A New Algorithm to Solve Large-Scale Fixed-Point Problems

Michel Bierlaire* Frank Crittin†

*Institute of Mathematics
Ecole Polytechnique Fédérale de Lausanne
CH-1015 Lausanne, Switzerland
michel.bierlaire@epfl.ch

†IDIAP
Rue du Simplon 4, Case Postale 592
CH-1920 Martigny, Switzerland
frank.crittin@idiap.ch

1 Introduction

Classical algorithms to solve fixed point problems are iterative methods based on the famous Banach Contraction Principle, which can be described as follows. Let $\mathcal{T} : X \to X$ be a mapping such that there exists $x^* \in X$ and $\mathcal{T}(x^*) = x^*$. Given $x_0 \in X$, the iterates of the method are given by

$$x_{k+1} = x_k + \alpha_k(\mathcal{T}(x_k) - x_k) \tag{1}$$

where $\alpha_k \in [0; 1]$. The *fixed point iterations*, also called *method of successive substitutions* and *nonlinear Richardson iterations*, is obtained with $\alpha_k = 1, \forall k$. It has been proved to be convergent if the mapping $\mathcal{T}$ is contracting.

The method of successive averages (MSA) uses $\alpha_k = \frac{1}{k}$. This method has been successively used for some classical transportation problems, as for example Sheffi & Powell (1982) who used it for stochastic user equilibrium or Cantarella (1997) who applied this algorithm to solve two general fixed point formulations of multimode multi-users equilibrium problems.

The Polyak averaging method is a simple off-line running average of points generated by (1). More precisely at each iteration we compute a new iterate, say $\Psi_k = \sum_i^k \frac{x_i}{k}$. Polyak & Juditsky (1992) have shown that the sequence Ψ_k converge to x^* at an optimal rate, if $\alpha_k \to 0$ slower than $o(1/k)$. Remarkably this procedure theoretically equals or surpasses asymptotic performances of any iterative methods defined by (1).

Another natural way to express fixed point problems is as resolution of systems of nonlinear equations $\mathcal{T}(x) - x = F(x) = 0$, with $F : X \to X$, $X \subseteq \mathbb{R}^n$. Most methods used to solve such systems of equations lie in the quasi-Newton framework. At each iteration k solve:

$$\begin{aligned} B_k s_k &= -F(x_k), && (2)\\ x_{k+1} &= x_k + s_k, && (3)\end{aligned}$$

and update B_{k+1}, where matrices $B_k \in \mathbb{R}^{n\times n}$ for all k. Specific instances are based on specific formula to compute B_{k+1}. In particular, if $B_{k+1} = \nabla F(x_{k+1})^T$, it is the Newton method. Although very efficient, this method suffer from the need for analytical derivatives. Secant quasi-Newton methods require B_{k+1} to verify the secant equation

$$B_{k+1}(x_{k+1} - x_k) = F(x_{k+1}) - F(x_k). \quad (4)$$

The most classical is Broyden's method, where

$$B_{k+1} = B_k + \frac{\left((F(x_k) - F(x_{k+1})) - B_k(x_k - x_{k+1})\right)(x_k - x_{k+1})^T}{(x_k - x_{k+1})^T(x_k - x_{k+1})}. \quad (5)$$

The main drawback of such methods in the context of large scale problems is the storage cost for the matrix B_{k+1} and also the resolution of the associated linear system (2).

Large-scale adaptations have been proposed in the literature. Broyden (1965) propose to update B_k^{-1} instead of B_k, avoiding to solve (2). Limited memory methods are based on a compact representation of matrices B_k^{-1} (Gomes-Ruggiero, Martinez & Moretti 1991), (Byrd, Nocedal & Schnabel 1994). The Inverse-Column Updating method (ICUM), by Martinez & Zambaldi (1992) is a secant algorithm where B_{k+1}^{-1} is obtain from B_k^{-1} by changing only one of its columns.

But the most successful methods for solving large-scale systems of nonlinear equations are probably Newton-Krylov methods (Kelley 2002). The use finite-difference approximation of the derivative within various small dimension Krylov subspaces. Their main drawback in our context is that they are not appropriate in the presence of stochasticity.

2 Large Scale Generalized Secant Method

We propose a method which is an extension of Broyden's idea. When Broyden's method uses the secant equation (4) to interpolate the linear model and the function at the two last iterates, we propose to use more than two of them. In this case, it is not appropriate anymore to impose exact interpolation. Instead, we propose a least-squares approach to obtain B_{k+1}^{-1}:

$$B_{k+1}^{-1} = \underset{J}{\operatorname{argmin}} \left\| J \begin{pmatrix} \Omega \cdot Y_{k+1} & \Gamma \cdot I_{n\times n} \end{pmatrix} - \begin{pmatrix} \Omega \cdot S_{k+1} & \Gamma \cdot (B_{k+1}^0)^{-1} \end{pmatrix} \right\|_F^2 \quad (6)$$

where $\Omega \in \mathbb{R}^{k+1}$ is a diagonal matrix with weights ω_{k+1}^i on the diagonal for $i = 0, \cdots, k$; the matrix Γ contains weights associated with the arbitrary term $(B_{k+1}^0)^{-1}$; $Y_{k+1} = (y_k, y_{k-1}, \ldots, y_0)$; $S_{k+1} = (s_k, s_{k-1}, \ldots, s_0)$ with $y_k = F(x_{k+1}) - F(x_k)$ and $s_k = x_{k+1} - x_k$.

