

A Decoupled Time-Domain Simulation Method via Invariant Subspace Partition for Power System Analysis

Dan Yang, *Student Member, IEEE*, and Venkataramana Ajjarapu, *Senior Member, IEEE*

Abstract—A decoupled method is proposed to deal with time-domain simulation for power system dynamic analysis. Traditionally, there are two main categories of numerical integration methods: explicit methods and implicit methods. The implicit methods are numerically stable but require more computational time to solve the nonlinear equations, while explicit methods are relatively efficient but may cause a numerical stability problem. This paper proposes a new hybrid method to take advantage of both explicit and implicit methods based on the invariant subspace partition. The original power system equations are decoupled into two parts that correspond to the stiff and nonstiff subspaces. For the stiff invariant subspace, the implicit method is applied to achieve numerical stability, and the explicit method is employed to handle nonstiff invariant subspace for the computational efficiency. As a result, the new hybrid method is both numerically stable and efficient. The approach is demonstrated through New England 39-bus and IEEE 118-bus systems.

Index Terms—A-stability, decoupled method, differential and algebraic equations (DAE), dynamical simulation, explicit methods, implicit methods, invariant subspace partition, power systems, stiff systems, time-domain simulation.

I. INTRODUCTION

TIME-domain simulation is an important tool for power system dynamic analysis. A set of differential and algebraic equations (DAE) are numerically solved to study the transient behavior of power systems. Power systems networks typically include thousands of generators, exciters, governors, loads, transformers, and other power electronics devices, where each individual component may need several differential and algebraic equations to represent; thus, the total number of DAEs of a real power system can be formidably large.

Time-domain simulations for the power systems include step-by-step numerical integration of DAEs. The numerical error introduced in each step can be measured via the local truncation error. To improve the accuracy, small step size and higher order approximations are usually required. The error accumulated in each step may also yield qualitatively wrong results, and numerical stability analysis is needed to guarantee the correctness.

Power system simulations involve various system components whose time constants vary in a large range. As a result,

time-domain simulations face numerical problems related to stiffness caused by different time scales. A considerable amount of effort has been spent to solve large-scale stiff problems in the literature [1]–[7].

Numerical integration methods can be classified into two categories: explicit methods and implicit methods. The explicit methods involve fixed-point iteration and are computationally efficient but have a numerical stability problem when dealing with stiff problems. The implicit methods involve solving nonlinear equations at each step. The implicit methods are slow but stable. Implicit methods are commonly used for solving power system dynamic simulation, and research effort has been done to improve computational efficiency. In [3], the trapezoidal method is improved by reducing a large nonlinear system into two small systems to study post fault power system dynamics for prediction use, and Newton method is used to solve each small system.

In this paper, a new method that combines explicit methods and implicit methods to solve the time-domain simulation is proposed. The main motivation is to propose an algorithm to take advantage of both methods: efficiency and stability. In the simulation process, the large-scale DAEs are decoupled into two parts to treat them separately based on invariant subspace partition to achieve the goals.

In the mathematical literature, the idea of partitioning a stiff system into a small stiff system and a large nonstiff part goes back to Hofer [8]. In [8], the differential equations are decomposed by the time constants of the equations where a few distinct equations introducing the small time constants are considered the stiff part. As stiffness may affect subspaces that are not parallel to the coordinate axis, the projection methods are applied as generalized partition methods later. In [9]–[11], the solutions of linear equations are approximated by the projection to the dominant subspace (stiff subspace) that is chosen as a Krylov subspace; the crude or precise partitions are accomplished by block QR iterations and the block power method in [12]–[14]. In this paper, the projection is based on Lyapunov–Schmidt decomposition of differential equations where the stability domains are explicitly considered, and the solutions are approximated in both stiff and nonstiff subspaces.

The remaining parts of this paper are organized as follows. A brief comparison between explicit and implicit methods and the stability of the numerical integration is introduced in Section II. Section III describes the decoupled scheme based on invariant subspace partition for ordinary differential equations (ODE) systems. Section IV extends this approach to the DAE

Manuscript received January 19, 2005; revised April 14, 2005. Paper no. TPWRS-00033-2005.

The authors are with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: yangdan@iastate.edu; va-ajarapu@iastate.edu).

Digital Object Identifier 10.1109/TPWRS.2005.860937

model and also demonstrates the identification of small stiff subspace. Section V provides numerical results to demonstrate the advantages of this approach, and the conclusions are given in Section VI.

II. CONVENTIONAL EXPLICIT AND IMPLICIT METHODS AND NUMERICAL STABILITY

Consider a general ODE system with a given initial condition as described by

$$\begin{cases} \dot{x} = f(x) \\ x(0) = x_0. \end{cases} \quad (1)$$

To solve the above initial value problem (IVP), the following two approaches are commonly employed [15]–[20].

A. Explicit Method

Explicit methods typically replace the ordinary differential equations by nonlinear recursive mappings

$$x_{k+1} = g(x_k). \quad (2)$$

Here the recursion is to be understood in the sense that from some initial value x_0 , the states x_1, x_2, \dots are generated as long as they remain in the domain of definition of the mapping.

Some of the methods under this category include: forward Euler method; explicit Runge–Kutta methods; and Adams–Bashforth methods.

The forward Euler method is formulated as

$$x_{k+1} = x_k + hf(t_k, x_k). \quad (3)$$

The fourth-order Runge–Kutta method is formulated as

$$\begin{aligned} K_1 &= f(t_k, x_k) \\ K_2 &= f(t_k + h/2, x_k + hK_1/2) \\ K_3 &= f(t_k + h/2, x_k + hK_2/2) \\ K_4 &= f(t_k + h, x_k + hK_3) \\ x_{k+1} &= x_k + h(K_1 + 2K_2 + 2K_3 + K_4)/6. \end{aligned} \quad (4)$$

The advantage of explicit methods is that at each simulation step, the next step value can be directly calculated from previous results. Thus, explicit methods are very efficient with fixed-point iteration techniques.

