining the previous methods and applying the improved truncation error estimator.

For all methods, a cubic line search method is applied for determining the relaxation factor at each non-linear iteration [9]. The triangular finite element mesh of the test problems contains 1419 nodes and 2772 elements. The magnetic vector potential is discretized by linear nodal elements.

Fig. 4 shows how the magnetic field is pushed out of the rotor by the induced rotor currents. For the three methods, the norm of the residual and the relaxation factor are plotted as a function of computation time in Figs. 5 and 6 respectively. This example illustrates why it is advantageous to initiate the non-linear iteration process by the Picard-method: the latter starts converging earlier with respect to the computation time than the Newton-method, although more iterations are required. Obviously, the Newton-method is more attractive when compared to the Picard-method if the desired accuracy is high. From Fig. 5, it follows that a significant improvement is obtained by combining both non-linear methods.

The value of the proposed truncation error estimator in (23) is plotted in Fig. 7, with and without the improvement given by (32) and (33). The computation of this estimator involves some extra floating point operations, which cause a slight speed reduction, as can be observed in Fig. 5. For larger problems, that difference would be negligible.

Initially, the estimator indicates a good agreement between the function and its quadratic model. This is caused by the strong underrelaxation in the first non-linear iterations, while starting from the zero solution, in which the model behaves linearly. Once saturation occurs at some places of the model, the

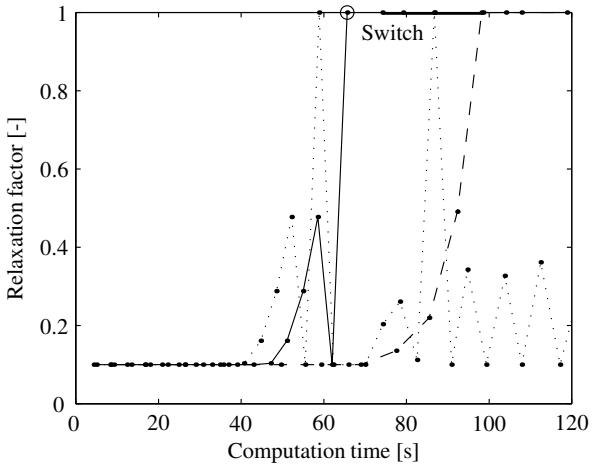

Fig. 6. The relaxation factor as a function of computation time, for the Newton-method (dashed), the Picard-method (dotted) and the hybrid Picard-Newton method (solid).

truncation error estimator quickly increases. Further on, the estimator steadily decreases due to the reduction of the step size. As this indicates that the quality of the quadratic model is increasing, it is decided to switch to the Newton iteration scheme when the estimator is smaller than a fixed value (25 % in this case). From Fig. 6 it is obvious that this limit is the most appropriate here, because no underrelaxation is required for the remaining Newton-iterations. This example shows that the overall computation time by adopting the hybrid Picard-Newton approach is decreased with approximately 25 %.

IX. Extension to the HBFEM

The harmonic balance finite element method allows the simulation of the steady state condition when multiple frequencies

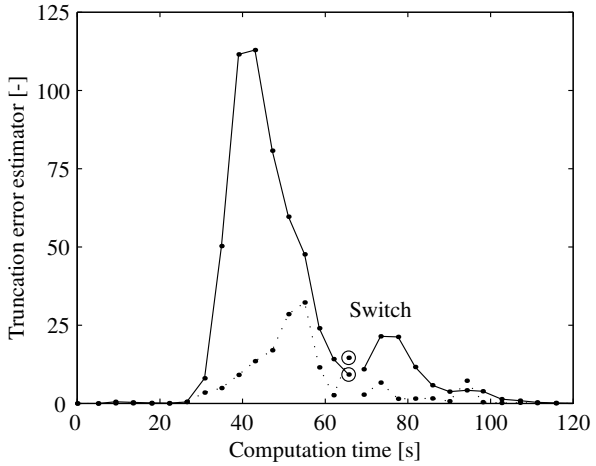

Fig. 7. The value of the basic (dotted) and improved (solid) truncation error estimator during the convergence process of the Picard-method.

are involved. Several applications of this method have been discussed in literature [4], [5], [6]. The approach found in [4] is directly extendible to be used in the proposed hybrid Picard-Newton method. If n_f frequencies are considered, one can define a residual

$$\tilde{\mathbf{r}}(\tilde{\mathbf{A}}) = \left(\mathbf{K}(\tilde{\mathbf{A}}) \star + \mathbf{L} \star \mathfrak{J}\right) \tilde{\mathbf{A}} - \tilde{\mathbf{T}} \,. \tag{34}$$

Here, $\mathbf{K}$ and $\mathbf{L}$ are symmetric $(n_f n \times n_f n)$ matrices with real-valued entries. The column vectors $\tilde{\mathbf{r}}$, $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{T}}$ are of size $(n_f n \times 1)$ and have complex-valued entries. For every connector, each component of the considered frequency spectrum is represented by a single entry in these vectors. The $\star$ denotes a convolution and $\mathfrak{J}$ is the operator representing the Fourier transform of the time derivative. The inherent non-linear system of equations in (34) is complex symmetric and the Picard-approach can be applied to solve it [4].