Let $A = \begin{pmatrix} \Omega \cdot Y_{k+1} & \Gamma \cdot I_{n\times n} \end{pmatrix}$ and $C = \begin{pmatrix} \Omega \cdot S_{k+1} & \Gamma \cdot (B_{k+1}^0)^{-1} \end{pmatrix}$, using these notations, (6) can be written as $B_{k+1}^{-1} = \underset{J}{\operatorname{argmin}} \|A - C\|_F^2$. Solving the normal equations we can directly compute the associated quasi-Newton step given in (2):

$$s_k = -(CA^T)(AA^T)^{-1}F(x_k) \tag{7}$$

With a small amount of linear algebra[1] we can show that s_k defined by (7) is equivalent to the following:

$$\begin{cases} 1.\ \text{Solve} & x = \operatorname*{argmin}_y \|Ay - F(x_k)\|_2^2 \\ 2.\ \text{Compute} & s_k = -Cx \end{cases} \tag{8}$$

Remark that the least-squares associated with (8) is now a vector least-squares, contrarily to (6) which is a generalized matrix least-squares. Moreover with this formulation there is no need to store or even construct the matrix B_{k+1}^{-1}, and consequently the method can be implemented as a matrix-free algorithm, *i.e.* only matrix-vector products have to be computed, which is decisive for large scale problems. The only matrices that we need to store are Y_k, S_k, $(B_{k+1}^0)^{-1}$, Ω and Γ. More precisely, matrices Y_k and S_k have size $n \times (\kappa - 1)$ where n is the size of the problem and κ the number of iterates kept in the population. The matrix $(B_{k+1}^0)^{-1}$ is an a priori matrix whose role is to overcome the possible under-determination of the problem (6).

The algorithm based on the least squares approach is described in detail by Bierlaire & Crittin (2003a), where a proof of convergence can be found. The adaptations for large scale problems are described in Bierlaire & Crittin (2003b). A transportation application is described in Bierlaire & Crittin (2003c). In this extended abstract, we provide various numerical results illustrating the good behavior of the algorithm.

3 Numerical results

In this section we present numerical results using the proposed algorithm. We first consider global performances of this method on medium scale problems. Then, we show that iGSM is efficient on large scale systems on a set of classical difficult problems and illustrate the behavior of iGSM on a specific problems, the standard Convection-Diffusion problem, particularly to underline the influence of the size of the population on the performance of the algorithm. Finally we propose some results on noisy systems of nonlinear equations to underline the robustness of our population approach compared to classical large scale methods in presence of different types of stochasticity. In conclusion, we illustrate the applicability of iGSM to the consistent anticipatory route guidance generation, a real transportation fixed-point problem of very high dimension and stochasticity.

For all these experiences using iGSM and ICUM we have only used undamped version, without any step control or globalization approach. This is motivated by Ruggiero, D.N.Kozakevich & J.M.Martnez (1996), who conclude that globalizations based on backtracking strategy do not seem to be efficient in the context of quasi-Newton methods for nonlinear systems of equations. Our experience confirms this observation. The tests we have performed indeed confirm that globalization techniques may significantly decrease the performance of the algorithms, and may sometimes even fail to converge, even when the local version does converge.

All algorithms and test functions have been implemented with the package Octave (Eaton 1997) and computations have been done on a laptop equipped with 1066MHz CPU in double

[1] Remarking that $A^T(AA^T)^{-1} = (A^TA)^{-1}A^T$

precision. The machine epsilon is about 2.2204e-16. In all the tests we used a standard stopping criterion in spite of potential scaling problems (Dennis & Schnabel 1996):

$$\|F(x_k)\| \leq 10^{-6}\|F(x_0)\| \tag{9}$$

All the algorithms are stopped after 200 function evaluations, or when $\|F(x)\|_2 > 10^{10}$. In those cases, the algorithm is declared to have failed to converge.

3.1 General Performance analysis

Before analyzing large scale problems, we first expose a performance analysis of iGSM method to solve medium scale nonlinear systems of equations, extending the results presented by Bierlaire & Crittin (2003a). We compare it with the Hybrid method proposed by Martinez (1982), the ICUM method by Martinez & Zambaldi (1992) and the GSM method. The hybrid method is based on conjectures allowing to choose, at each iteration, between the Broyden Good or the Broyden Bad update. Martinez (2000) observes a systematic improvement of the Hybrid approach with respect to each individual approach BGM and BBM. iGSM method has been implemented with the large-scale features, that is $B^0_{k+1} = I_{n\times n}$, $\kappa = 10$, $\Gamma = \tau I_{n\times n}$, and $\tau = \sqrt{\epsilon}$.

The numerical experiments have been carried out on the same set of test functions as in Bierlaire & Crittin (2003a), with dimensions 10 and 50, and using the identity matrix as the initial approximation of the Jacobian.

The results are presented using the performance profiles proposed by Dolan & More (2002). It is a powerful visual tool for evaluating and comparing the performance of several algorithms applied to many problems. The performance profile for a method is the cumulative distribution function for a given performance metric.

In the following we use the number of function evaluations to reach convergence as performance metric. If $f_{p,a}$ is the performance metric of algorithm a on problem p, then the *performance ratio* is defined by

$$r_{p,a} = \frac{f_{p,a}}{\min_a\{f_{p,a}\}}, \tag{10}$$

if algorithm a has converged for problem p, and $r_{p,a} = r_{\text{fail}}$ otherwise, where r_{fail} must be strictly larger than any performance ratio (10). For any given threshold π, the overall performance of algorithm a is given by

$$\rho_a(\pi) = \frac{1}{n_p}\Phi_a(\pi) \tag{11}$$

where n_p is the number of problems considered, and $\Phi_a(\pi)$ is the number of problems for which $r_{p,a} \leq \pi$.

In particular, the value $\rho_a(1)$ gives the probability that algorithm a wins over all other algorithms. The value $lim_{\pi\to r_{\text{fail}}}\rho_a(\pi)$ gives the probability that algorithm a solves a problem and, consequently, provides a measure of the robustness of each method.