B. Implicit Method

The implicit methods use both current state and past state to solve the initial value problem. As a result, a set of nonlinear equations needs to be solved at each individual step, which causes a higher computational burden compared with the explicit methods.

Backward Euler method, trapezoidal method, implicit Runge–Kutta method, Adams–Moulton methods, and backward differential formulas methods are part of the whole family of the implicit methods.

The backward Euler method is formulated as

$$x_{k+1} = x_k + hf(t_{k+1}, x_{k+1}). \quad (5)$$

The trapezoidal method is formulated as

$$x_{k+1} = x_k + h[f(t_k, x_k) + f(t_{k+1}, x_{k+1})]/2. \quad (6)$$

In the formulas of implicit methods, the next step state cannot be obtained directly, and the Newton method is usually used to solve the nonlinear equations in the implicit methods. However, the implicit methods have better numerical stability properties than explicit methods, despite their slow computational performance.

C. Numerical Stability Analysis: Stiffness and Linear Stability (A-Stability)

Power systems usually have components with vastly different time scales. The problems of such systems are described as being stiff. The time constants of the various physical processes differ greatly (from tens of milliseconds for fast transients up to 100 s for slow adjustments). Physically speaking, the stiffness is caused by the presence of different time-scale components, while mathematically speaking, the stiffness of the problem is associated with the existence of both large and small eigenvalues. The quotient of the largest and the smallest eigenvalues can be considered as the stiffness ratio to measure the degree of stiffness [18].

The numerical methods may produce qualitatively wrong results for stiff problems due to the error accumulations in the simulation steps. It is crucial that the numerical methods correctly identify whether the system is stable or unstable. In some cases, the numerical solution may indicate unstable behavior for the case where the actual system is stable and vice versa. Explicit methods may require a significant reduction of the step size to maintain numerical stability such that the step size is smaller than the step size needed to represent the solution accurately. The required step size for explicit methods to guarantee numerical stability may be too small for practical implementation. The implicit methods are necessary to guarantee the numerical stability for stiff systems.

A simple example given in [18] is used to show the difference between explicit and implicit methods for stiff systems

$$\begin{cases} \dot{x}_1 = -100x_1 + x_2 \\ \dot{x}_2 = -0.1x_2. \end{cases} \quad (7)$$

The initial value is $(-3, -1)^T$, and the step size is chosen as 0.1. The eigenvalues of (7) are -100 and -0.1 , which are both negative; therefore, the system trajectory should converge to the origin as time goes into infinity. To obtain the time response, both the forward Euler method and the trapezoidal method are applied. The result by the forward Euler method is shown in Fig. 1(a). Actually, in this case, the forward Euler iterates grow geometrically in magnitude if the step size is greater than 0.02, in contrast with the asymptotic behavior of the true solution. The simulation result by the trapezoidal method is shown in Fig. 1(b), which demonstrates proper stable behavior.

To analyze possible instability caused by numerical methods, the concepts of A-stability and stability domain are proposed in the literature. Suppose that a given numerical method is applied with a step size $h > 0$ to the linear test system $\dot{x} = \lambda x$, the stability domain of the underlying numerical method is the set

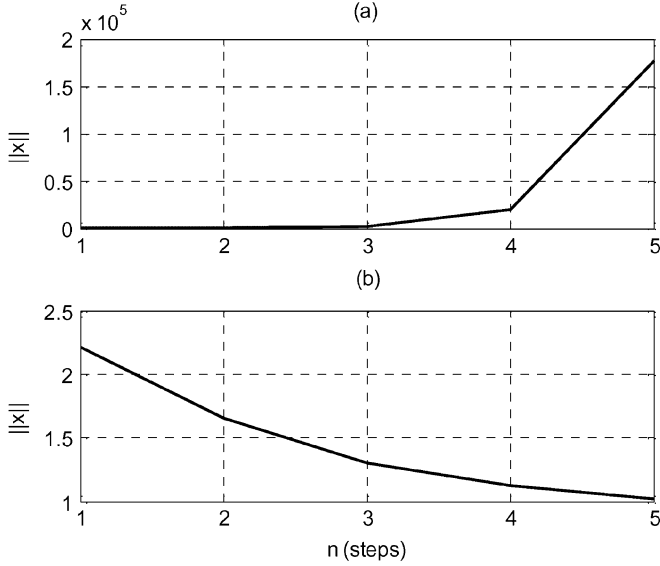


Fig. 1. Simulation results by explicit and implicit methods. (a) Forward Euler method. (b) Trapezoidal method.

of all numbers $h\lambda$ such that x_n approach zero as $n \rightarrow \infty$. In other words, the stability domain is the set of all $h\lambda$ for which the correct asymptotic behavior is recovered, provided that the linear system is stable.

A method is *A-stable* if x_n approach zero as $n \rightarrow \infty$ for all values of the step size h when this method is applied to the equation $\dot{x} = \lambda x$ for all $\lambda \in \mathbb{C}$ with $\text{Re}(\lambda) < 0$. Note that for this equation, the exact solution also goes to zero. In other words, for $\text{Re}(\lambda) < 0$, the solution of corresponding differential equations should be stable for any positive value of h . It implies that the stability domain includes the whole left half plane. Thus, whether or not a method is *A-stable* can be judged from the stability domain. The stability domains of the forward Euler and trapezoidal methods are shown in Fig. 2. The forward Euler method is not *A-stable*, while the trapezoidal method is *A-stable*. It is proven that no explicit Runge–Kutta method may be *A-stable* [18]. In general, an *A-stable* linear multistep method is necessarily implicit, but not every implicit method is *A-stable*.

