(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
22 December 2005 (22.12.2005)

PCT

(10) International Publication Number
WO 2005/121840 A1

(51) International Patent Classification[7]: **G01V 01/00**, 01/28, G06F 19/00, 17/10, 07/60, G06G 7/48

(21) International Application Number: PCT/US2005/012629

(22) International Filing Date: 13 April 2005 (13.04.2005)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/577,541 7 June 2004 (07.06.2004) US

(71) Applicant *(for all designated States except US)*: **EXXON-MOBIL UPSTREAM RESEARCH COMPANY** [US/US]; CORP-URC-SW348, P.O. Box 2189, Houston, TX 77252-2189 (US).

(72) Inventor; and
(75) Inventor/Applicant *(for US only)*: **WATTS, James, W., III** [US/US]; 3014 Sunset Boulevard, Houston, TX 77005-2150 (US).

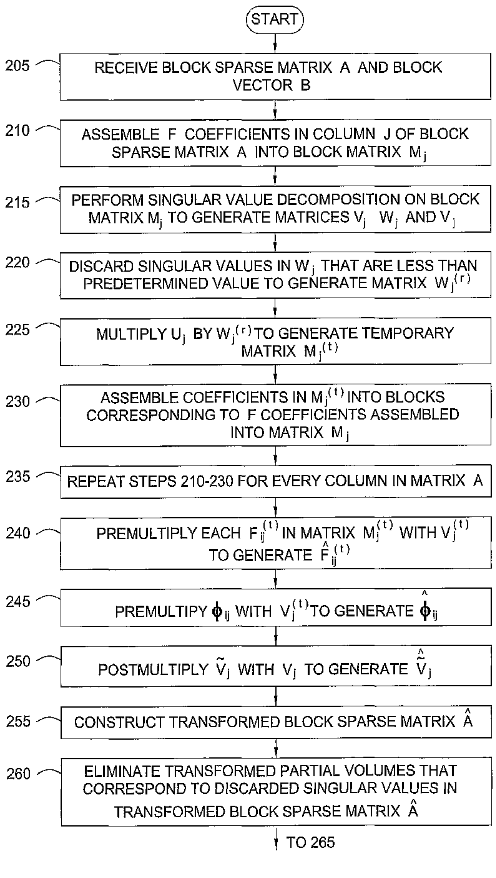
(74) Agents: KNIGHT, Brent, R. et al.; ExxonMobil Upstream Research Company, CORP-URC-SW348, P.O. Box 2189, Houston, TX 77252-2189 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, MC, NL, PL, PT, RO,

[Continued on next page]

(54) Title: METHOD FOR SOLVING IMPLICIT RESERVOIR SIMULATION MATRIX EQUATION



(57) Abstract: A method for solving a matrix equation AX=B, wherein A represents a block sparse matrix (205), B represents a right hand side block vector (205) and X represents a solution block vector (298). In one embodiment, the method includes receiving the block sparse matrix (205) and the right hand side block vector (205), constructing a reduced transformed block vector (205) from the block sparse matrix (205), and the right hand side block vector (205), and solving for the solution block vector (298) using the reduced transformed block sparse matrix (265) and the reduced transformed residual block vector (275).

WO 2005/121840 A1

SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii)) for the following designations AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW, ARIPO patent (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)*

— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii)) for all designations*

— *of inventorship (Rule 4.17(iv)) for US only*

Published:

— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

METHOD FOR SOLVING IMPLICIT RESERVOIR SIMULATION MATRIX EQUATION

BACKGROUND

Field of Inventions

[0001] Embodiments of the present inventions generally relate to exploitation and development of hydrocarbons in an underground reservoir and, more preferably, to an improved process for predicting the behavior of a subterranean, hydrocarbon-bearing formation.

Description of Related Art

[0002] Reservoir simulation is a process of inferring the behavior of a real reservoir from the performance of a model of that reservoir. Because mass transfer and fluid flow processes in petroleum reservoirs are so complex, reservoir simulations are done using computers. Computer programs that perform calculations to simulate reservoirs are called reservoir simulators. The objective of reservoir simulation is to understand the complex chemical, physical, and fluid flow processes occurring in a petroleum reservoir sufficiently well to be able to predict future behavior of a reservoir and to maximize recovery of hydrocarbons. The reservoir simulator can solve reservoir problems that are generally not solvable in any other way. For example, a reservoir simulator can predict the consequences of reservoir management decisions. Reservoir simulation often refers to the hydrodynamics of flow within a reservoir, but in a larger sense it also refers to the total petroleum system which includes the reservoir, the surface facilities, and any interrelated significant activity.

[0003] Figure 1 illustrates schematically four basic steps in one example of a reservoir simulation of a petroleum reservoir. The first step (step 1) is to construct a mathematical model of a real reservoir based on the chemical, physical, and fluid flow processes occurring in the reservoir. That mathematical model may include a set of nonlinear partial differential equations. The second step (step 2) involves discretization of the reservoir in both time and space. Space is discretized by dividing the reservoir into suitable gridcells with each gridcell having a set of nonlinear finite difference equations. The third step (step 3) is to linearize the nonlinear terms that appear in the nonlinear finite difference equations and, based on this linearization, construct linear algebraic equations assembled in a matrix equation. The fourth step (step 4) is to solve the linear algebraic equations assembled in the matrix equation. The simulation proceeds in a series of timesteps, and steps 3 and 4 are performed at each timestep. The simulation provides a prediction of reservoir behavior, which enables a petroleum engineer to predict reservoir performance, including the rate at which the reservoir can be produced. The accuracy of the model can be checked against the history of the reservoir after the model has been subjected to a simulated recovery process.