The performance profile presented in Figure 1 shows that iGSM (with large-scale features) has a similar behavior than GSM, in comparison to ICUM and the Hybrid Method. Actually,

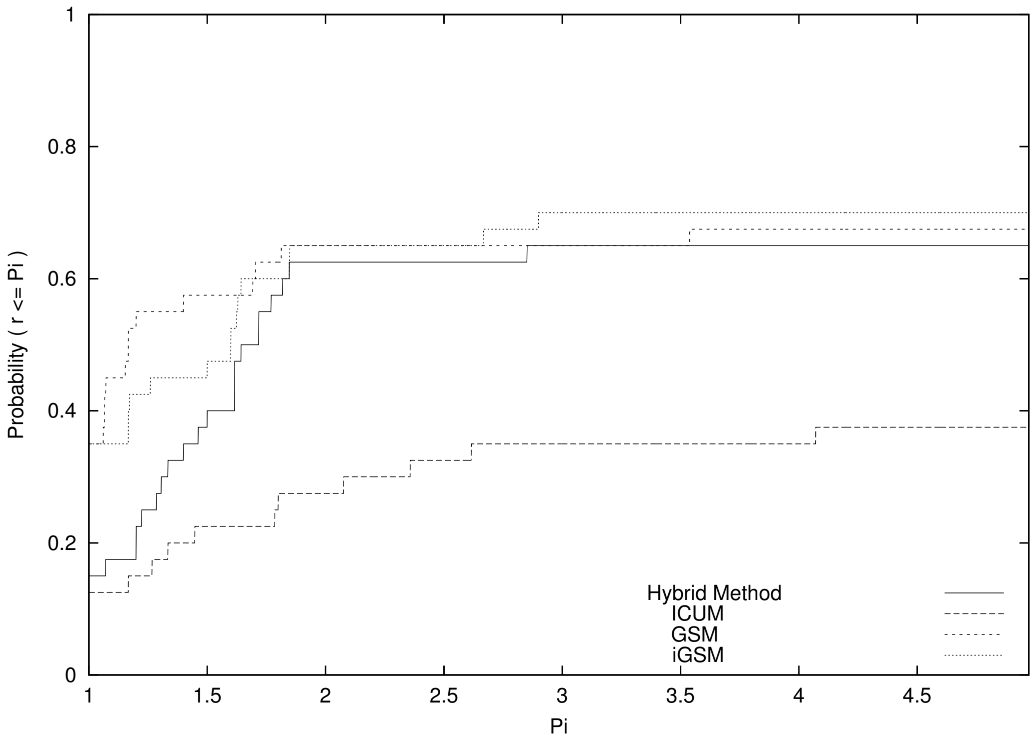


Figure 1: Performance profiles for medium scale problems

GSM performs a little faster (small values of π), while iGSM seems a little more robust (high values of π), but we do not consider these differences being significant. This is encouraging, as the large scale features are not associated with a loss of performance, compared to the original GSM design.

Comparing iGSM and ICUM (both designed for large scale problems and using the identity matrix as initial approximation of the Jacobian), iGSM solves about 70% of the problems, while ICUM solves about 40%. iGSM is the fastest method for 35% of the problems, and so is GSM.

The numerical experiments for the large scale problems were carried out using a set of 33 problems. The first 18 problems are classical nonlinear system of equations: Countercurrent reactors problem [CRP] (Bogle & Perkins 1990), Extended Powell badly scaled function [EPBSF] ((Moré, Garbow & Hillstrom 1981)), A trigonometric system [TS] (Toint 1986), A trigonometric-exponential system (I) [TESI] (Toint 1986), Singular Broyden problem [SBP] (Gomes-Ruggiero et al. 1991), Tridiagonal System [TdS] (Li 1989), Five-diagonal system [FdS] (Li 1989), Seven-diagonal system [SdS] (Li 1989), Structured Jacobian problem [SJP] (Gomes-Ruggiero et al. 1991), Extended Rosenbrock function [ERF] (Luksan 1994), Extended Powell singular function [EPSF] (Luksan 1994), Extended Gragg and Levy function [EGLF] (Luksan 1994), Broyden tridiagonal function [BTF] (Moré et al. 1981), Broyden Banded problem [BBP] (Moré et al. 1981), Discrete boundary value problem [DBVP] (Moré et al. 1981), Chandrasekhar H-equation residual [CHR] (Kelley 2002), Ornstein-Zernike Equation [OZE]

(Kelley 2002), Convection-Diffusion Equation [CDE] (Kelley 2002). The remaing 15 systems without name have been selected from (Spedicato & Huang 1997), the number in brackets coincide with the numbering used in the paper. [SPED1], [SPED2], [SPED4], [SPED5], [SPED6], [SPED7], [SPED9], [SPED12], [SPED13], [SPED17], [SPED18], [SPED20], [SPED22] have been proposed by Roose, Kulla, Lomp & Meressoo (1990). [SPED27] has been proposed by M.Robert & Shipman (1976) and [SPED28] by Ascher & Russel (1985).

The runs were done with $n = 100$ and $n = 1000$ for all problems, except the Ornstein-Zernike Equation where $n = 402$ and the right preconditioned Convection-Diffusion Equation with $n = 961$. For the three problems [CHR], [OZE] and [CDE] we have used the implementation furnished by Kelley (2002). For each problem, we have used the starting point proposed in the original papers. All the problems and methods have been implemented using Octave.

The performance of iGSM and ICUM may be sensitive to the first matrix B_0. There is a trade-off here between the resources consumed to compute B_0 and its impact on the methods efficiency. In addition to the default $B_0 = I_{n\times n}$, we have tested 4 different possibilities for B_0, all based on a banded approximation of the Jacobian obtained from finite differences, that is a diagonal, a 3-diagonal, a 5-diagonal and a 7-diagonal matrix. Note that a p-diagonal approximation of the Jacobian can be evaluated with p function evaluations (Kelley 2002).

A detailed description of the results is presented in Bierlaire & Crittin (2003b) as well as the performance profiles of all considered methods on all problems. Figure 2 plots the profile of each instances of iGSM.