The drawback of *A-stability* is that the stability domain may include part of the right half plane; thus, the real unstable phenomena will be simulated as a stable one. The spurious damping is called hyper stability [1]. Hyper stability can be avoided by reducing the step size during the simulation on the basis of the experience of the end user or the evaluation of eigenvalues.

III. DECOUPLED METHOD

Since power systems are stiff problems, implicit methods are commonly used to simulate the dynamic behavior. Each integration step of a stiff equation involves the solution of a non-linear equation, which leads to a set of linear problems involving the Jacobian of the system. As a result, the methods for solving stiff systems spend most of the time solving systems of linear equations. The numerical stability properties of the time-domain simulation algorithms are determined by the eigenvalues of the linearized matrix, and frequently, the eigenvalues that cause stiff problems are only a small portion of the whole spectra. It seems

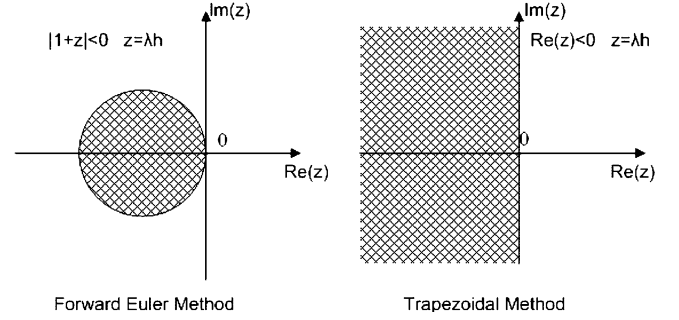


Fig. 2. Stability domain of forward Euler and trapezoidal methods.

inefficient to solve these problems only with implicit methods. If the problem can be partitioned into a stiff part and a nonstiff part such as

$$\begin{cases} \dot{x}_s = f_s(x_s, x_n) \\ \dot{x}_n = f_n(x_s, x_n) \end{cases} \quad (8)$$

where f_s, f_n are representations of stiff and nonstiff subsystems with variables x_s and x_n , respectively. The system can be treated with an implicit method for the stiff equations and an explicit method for the nonstiff equations [16].

In the numerical stability analysis of the algorithms, it is required that eigenvalues are located inside the stability domain to yield convergence behavior. If some eigenvalues are outside the stability domain of explicit methods, numerical stability may not be revealed by dynamic simulation. However, the numerical results can be corrected by treating those outside eigenvalues differently. The decoupled method is based on the idea of separating stiff eigenvalues from the others.

From the geometric viewpoint, the solutions of the ODE and DAE systems are points or vectors in the multidimensional space. This space can be divided into two or more subspaces, and the solution vectors can be decomposed into the corresponding two or more subvectors in each subspace. Thus, by decomposing the space into a number of small subspaces, the solution vectors can be divided into subvectors, and the original ODE and DAE systems can be decoupled into several small dimensional systems.

Denote $\lambda_1, \dots, \lambda_m, \lambda_{m+1}, \dots, \lambda_n$ as the eigenvalues of linearized n -dimensional matrix $A = D_x f(x)$, and suppose only first m eigenvalues are located outside the stability domain of an explicit method. Let \mathbf{P} be the invariant subspace corresponding to these m eigenvalues, and let Z_1 be an orthonormal basis in \mathbf{P} . Thus, Z_1 is an $n \times m$ matrix that satisfies the following conditions:

$$AZ_1 = Z_1\Lambda_1 \quad (9)$$

$$Z_1^T Z_1 = I_m \quad (10)$$

where I_m is an $m \times m$ identity matrix, and Λ_1 is a square matrix with m eigenvalues outside the stability domain.

Furthermore, there exists an orthogonal complement \mathbf{Q} such that $\mathbf{Q} = \mathbf{P}^\perp$, and let Z_2 be the orthonormal basis in \mathbf{Q} ; then

$$Z_2^T Z_2 = I_{n-m} \quad (11)$$

and since \mathbf{Q} is an orthogonal complement, it follows that

$$Z_1^T Z_2 = 0 \quad Z_2^T Z_1 = 0. \quad (12)$$

Therefore, the n -dimensional space can be represented by the direct sum of \mathbf{P} and \mathbf{Q} , where Z_1 and Z_2 are the corresponding basis, respectively. Moreover, $n \times n$ dimension matrix $Z_1 Z_1^T$ and $Z_2 Z_2^T$ are the orthogonal projectors into the two subspaces according to the definition in [25].

Because $Z_1 Z_1^T$ and $Z_2 Z_2^T$ are the orthogonal projectors, any vector in the full space can be projected into two subspaces by multiplying the projectors on the left. In other words, once the projections in these two subspaces are known, the original vector in the full space can be recovered. Let p and q be the vectors in m -dimensional and $(n-m)$ -dimensional subspaces, and the original n -dimensional vector x can be recovered from p and q by setting $p = Z_1^T x$ and $q = Z_2^T x$.

Proposition 1: For each vector x in R^n space, there exist unique vector $p \in R^m$ and $q \in R^{n-m}$ such that $x = Z_1 p + Z_2 q$.

Proof: Omitted. \square

Since the vector in the original n -dimensional space can be decomposed into the sum of two small dimension vectors, the original system can be split into two subsystems according to the Lyapunov–Schmidt decomposition [21], [22]

$$f^P(p, q) = Z_1^T f(Z_1 p + Z_2 q) \quad (13)$$

$$f^Q(p, q) = Z_2^T f(Z_1 p + Z_2 q) \quad (14)$$

and the ODE system equations can be decoupled into two systems

$$\begin{cases} \dot{p} = f^P(p, q) = Z_1^T f(Z_1 p + Z_2 q) \\ \dot{q} = f^Q(p, q) = Z_2^T f(Z_1 p + Z_2 q) \end{cases}. \quad (15)$$

By solving the above equations, variables p and q can be calculated separately, and the original states are given as $x = Z_1 p + Z_2 q$.