In analogy to the time-harmonic case, the Cauchy-Riemann condition (7) is not satisfied. Therefore, an improvement of the asymptotic convergence rate can only be obtained by introducing the Jacobian of the real-valued equivalent of $\tilde{\mathbf{r}}$. Splitting up (34) in its real and imaginary components yields the real equivalent formulation given in [4], [5], [6]. The non-linear contribution to the Jacobian can be obtained in the same way as presented in [10]. As a result, one can apply the Newton-method as well for solving the non-linear problem. Analogous to the TH case, the underlying linear system which has to be solved here is non-symmetric and of double size. Therefore, the same reasoning can be made as in the TH case. As a consequence, applying the proposed hybrid Picard-Newton approach to the HBFEM can lead to a significant improvement of the overall computation time too.

X. Conclusions

Non-linear time-harmonic problems are commonly solved by iterative methods, such as the Picard-method (successive substitution) or the Newton-method. The Picard-method is attractive because the time for solving the underlying linear systems of equations is the smallest. This is a consequence of their symmetric nature. On the other hand, the Newton-method features a higher asymptotic convergence rate. To make use of both properties, it is suggested to combine these methods. The Newton-method is initiated when the actual convergence rate of the latter is expected to be close to quadratic. The switching moment is indicated by a truncation error estimator. This estimator is based on the equivalence of the Newton-method with methods for minimizing multivariate functions. The resulting hybrid Picard-Newton method is applied for simulating the short-circuit operation of an induction motor, revealing an decrease of the overall computation time of approximately 25 %. It is pointed out how the same approach can be adopted for solving non-linear multi-harmonic problems.

Acknowledgments

The authors are grateful to the Belgian "Fonds voor Wetenschappelijk Onderzoek Vlaanderen" (project G.0427.98) and the Belgian Ministry of Scientific Research (IUAP No. P5/34).

References

[1] K.J. Binns, P.J. Lawrenson, and C.W. Trowbridge, *The Analytical and Numerical Solution of Electric and Magnetic Fields*, Wiley, Chichester, 1992.

[2] D. Lederer and A. Kost, "Modelling of nonlinear magnetic material using a complex effective reluctivity," *IEEE Transactions on Magnetics*, vol. 34, no. 5, pp. 3060–3063, Sept. 1998.

[3] G. Paoli, O. Bíró, and G. Buchgraber, "Complex representation in nonlinear time harmonic eddy current problems," *IEEE Transactions on Magnetics*, vol. 34, no. 5, pp. 2625–2628, Sept. 1998.

[4] H. De Gersem, H. Vande Sande, and K. Hameyer, "Strong coupled multiharmonic finite element simulation package," *COMPEL*, vol. 20, no. 2, pp. 535–546, 2001.

[5] J. Gyselinck, P. Dular, C. Geuzaine, and W. Legros, "Harmonic-balanced finite-element modeling of electromagnetic devices: A novel approach," *IEEE Transactions on Magnetics*, vol. 38, no. 2, pp. 521–524, Mar. 2002.

[6] J. Lu, S. Yamada, and K. Bessho, "Time-periodic magnetic field analysis with saturation and hysteresis characteristics by harmonic balance finite element method," *IEEE Transactions on Magnetics*, vol. 26, no. 2, pp. 995–998, Mar. 1990.

[7] J.J. Moré and D.J. Thuente, "Line search algorithms with guaranteed sufficient decrease," *ACM Transactions on Mathematical Software*, vol. 20, pp. 286–307, 1994.

[8] I. Munteanu, S. Drobny, T. Weiland, and D. Ioan, "Triangle search method for nonlinear electromagnetic field computation," *COMPEL*, vol. 20, no. 2, pp. 417–430, 2001.

[9] J. Nocedal and S.J. Wright, *Numerical Optimization*, Springer Series in Operations Research. Springer, New York, USA, 1st edition, 1999.

[10] D. Lederer, H. Igarashi, and A. Kost, "The Newton-Raphson method for complex equation systems," in *Proceedings of the 7th international IGTE symposium on numerical field calculation in electrical engineering (IGTE96)*, Graz, Austria, Sept. 1996, pp. 391–394.

[11] H.A. Van der Vorst and J.B.M. Melissen, "A Petrov-Galerkin type method for solving $Ax = b$, where A is symmetric complex," *IEEE Transactions on Magnetics*, vol. 26, no. 2, pp. 706–708, Mar. 1990.

[12] R.W. Freund, "Conjugate gradient-type methods for linear systems with complex symmetric coefficient matrices," *SIAM Journal on Scientific Computing*, vol. 13, pp. 425–448, Jan. 1992.

[13] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, 1996.

[14] S. Balay, W.D. Gropp, L. Curfman McInnes, and B.F. Smith, "PETSc users manual," Tech. Rep. ANL-95/11 - Revision 2.1.2, Mathematics and Computer Science Division, Argonne National Laboratory, 2002, See http://www.mcs.anl.gov/petsc.

[15] S. Benson, L. Curfmann McInnes, J. Moré, and J. Sarich, "TAO users manual," Tech. Rep. ANL/MCS-TM-242, Mathematics and Computer Science Division, Argonne National Laboratory, 2002, See http://www.mcs.anl.gov/tao.