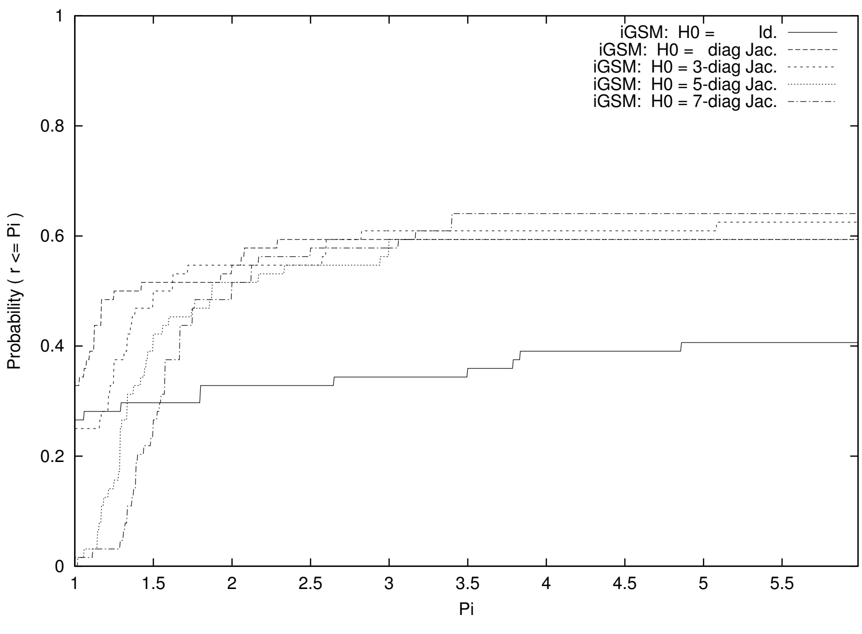
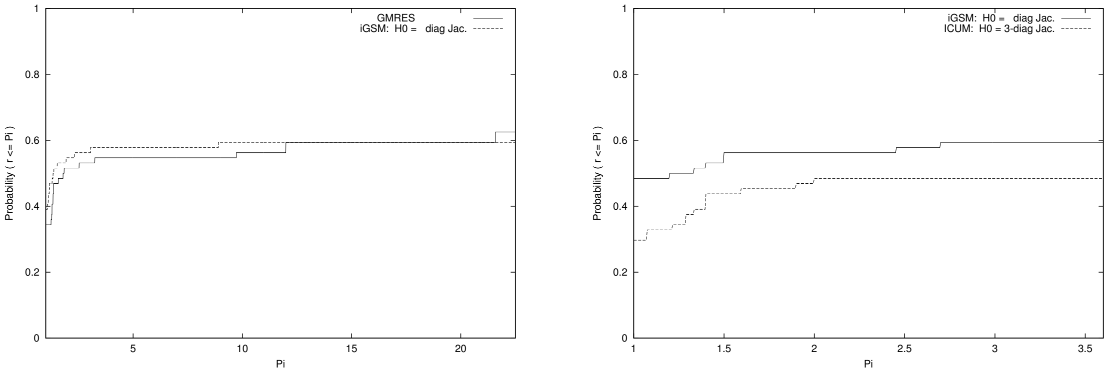


Figure 2: Performance profiles of iGSM with different B_0

It clearly emphasizes that $B_0 = I_{n\times n}$ is a poor choice in terms of robustness for the method. As anticipated the best approximation (7-diagonal) produces the most robust method as it solves nearly 65% of the problems. But choosing a diagonal or a 3-diagonal approximation of the Jacobian seems to be a better compromise between efficiency and reliability. For higher order approximation (5 or 7-diagonal approximation) the efficiency is penalized by the number of function evaluations necessary to compute the initial Jacobian. In the following, we choose to start iGSM with a diagonal B_0. We have performed a similar analysis for ICUM. In this case, we have decided to select the 3-diagonal B_0.



(a) GMRES and iGSM (b) ICUM and iGSM

Figure 3: Head-to-Head Performance Profiles

Comparing iGSM and ICUM (Figure 3(b)), we observe that iGSM significantly outperforms ICUM, both in efficiency and in robustness. For the comparison of iGSM with a Newton-Krylov method, we have used a matrix-free implementation (Kelley 2002) of the GMRES method (Saad & Schultz 1986). The behavior of the two methods (see Figure 3(a)) is comparable. These results are confirmed when the three methods are compared together (Figure 4).

These results are very encouraging. In the quasi-Newton literature, ICUM is currently considered as one of the best methods solving large scale nonlinear systems of equations without derivatives. The significant improvement brought by iGSM is noticeable. Moreover, Newton-Krylov methods are recognized as the most efficient methods to date to solve large scale problems (Kelley 2002). Therefore, the fact that our quasi-Newton approach reaches the same level of performance is an achievement, taking into consideration that the implementation of GMRES involves globalization techniques which is not the case for iGSM.

3.2 The Convection-Diffusion problem

To illustrate the behavior and the potential of our method, we propose to analyze more deeply the Convection-Diffusion problem mentioned by Kelley (2002). It is is a semi-linear convection-diffusion equation of size 961. As the problem is difficult, we use the preconditioner proposed by Kelley (2002). The objective function is right preconditioned using a fast Poisson solver.

We analyze the impact of the population size on the efficiency of iGSM. Table 3.2 reports, for each size of the population, the number of function evaluations and the time to reach convergence. With a population of two iterates, the method does not converge. Clearly, using more than two iterates significantly improves the algorithm performance, illustrating the added

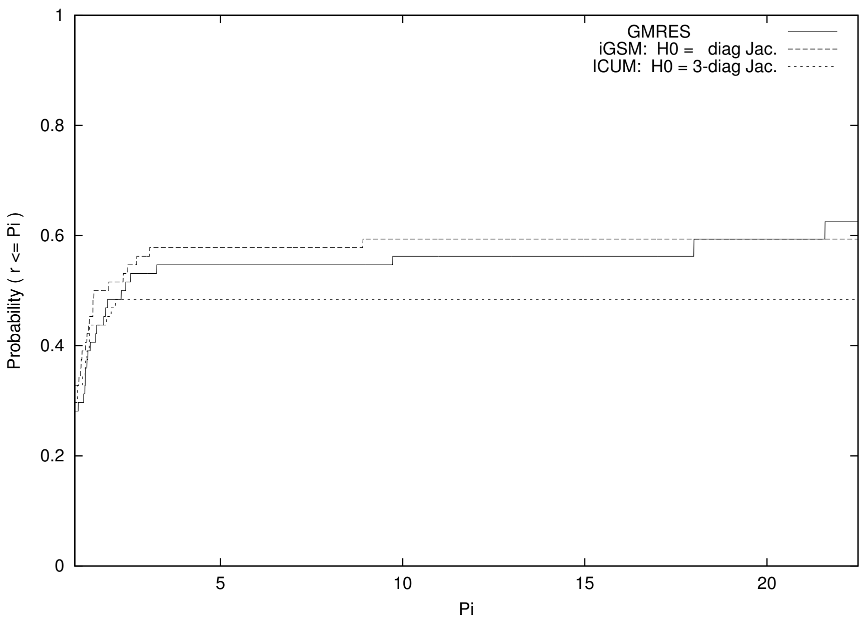


Figure 4: Three Algorithms Performance Profiles

value of our generalized approach contrasting with other classical quasi-Newton methods. More interestingly, the performance improvement reaches a plateau around a population of 10.