For the decoupled systems, the second set of equations $Z_2^T f(Z_1 p + Z_2 q)$ has the derivative $Z_2^T A Z_2$. It is desirable if the eigenvalues are still in the stability region of the explicit methods, and we have the following conclusion.

Proposition 2: The matrix $Z_2^T A Z_2$ has the remaining $n-m$ eigenvalues $\lambda_{m+1}, \dots, \lambda_n$.

Proof is given in [21] and [22]. \square

Equation (15) has the desired form as (8), and all the eigenvalues of the second equation set are inside the stability domain of the explicit method. Therefore, an explicit method can be applied to solve the second set of equations, and an implicit method can be applied to solve the first set of equations. To eliminate the need for Z_2 , let $v = Z_2 q$; thus

$$\dot{v} = Z_2 Z_2^T f(Z_1 p + v) = (I - Z_1 Z_1^T) f(Z_1 p + v).$$

The new system is

$$\begin{cases} \dot{p} = Z_1^T f(Z_1 p + v) \\ \dot{v} = (I - Z_1 Z_1^T) f(Z_1 p + v). \end{cases} \quad (16)$$

IV. DECOUPLED METHOD FOR DAE

The simulation of DAE systems involves solving a set of differential equations and a set of algebraic equations simultaneously. The solutions of DAE equations can be obtained either separately or simultaneously [23]. The decoupled method can be applied to the differential equations in a similar way as applied to ODE systems. To demonstrate the approach, the forward Euler method is chosen as an example of the explicit method and the trapezoidal method as an example of the implicit method. Also, we denote the number of differential equations as n , the number of the algebraic equations as l , and the dimension of stiff invariant subspace \mathbf{P} as m .

A. Decoupled Forward Euler–Trapezoidal Method for DAE

Similar to the ODE system, the DAE system can be decomposed into the following form:

$$\begin{cases} \dot{p} = Z_1^T f(Z_1 p + v) \\ \dot{v} = (I - Z_1 Z_1^T) f(Z_1 p + v) \\ 0 = g(p, v, y). \end{cases} \quad (17)$$

The initial conditions are given as

$$\begin{cases} x(0) = x_0 \\ y(0) = y_0 \\ p(0) = Z_1^T x_0 \\ v(0) = (I - Z_1 Z_1^T) x_0. \end{cases} \quad (18)$$

The decoupled forward Euler–trapezoidal method is formulated as in (19), shown in the equation at the bottom of the page. The first set of equations can be solved via fixed-point iteration, and the second and third sets of the equations actually are nonlinear equation, and the Newton method is needed to solve it. The second and third equation sets are reformulated as in (20), shown at the bottom of the next page, where the unknowns are p_{k+1} and y_{k+1} .

Meanwhile, the full implicit method needs to solve the following nonlinear equation set:

$$\begin{cases} x_{k+1} - hf(x_{k+1})/2 = x_k + hf(x_k)/2 \\ g(x_{k+1}, y_{k+1}) = 0. \end{cases} \quad (21)$$

The dimension of the full implicit method is $n+l$, while the dimension of the decoupled system is $m+l$. Since $m \ll n$, the dimension of the nonlinear systems can be significantly reduced.

$$\begin{cases} v_{k+1} = v_k + h(I - Z_1 Z_1^T) f(Z_1 p_k + v_k) \\ p_{k+1} = p_k + hZ_1 [(f(Z_1 p_k + v_k) + f(Z_1 p_{k+1} + v_{k+1}))]/2 \\ 0 = g(p_{k+1}, v_{k+1}, y_{k+1}) \end{cases} \quad (19)$$

B. Identification of Stiff Invariant Subspace Basis Z_1

To identify the basis Z_1 of invariant subspace \mathbf{P} , it is only required to identify a few eigenvectors of corresponding eigenvalues instead of all of the eigenvectors. As for the forward Euler method, the stability domain boundary is a circle that has the center $(-1/h, 0)$ and radius $1/h$ for eigenvalue λ . Thus, the stiff invariant subspace is associated with eigenvalues outside the circle. Because the algorithms for dominant eigenvalues calculation such as the Arnoldi method tend to converge to eigenvalues with the largest moduli, it is not straightforward to directly identify the invariant subspace if the center of the circle is not the origin. To make the origin as the center of the circle, a linear transformation is performed according to Proposition 3.

Proposition 3: Let (λ_i, v_i) be the eigenpair of n -by- n matrix A ; then, $(\lambda_i + 1/h, v_i)$ is the eigenpair of matrix $A + I/h$.

Proof: Omitted. \square

Now the original problem is transformed into the new problem to find out a few eigenvalues outside a circle of the matrix $A + I/h$. These eigenvalues can be computed efficiently by the Arnoldi method.

V. NUMERICAL EXPERIMENTS

The decoupled forward Euler–trapezoidal method, the full explicit forward Euler method, and the full implicit trapezoidal methods are applied to the New England 39-bus system and the IEEE 118-bus system in the MATLAB environment. The New England system has 39 buses and ten generators, and the IEEE 118-bus system has 118 buses and 48 generators. The generators are represented by the two-axis model as in [23], and the exciter and governor models are the same as in [24]. There are nine differential states for each generator, and the load model consists of 50% constant power, 30% constant current, and 20% constant impedance.