Figure 5 reports the evolution of the relative residual $\frac{\|F(x_k)\|}{\|F(x_0)\|}$ with the number of function evaluations for iGSM (with 10 iterates in the population) and GMRES. Note that ICUM does not converge on this problem. Clearly, iGSM converges faster than GMRES. Remark the horizontal "steps" of the GMRES curve. They correspond to stages of the algorithm when partial derivatives of F are evaluated by finite difference. Moreover the computational time to reach convergence on this problem is 14.1 sec. for iGSM and 26.4 sec. for GMRES.

3.3 Behavior in the presence of noise

We present a preliminary analysis of the behavior of these methods in the presence of noise. Indeed, it is an important motivation for population-based methods, as discussed by Bierlaire & Crittin (2003a).

Following Choi & Kelley (2000), we consider a smooth deterministic system of nonlinear equations $F_s(x) = 0$, and define a noisy version as

$$G(x) = F_s(x) + \phi(x) \tag{12}$$

where $\phi(x)$ is a white noise such that its variance decreases in the vicinity of the solution, that is

$$\phi(x) \sim N(0, \alpha^2 \|x - x^*\|^2), \tag{13}$$

where $\alpha \in \mathbb{R}$.

Population size	Function evaluations	Time [sec]
2	max	-
3	43	30.3
4	38	26.7
5	36	26.2
6	29	21.6
7	21	15.9
8	21	16.2
9	19	14.9
10	18	14.1
11	17	14
12	18	14.9
13	18	15.4
14	18	15.6
15	19	16.8
16	19	16.9
17	19	17.3
18	19	17.4
19	19	17.2
20	19	17.2

Table 1: Impact of the size of the population on iGSM

We have selected problem [SPED12], as the behavior of GMRES, ICUM and iGSM is almost the same in the deterministic case (see Figure 6(a)). It is defined as

$$\begin{cases} f_1 = x_1, \\ f_i = cos(x_{i-1}) + x_i - 1, \quad \forall i = 1, \ldots, n \end{cases} \tag{14}$$

with initial point $x_0 = (0, \ldots, 0)$. In the presence of a very small noise ($\alpha = 10^{-9}$, Figure 6(b)), we observe a slight decrease of the performance of the Newton-Krylov method, compared to the two others which are not notably affected. It makes sense, as Newton-Krylov relies on finite difference approximations of the derivatives, which are sensitive to the noise. When the magnitude of the noise increases ($\alpha = 10^{-4}$, Figure 6(c)), the GMRES method is stalled at the starting point. The ICUM method does not make much progress in the early iterations, and starts diverging after about 15 function evaluations. A similar phenomenon is observed for the larger noise ($\alpha = 10^{-1}$, Figure 6(d)). For this problem iGSM converges even in presence of a high level of stochasticity.

We have also considered problems with a constant noise, that is where

$$\phi(x) \sim N(0, \alpha^2). \tag{15}$$

In this case, the performance of Newton-Krylov is much more affected by a small noise ($\alpha = 10^{-9}$, Figure 7(b)). For the medium noise ($\alpha = 10^{-4}$, Figure 7(c)) and the large noise ($\alpha = 10^{-1}$, Figure 7(d)), we observe again the same behavior as before: Newton-Krylov is stalled, ICUM decreases first and then explodes, and iGSM decreases to reach a level where no more progress is made due to the definition of the problem.

These analysis emphasizes the added value of the iGSM method on the Newton-Krylov approach. If they perform similarly on deterministic problems, iGSM is much more robust in the presence of noise.

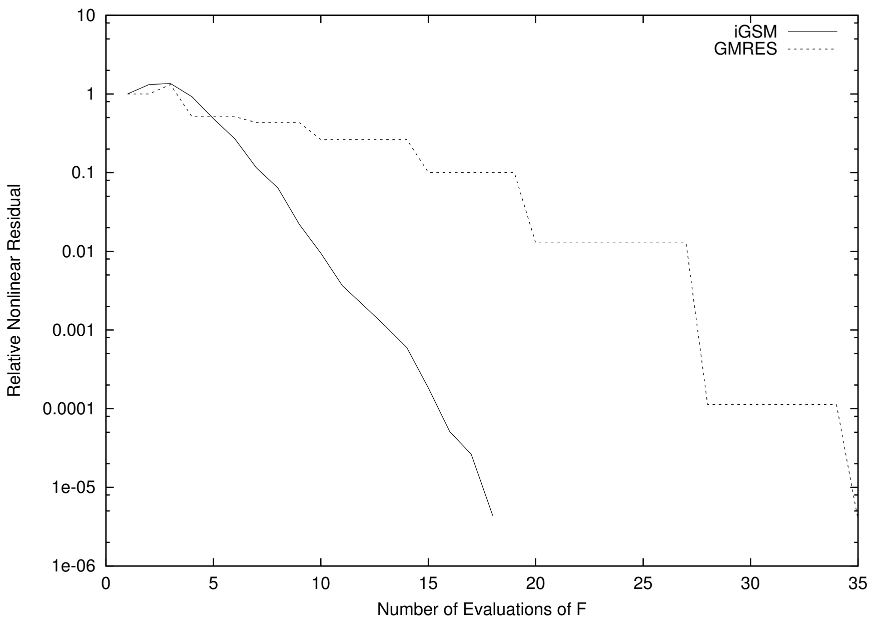


Figure 5: Convection-Diffusion equation

We provide similar analysis on two other problems: the Chandrasekhar H-equation residual problem and the Broyden Banded problem in Bierlaire & Crittin (2003b).

3.4 The CARG problem

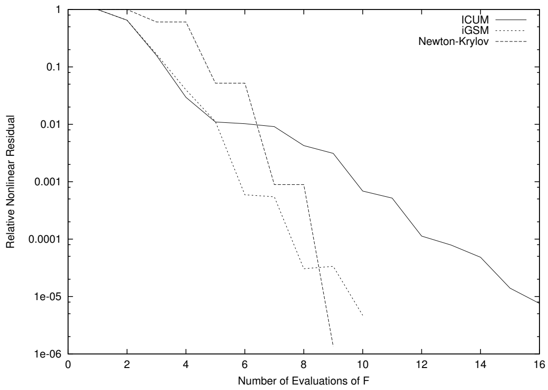
We conclude this section by analyzing a difficult problem which is both large and noisy.

Route guidance refers to information disseminated to road users with the intent of influencing their route choice decisions. We are interested here in *anticipatory* route guidance where real-time traffic conditions are used to make predictions of the evolution of the network. So the information provided to a driver will reflect the conditions that are expected to prevail at network locations at the times when he will actually be there.

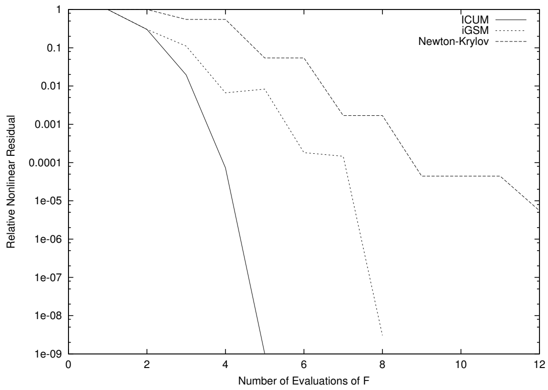
A tricky problem in generating anticipatory route guidance is the fact the system under consideration is affected by the dissemination of information. Indeed, contrarily to weather forecast, the reactions of the users receiving the guidance can affect the future conditions of the network and therefore invalidate the predictions on which the guidance was based. The anticipatory guidance is said to be *consistent* if the predictions on which the guidance is based are the same as those that are forecast to result after drivers react to the guidance.

This problem was introduced by Ben-Akiva, de Palma & Kaysi (1996) and a fixed point formulation has been proposed by Bottom, Ben-Akiva, Bierlaire, Chabini, Koutsopoulos & Yang (1999) and developed by Bottom (2000) in his PhD dissertation. This problem is defined by:

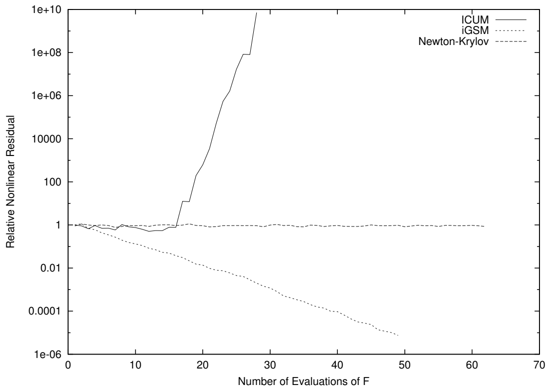
$$\text{Find } x \text{ such that } x = \mathcal{T}(x) \tag{16}$$