A. New England 39-Bus System Results

The New England system has 39 buses and ten generators. As there are nine states for each generator, the total number of differential states and algebraic states are 90 and 78, respectively. The step size during the simulation is chosen as 0.025 s. The stiff invariant subspace is calculated at the initial state with dimension as 19; thus, the dimension of the nonlinear equation system is 97 for the decoupled method (19 stiff differential states and 78 algebraic states), while the dimensions of the nonlinear equation systems are 78 and 168 for the explicit method and implicit method, respectively. The computational time for stiff invariant subspace is 0.235 s.

In the following section, several disturbances and control actions are considered to demonstrate both the accuracy and computational efficiency of the decoupled method for the New England system. Here is a summary of the cases:

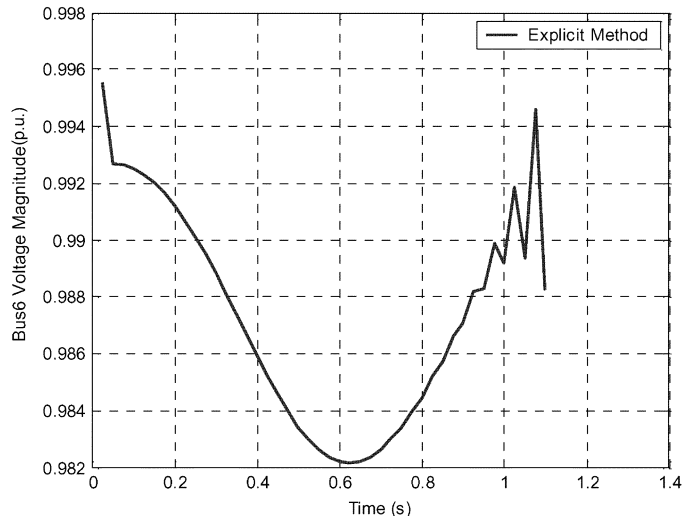


Fig. 3. Case A simulation result by forward Euler method.

- Case A) line outage;
- Case B) line outage with load variation;
- Case C) line outage with load variation and shunt capacitor compensation.

1) *Case A: Time-Domain Simulation With Line Trip Contingency:* The contingency is the transmission line trip between bus 6 and bus 7 at 0.05 s, and the simulation duration is 20 s. The actual post-disturbance behavior is that the system stability can be maintained. The simulation results of the decoupled method and the full implicit method yield stable cases; however, the full explicit method fails to give the correct answer. The full explicit method (forward Euler method) diverges at about 1.1 s, as shown in Fig. 3. Before the explicit method diverges, an oscillatory behavior can be observed from the result, which is only due to numerical error instead of real system response.

The simulation results of the decoupled method and the full implicit method give the stable system behaviors, as shown in Fig. 4. Both methods give stable post disturbance behavior, and the results from two methods match very well.

The computational time of the decoupled method, the explicit method and the full implicit method in case A is shown in Table I (N/A means the dynamical simulation cannot be finished due to numerical divergence under the given step size; the maximum step size for the forward Euler method to give similar results as the trapezoidal method is 0.002 s, and the simulation time is about 2800 s under such step size). It shows that the decoupled method requires much less time than the implicit method to finish the dynamic simulation.

2) *Case B: Time-Domain Simulation with Line Trip and Load Variation:* The disturbances in case B are both loss of transmission line and load increment. The line between bus 6 and bus 7 is tripped off at 0.05 s, as in case A, and the system loads also increase by 10% per second until the end of simulation (5-s

$$\begin{cases} p_{k+1} - \frac{1}{2}hZ_1f(Z_1p_{k+1} + v_{k+1}) = p_k + \frac{1}{2}hZ_1f(Z_1p_k + v_k) \\ g(p_{k+1}, v_{k+1}, y_{k+1}) = 0 \end{cases} \quad (20)$$

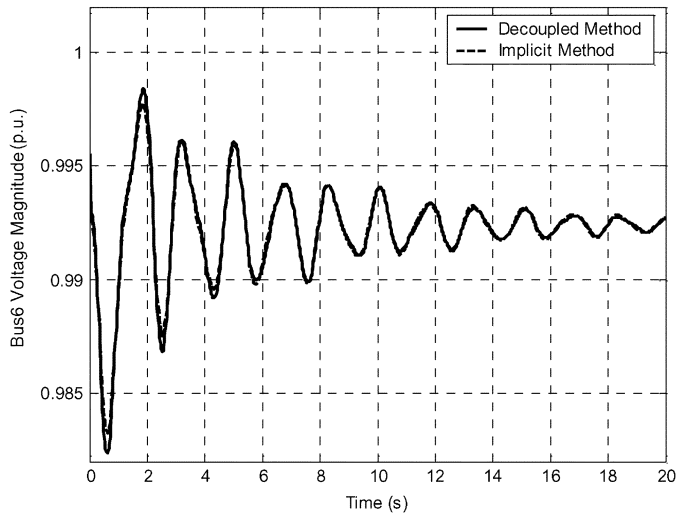


Fig. 4. Case A simulation results by decoupled and implicit methods.

TABLE I
COMPUTATIONAL TIME FOR CASE A

Methods	CPU Time (s)
Explicit Method	N/A
Implicit Method	745
Decoupled Method	405

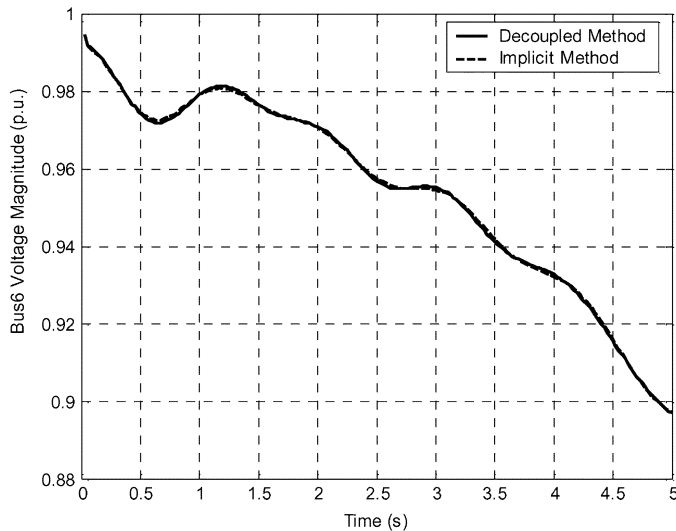


Fig. 5. Case B simulation results by decoupled and implicit methods.