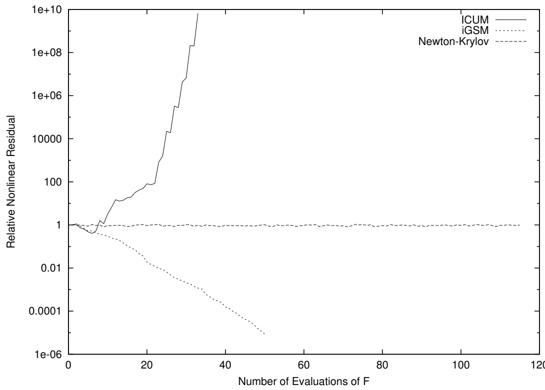
(a) Without noise



(b) Small variance noise



(c) Medium variance noise

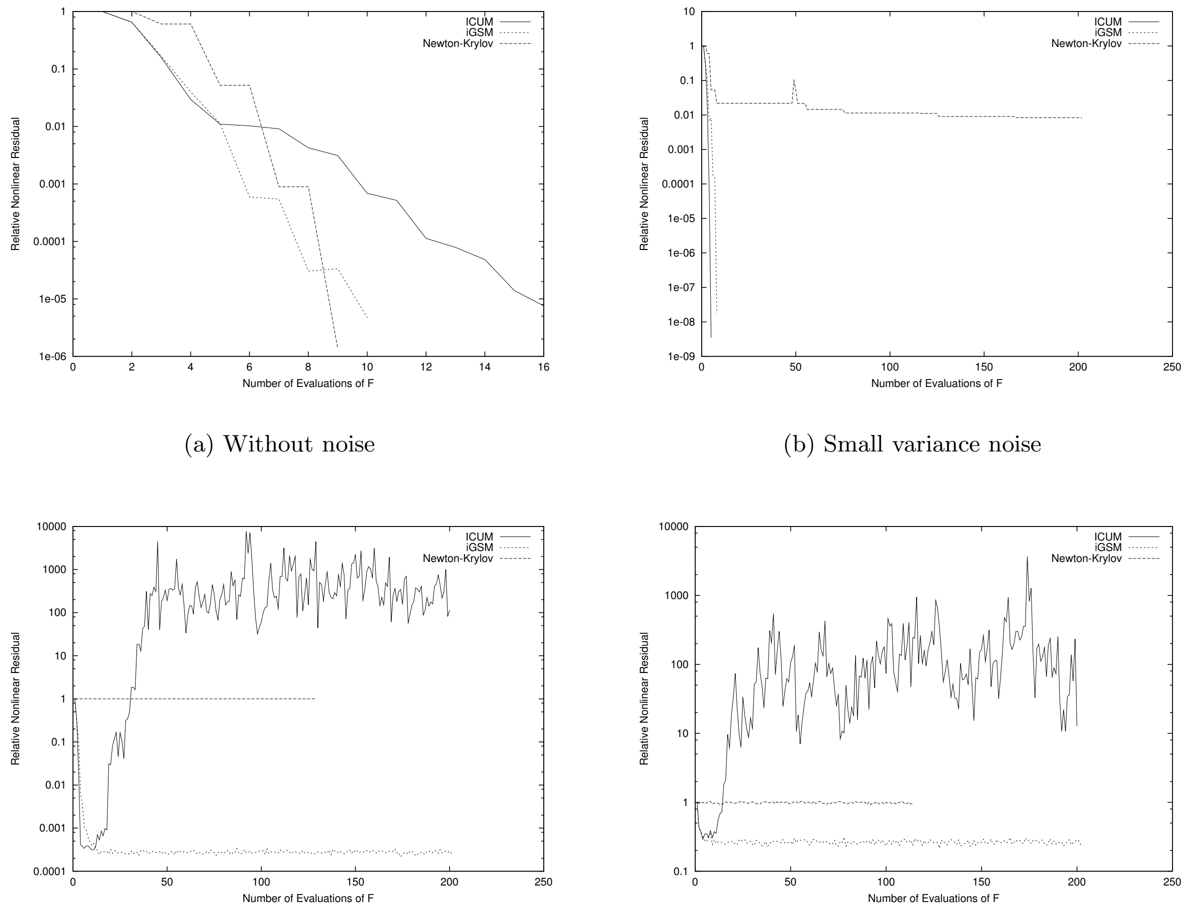


(d) Big variance noise

Figure 6: Behavior with stochasticity

The complexity of transportation system and the necessity of capturing traveler behavior impose the use of disaggregated models and simulation-based tools to compute $\mathcal{T}$. Consequently this fixed point problems, defined in (16), is non-analytical and stochastic. Moreover x involves a high number of variables.

Note that averaging methods defined by (1) can be considered as quasi-Newton methods where B_k is defined at each iteration k by $B_k = \alpha_k I_{n\times n}$. We provide a numerical comparison between an averaging method and iGSM to solve the consistent anticipatory route guidance problem in order to illustrate the applicability of this method to very large scale problems. This test has been done using DynaMIT is a state-of-the-art, real-time computer system for traffic estimation prediction and generation of traveler information and route guidance. DynaMIT is the result of about 10 years of intense research and development at the Intelligent Transportation Systems Program of the Massachusetts Institute of Technology (for description and details, see (Ben-Akiva, Bierlaire, Koutsopoulos & Mishalani 2002), (Bottom et al. 1999) and (Ben-Akiva, Bierlaire, Koutsopoulos & Mishalani 1998)). DynaMIT is designed to operate in real time, using traffic volume and control system state data to estimate and predict time-dependent



(a) Without noise

(b) Small variance noise

(c) Medium variance noise

(d) Big variance noise

Figure 7: Behavior with stochasticity

origin-destination flows and network conditions, and generating descriptive and prescriptive information that should be consistent with the predicted traffic conditions. DynaMIT's consistent guidance generation algorithm is currently an averaging method with $\alpha_k = 0.5$ for all k. iGSM has been implemented in C++ in DynaMIT with the following parameters: as initial approximation of the Jacobian we choose $B_0^{-1} = I_{n \times n}$ and the size of population is set to $\kappa = 5$. Computations have been done with the same laptop. The network considered is large-scale network representing the swiss highway system from Geneva to Schaffausen and is composed of 1661 links. We simulate from 7h00 to 8h15 in the morning with time interval of one minute and analyze the guidance generation for 75 minutes. The size of the fixed point problem associated with the CARG problem is 124'575 (1661×75). It appears in Figure 8 that iGSM decreases the consistency very fast during the first iterates, after which it seems to struggle in spite of the logarithmic scale. The averaging method reaches the same consistency about 28 iterations later. In terms of real-time applications, the fast decreasing consistency, at the beginning, associated with iGSM algorithm seems a very good alternative to averaging methods. More results about the CARG are discussed in Bierlaire & Crittin (2003c).Those preliminary results on the consistent anticipatory route guidance problem are very encouraging, principally for

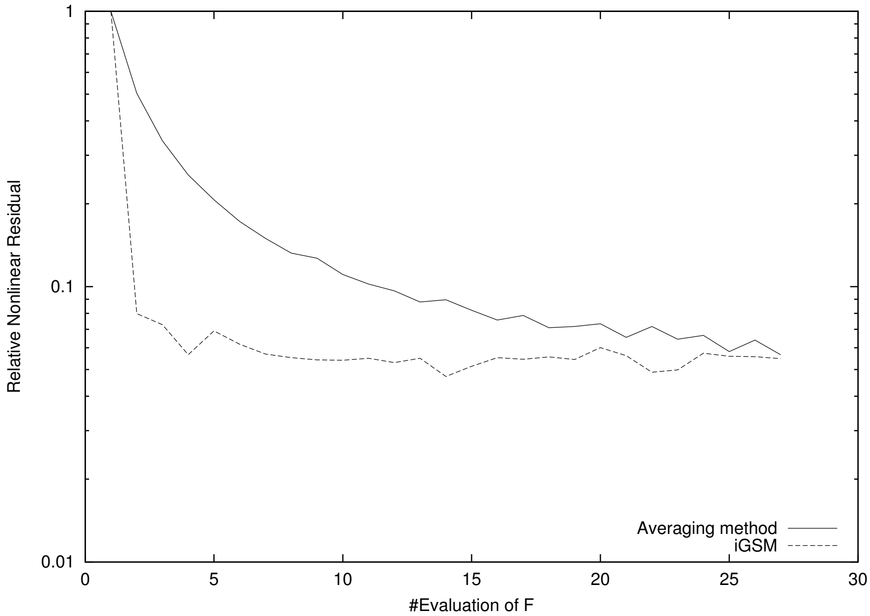


Figure 8: Swiss Network

real-time applications.