TABLE II
COMPUTATIONAL TIME FOR CASE B

Methods	CPU Time (s)
Implicit Method	230
Decoupled Method	130

duration). The effect of continuous load increment will cause system instability in this case. Here also the explicit method experiences convergence problems and provides the wrong result during the simulation, while the decoupled method and the implicit method yield correct system behaviors, as shown Fig. 5.

The computational time of the decoupled method and the full implicit method in case B is shown in Table II. The decoupled method is almost twice as fast as the implicit method.

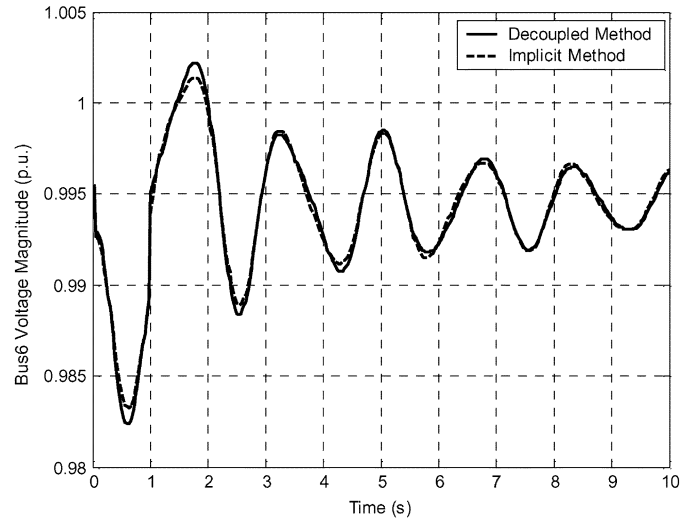


Fig. 6. Case C simulation results by decoupled and implicit methods.

TABLE III
COMPUTATIONAL TIME FOR CASE C

Methods	CPU Time (s)
Implicit Method	442
Decoupled Method	252

TABLE IV
SUMMARY OF NEW ENGLAND SYSTEM RESULTS

Cases	Nonlinear Equations Dimension		Computational Time (s)		Computational Error
	Implicit	Decoupled	Implicit	Decoupled	
Case A	168	97	745	405	9.4689×10^{-4}
Case B	168	97	230	130	9.0860×10^{-4}
Case C	168	97	442	252	12.0000×10^{-4}

3) *Case C: Time-Domain Simulation with Line Trip, Load Variation, and Shunt Compensation:* In case C, the transmission line between bus 6 and bus 7 is tripped at 0.05 s, and system loads increase by 10% in the first second. At 1 s (right after the end of load variation), a 250-KVAR shunt capacitor is switched on at bus 7. The dynamical response of the decoupled method and the implicit method is shown in Fig. 6.

The computational time of the decoupled method and the implicit method in case C is shown in Table III.

The simulation results summary of the above cases are given in Table IV. It includes the dimension of the nonlinear equations for the decoupled and implicit methods, computational time, and the computational errors between the decoupled and implicit methods. The dimension of stiff subspace is 19, and the computational time of the decoupled method includes 0.235 s for the stiff subspace computation. The computational error in Table IV is defined as the infinity norm of the difference between the implicit method and the decoupled method.

B. IEEE 118-Bus System Results

The IEEE 118-bus system has 118 buses and 48 generators, and the total number of differential states and algebraic states are 432 and 236, respectively. The step size during the simulation

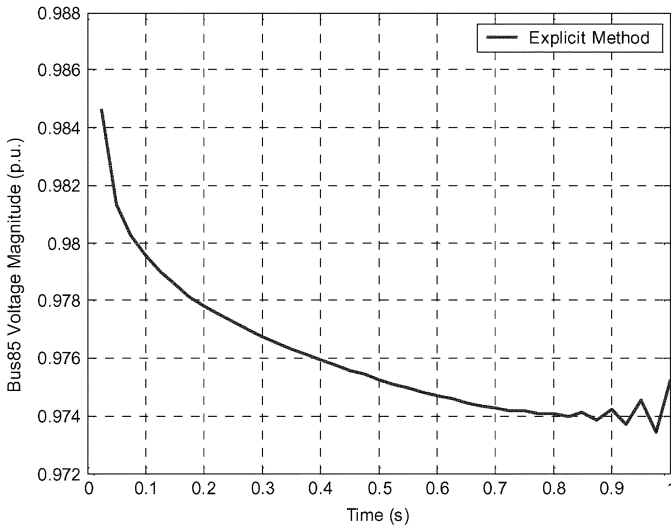


Fig. 7. Case D simulation result by forward Euler method.

is chosen as 0.025 s. The stiff invariant subspace is calculated at the initial state with dimension as 31; thus, the dimension of the nonlinear equation system is 267 for the decoupled method (31 stiff differential states and 236 algebraic states), while the dimensions of the nonlinear equation systems are 236 and 668 for the explicit method and implicit method, respectively. The computational time for stiff invariant subspace is 1.062 s.

Three cases are considered in the IEEE 118-bus system. Here is a summary of the cases.

Case D) *Line outage*. In case D, the transmission line between bus 85 and bus 89 is tripped off at the time of 0.05 s. The simulation duration is 2 s.