References

Ascher, U. & Russel, R. D. (1985). Numerical boundary value odes, Boston Birkhauser.

Ben-Akiva, M., Bierlaire, M., Koutsopoulos, H. N. & Mishalani, R. (2002). Real-time simulation of traffic demand-supply interactions within DynaMIT, *in* M. Gendreau & P. Marcotte (eds), *Transportation and network analysis: current trends. Miscellenea in honor of Michael Florian*, Kluwer Academic Publishers, Boston/Dordrecht/London.

Ben-Akiva, M. E., Bierlaire, M., Koutsopoulos, H. N. & Mishalani, R. G. (1998). DynaMIT: a simulation-based system for traffic prediction, *Proceedings of the DACCORD Short-Term forecasting workshop*.

Ben-Akiva, M. E., de Palma, A. & Kaysi, I. (1996). The impact of predictive information on guidance efficiency: an analytical approach, *in* L. Bianco & P. Toth (eds), *Advanced Methods in Transportation Analysis*, Springer-Verlag, pp. 413–432.

Bierlaire, M. & Crittin, F. (2003a). Generalization of secant methods for solving systems of nonlinear equations, *Proceedings of the 3rd Swiss Transportation Research Conference*, Ascona, Switzerland.

Bierlaire, M. & Crittin, F. (2003b). Solving large-scale systems of nonlinear equations, *Technical Report RO-2003.0804*, Swiss Insitute of Technology, Lausanne, ROSO-DMA-EPFL CH-1015 Lausanne. Submitted to Mathematical Programming.

Bierlaire, M. & Crittin, F. (2003c). Solving the anticipatory route guidance generation problem using a generalization of secant method, *Proceedings of the 3rd Swiss Transportation Research Conference*, Ascona, Switzerland.

Bogle, I. & Perkins, J. (1990). A new sparsity preserving quasi-Newton update for solving nonlinear equations, *SIAM Journal on scientific and statistical computations* **11**: 621–630.

Bottom, J. (2000). *Consistent Anticipatory Route Guidance*, PhD thesis, Massachusetts Institute of Technology.

Bottom, J., Ben-Akiva, M., Bierlaire, M., Chabini, I., Koutsopoulos, H. & Yang, Q. (1999). Investigation of route guidance generation issues by simulation with DynaMIT, *in* A. Ceder (ed.), *Transportation and Traffic Theory. Proceedings of the 14th ISTTT*, Pergamon, pp. 577–600.

Broyden, C. G. (1965). A class of methods for solving nonlinear simultaneous equations, *Mathematics of Computation* **19**: 577–593.

Byrd, R., Nocedal, J. & Schnabel, R. B. (1994). Representation of quasi-Newton matrices and their use in limited memory methods, *Mathematical Programming* **63**: 129–136.

Cantarella, G. E. (1997). A general fixed-point approach to multimode multi-user equilibrium assignement with elastic demand, *Transportation Science* **31**(2): 107–128.

Choi, T. D. & Kelley, C. (2000). Superlinear convergence and implicit filtering, *Siam Journal of Optimization* **10**.

Dennis, J. E. & Schnabel, R. B. (1996). *Numerical methods for unconstrained optimization and nonlinear equations*, Society for Industrial and Applied Mathematics (SIAM).

Dolan, E. & More, J. (2002). Benchmarking optimization software with performance profiles, *Mathematical Programming, Serie A* **91**: 2.

Eaton, J. W. (1997). GNU Octave: a high level interactive language for numerical computations, `www.octave.org`.

Gomes-Ruggiero, M. A., Martinez, J. M. & Moretti, A. C. (1991). Comparing algorithms for solving sparse nonlinear systems of equations, *Siam J. Sci. Stat Comp.* .

Kelley, C. T. (2002). *Solving nonlinear equations with Newton's method*, Draft stage. http://www4.ncsu.edu/~ctk/newton.html.

Li, G. (1989). Successive column correction algorithms for solving sparse nonlinear systems of equations, *Mathematical Programming* **43**: 187–207.

Luksan, L. (1994). Inexact trust region method for large sparse systems of nonlinear equations, *Journal of Optimization Theory and Applications* **81**(3): 569–590.

Martinez, J. M. (1982). Sobre dois metodos de broyden, *Matematica Aplicada e Computational* **1**.

Martinez, J. M. (2000). Practical quasi-newton methods for solving nonlinear systems, *Journal of Computational and Applied Mathematics* **124**: 97–122.

Martinez, J. M. & Zambaldi, M. C. (1992). An inverse column-updating method for solving large-scale nonlinear systems of equations, *Optimization method and software* **1**.

Moré, J. J., Garbow, B. S. & Hillstrom, K. E. (1981). Testing unconstrained optimization software, *ACM Transactions on Mathematical Software* **7**(1): 17–41.

M.Robert, S. & Shipman, J. S. (1976). On the closed form solution of troesch's problem, *J. Comput. Physics* .

Polyak, B. T. & Juditsky, A. B. (1992). Acceleration of stochastic approximation by averaging, *SICON* **30**(4): 838–855.

Roose, A., Kulla, V., Lomp, M. & Meressoo, T. (1990). Test examples of systems of non-linear equations, *Estonian software and Computer Service Company* .

Ruggiero, M. G., D.N.Kozakevich & J.M.Martnez (1996). A numerical study on large-scale nonlinear solvers, *Computers and Mathematics with Applications: An International Journal* **32**: 1–13.

Saad, Y. & Schultz, M. (1986). Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM Journal on Numerical Analysis* **7**: 856–869.

Sheffi, Y. & Powell, W. (1982). An algorithm for the equilibrium assignment problem with random link times, *Networks* **12**: 191–207.

Spedicato, E. & Huang, Z. (1997). Numerical experience with newton-like methods for non-linear algebraic systems, *Computing* **58**: 69–99.

Toint, P. L. (1986). Numerical solution of large sets of algebraic nonlinear equations, *Mathematics of Computation* **46**(173): 175–189.