Case E) *Line outage with shunt capacitor compensation*. In case E, line 85–89 is tripped off at the same time as in case D; after the line trip contingency, a 250-KVAR shunt is switched on at bus 85 at the time of 0.5 s. The simulation duration is 2 s.

Case F) *Three-phase short circuit*. In case F, a three-phase short-circuit fault occurs at the middle point of line 85–89 at the time of 0.05 s. The fault duration is 0.1 s (6 cycles for 60-Hz system), and the fault is cleared at the time of 0.15 s by opening line 85–89.

The simulation results in case D are plotted in Figs. 7 and 8. In Fig. 7, the results by the explicit method show oscillatory behavior near 1 s, which is only due to numerical error accumulation. The simulation results by both the decoupled method and the implicit method are shown in Fig. 8, and both methods give similar results while the decoupled method requires much less time.

The simulation results in case E and case F are given in Figs. 9 and 10. The decoupled method and the implicit method also give very close results.

The simulation results summary of the above IEEE 118-bus system cases are given in Table V. It includes the dimension of the nonlinear equations for decoupled and implicit methods, computational time, and the computational errors between the decoupled and implicit methods. The dimension of stiff subspace is 31, and the computational time of the decoupled method includes 1.062 s for the stiff subspace computation. The compu-

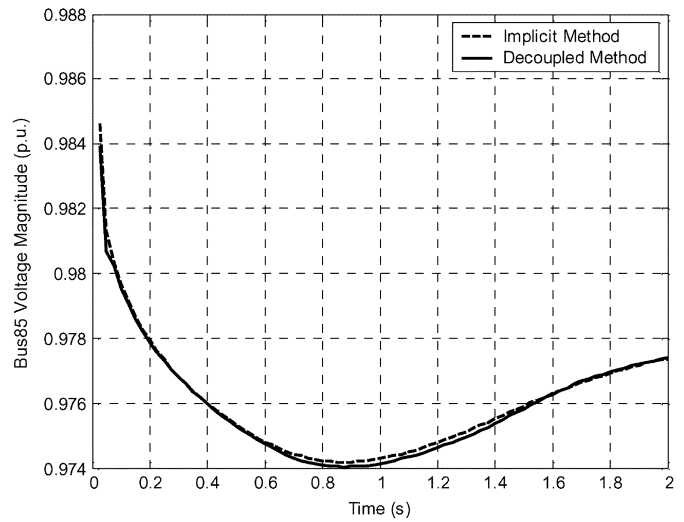


Fig. 8. Case D simulation results by decoupled and implicit methods.

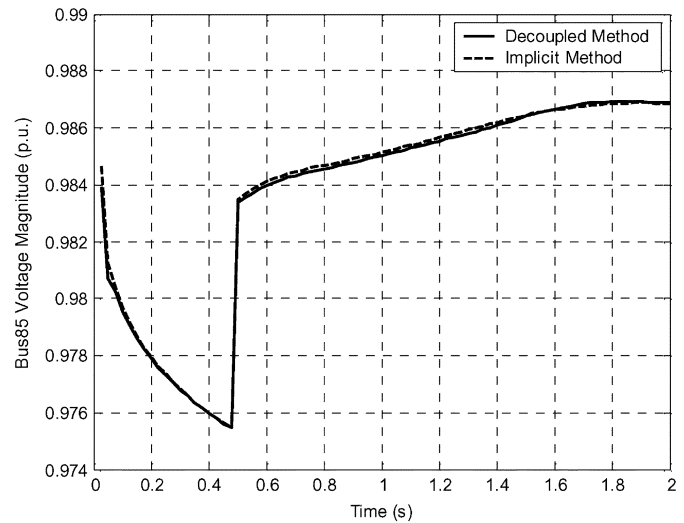


Fig. 9. Case E simulation results by decoupled and implicit methods.

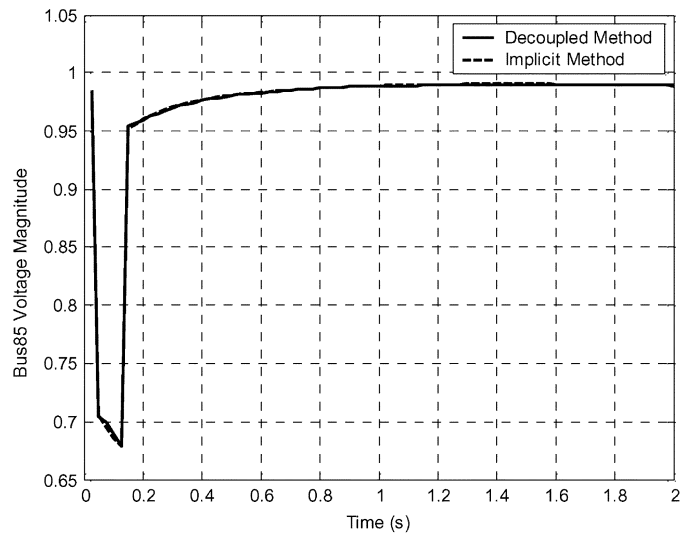


Fig. 10. Case F simulation results by decoupled and implicit methods.

tational error is the infinity norm of the difference between the implicit method and the decoupled method.

TABLE V
SUMMARY OF IEEE 118-BUS SYSTEM RESULTS

Cases	Nonlinear Equations Dimension		Computational Time (s)		Computational Error
	Implicit	Decoupled	Implicit	Decoupled	
Case D	668	267	956	247	6.9335e-004
Case E	668	267	997	246	6.9335e-004
Case F	668	267	998	299	49.0000e-004

VI. CONCLUSION

A decoupled method is presented in this paper for power system dynamical simulation where the stiff and nonstiff subspaces are separated with each other and are treated with different methods. As a result, the decoupled method combines the advantages of pure explicit and pure implicit methods to achieve both numerical stability and computational efficiency.

REFERENCES

- [1] J. Y. Astic, A. Bihain, and M. Jerosolimski, "The mixed Adams-BDF variable step size algorithm to simulate transient and long-term phenomena in power systems," *IEEE Trans. Power Syst.*, vol. 9, no. 2, pp. 929–935, May 1994.
- [2] F. Iavernaro, M. L. Scala, and F. Mazzia, "Boundary values methods for time-domain simulation of power system dynamic behavior," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 45, no. 1, pp. 50–63, Jan. 1998.
- [3] C.-W. Liu and J. S. Thorp, "New methods for computing power system dynamic response for real-time transient stability prediction," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 47, no. 3, pp. 324–337, Mar. 2000.
- [4] H. Fankhauser, K. Aneros, A. Edris, and S. Torseng, "Advanced simulation techniques for the analysis of power system dynamics," *IEEE Comput. Appl. Power*, vol. 3, no. 4, pp. 31–36, Oct. 1990.
- [5] A. Kurita, H. Okubo, K. Oki, S. Agematsu, D. B. Klapper, N. W. Miller, W. W. Price, J. J. Sanchez-Gasca, K. A. Wirgau, and T. D. Younkins, "Multiple time-scale power system dynamic simulation," *IEEE Trans. Power Syst.*, vol. 8, no. 1, pp. 216–223, Feb. 1993.
- [6] M. Stubbe, A. Bihain, J. Deuse, and J. C. Baader, "STAG—A new unified software program for the study of the dynamic behavior of electrical power systems," *IEEE Trans. Power Syst.*, vol. 4, no. 1, pp. 129–138, Feb. 1989.
- [7] Y. Zhou and V. Ajjarapu, "Local parameterization approach for unified time-domain simulation of power system dynamics," in *Proc. IEEE Power Engineering Society Winter Meeting*, 2002, pp. 244–249.
- [8] E. Hofer, "A partially implicit method for large stiff systems of ODE's with only few equations introducing small time-constants," *SIAM J. Numer. Anal.*, vol. 13, pp. 645–663, 1976.
- [9] C. W. Gear and Y. Saad, "Iterative solution of linear equations in ODE codes," *SIAM J. Sci. Stat. Comput.*, vol. 4, pp. 583–601, 1983.
- [10] Y. Saad, "Krylov subspace methods for solving large unsymmetric linear systems," *Math. Comput.*, vol. 37, pp. 105–126, 1981.
- [11] L. Lopez and D. Trigiante, "A projection method for the numerical solution of linear systems in separable stiff differential equations," *Int. J. Comput. Math.*, vol. 30, pp. 191–206, 1989.
- [12] A. Bjorck, "A block QR algorithm for partitioning stiff differential systems," *BIT (Denmark)*, vol. 23, pp. 329–345, 1983.
- [13] W. H. Enright and M. S. Kamel, "Automatic partitioning of stiff systems and exploiting the resulting structure," *ACM Trans. Math. Softw.*, vol. 5, pp. 374–385, 1979.
- [14] D. S. Watkins and R. W. HansonSmith, "The numerical solution of separably stiff systems by precise partitioning," *ACM Trans. Math. Softw.*, vol. 9, pp. 293–301, 1983.
- [15] U. M. Ascher and L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1998.
- [16] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. New York: Springer-Verlag, 1991.
- [17] J. D. Lambert, *Numerical Methods for Ordinary Differential Systems: The Initial Value Problems*. New York: Wiley, 1991.
- [18] A. Iserles, *A First Course in the Numerical Analysis of Differential Equations*. Cambridge, U.K.: Cambridge Univ. Press, 1996.
- [19] J. Deuffhard and F. Bornemann, *Scientific Computing with Ordinary Differential Equations*. New York: Springer-Verlag, 2002.
- [20] J. C. Butcher, "Numerical methods for ordinary differential equations in the 20th century," in *Numerical Analysis: Historical Developments in the 20th Century*, C. Brezinski and L. Wuytack, Eds. New York: Elsevier, 2001.
- [21] V. Janovsky and O. Liberda, "Continuation of invariant subspaces via the recursive projection method," *Appl. Math.*, vol. 48, pp. 241–255, 2003.
- [22] G. M. Shroff and H. B. Keller, "Stabilization of unstable procedures: the recursive projection method," *SIAM J. Numer. Anal.*, vol. 30, pp. 1099–1120, Aug. 1993.
- [23] P. W. Sauer and M. A. Pai, *Power System Dynamics and Stability*. Englewood Cliffs, NJ: Prentice-Hall, 1998.
- [24] Z. Feng, V. Ajjarapu, and B. Long, "Identification of voltage collapse through direct equilibrium tracing," *IEEE Trans. Power Syst.*, vol. 15, no. 1, pp. 342–349, Feb. 2000.
- [25] G. H. Golub and C. Van Loan, *Matrix Computations*. Baltimore, MD: The Johns Hopkins Univ. Press, 1983.

Dan Yang (S'03) received the B.S. and M.S. degrees in electrical engineering from Wuhan University, Wuhan, China, and Tsinghua University, Beijing, China, in 1999 and 2002, respectively. Currently, he is working toward the Ph.D. degree from the Department of Electrical and Computer Engineering and working toward the M.S. degree from the Department of Economics with a minor in the Department of Mathematics at Iowa State University, Ames.

His research interests include dynamical systems, power system stability analysis, control, and optimization.

Venkataramana Ajjarapu (S'86–M'86–SM'91) received the Ph.D. degree in electrical engineering from the University of Waterloo, Waterloo, ON, Canada, in 1986.

Currently, he is an Associate Professor in the Department of Electrical and Computer Engineering at Iowa State University, Ames. His research is in the area of reactive power planning, voltage stability analysis, and nonlinear voltage phenomena.