[0004] However, many simulation methods have been proposed. The method chosen can affect the stability and accuracy of the solution. Some methods require more computational work than other methods on a per-timestep basis. The methods differ primarily on how they treat the way the reservoir variables (such as pressure and saturation) vary in time. Most methods involve variations of the following two procedures:

(1) Explicit procedures use mobilities and capillary pressures computed as functions of saturations at the beginning of a timestep. The saturations are known from the previous timestep calculations. The mobilities and capillary pressures are assumed to maintain the same values during a timestep that they had at the beginning of the timestep.

(2) Implicit procedures use mobility and capillary pressure calculated as functions of saturation at the end of the timestep. The values are not known until calculations for the timestep have been completed. As a result, they must be determined using an iterative process.

[0005] The Fully Implicit method is a commonly used implicit procedure. This method is unconditionally stable because it treats both pressure and saturations implicitly. Flow rates are computed using phase pressures and saturations at the end of each timestep. In this method, saturations cannot fall below zero because a fluid can flow only if it is mobile at the end of a timestep. Fluids are mobile only for saturations greater than zero. The calculation of flow rates, pressure and saturation solutions involves the solution of nonlinear equations using a suitable iterative technique. Once the pressures and saturations are solved, these terms will continue to be updated using the new values of pressure and saturation. The iteration process terminates when the convergence criteria are satisfied.

[0006] The main drawback of the Fully Implicit method is the amount of computer time that it requires. In terms of computing cost, the method is generally satisfactory in models of single wells or parts of a reservoir, but it can be quite expensive to use in models of entire reservoirs. Several attempts have been made to reduce the computations required, possibly at the cost of accepting a method that does not permit the timestep sizes of the Fully Implicit method. The sequential implicit method, the adaptive implicit method, and the Cascade method have been proposed as ways of reducing the computational time. However, those methods have their own drawbacks. The largest consumer of computational time in the Fully Implicit method is the equation solving step (Step 4 of Figure 1). This typically consumes about three-fourths of the total computational time.

[0007] Accordingly, a need exists for a more computationally efficient method for solving the linear algebraic equations arising in Fully Implicit reservoir simulation.

SUMMARY

[0008] Various embodiments of the invention are directed to a method for solving a matrix equation AX=B, wherein A represents a block sparse matrix, B represents a right hand side block vector and X represents a solution block vector. In one embodiment, the method includes receiving the block sparse matrix and the right hand side block vector, constructing a reduced transformed block sparse matrix from the block sparse matrix, constructing a reduced transformed residual block vector from the block sparse matrix and the right hand side block vector, and solving for the solution block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector.

[0009] In another embodiment, the method includes constructing a reduced transformed block sparse matrix from the block sparse matrix, constructing a reduced transformed residual block vector from the block sparse matrix and the right hand side block vector, solving for a reduced transformed solution change block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector, converting the reduced transformed solution change block vector to a solution change block vector having one or more changes in mass unknowns and one or more changes in pressure unknowns, and adding the solution change block vector to a current estimate of the solution block vector to update the solution block vector.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] Figure 1 illustrates a schematic diagram of the basic steps in an illustrative reservoir simulation process.

[0011] Figure 2 illustrates a method for solving one or more linear algebraic equations in a matrix in accordance with one or more embodiments of the invention.

[0012] Figure 3 illustrates a computer network into which one or more embodiments of the invention may be implemented.

DETAILED DESCRIPTION

Introduction and Definitions

[0013] A detailed description will now be provided. Each of the appended claims defines a separate invention, which for infringement purposes is recognized as including equivalents to the various elements or limitations specified in the claims. Depending on the context, all references below to the "invention" may in some cases refer to certain specific embodiments only. In other cases it will be recognized that references to the "invention" will refer to subject matter recited in one or more, but not necessarily all, of the claims. Each of the inventions will now be described in greater detail below, including specific embodiments, versions and examples, but the inventions are not limited to these embodiments, versions or examples, which are included to enable a person having ordinary skill in the art to make and use the inventions, when the information in this patent is combined with available information and technology. Various terms as used herein are defined below. To the extent a term used in a claim is not defined below, it should be given the broadest definition persons in the pertinent art have given that term as reflected in one or more printed publications or issued patents.

[0014] As used herein, the term "gridcell" is defined as a unit or block that defines a portion of a three dimensional reservoir model. As such, a three dimensional reservoir model may include a number of gridcells, ranging from tens and hundreds to thousands and millions of gridcells. Each gridcell may in certain cases represent a specifically allocated portion of the three dimensional reservoir model. An entire set of gridcells may constitute a geologic model that represents a subsurface earth volume of interest. Each gridcell preferably represents a unique portion of the subsurface. Such gridcells preferably do not overlap each other. Dimensions of the gridcells are preferably chosen so that the reservoir properties within a gridcell are relatively homogeneous, yet without creating an excessive number of gridcells. These gridcells have sides ranging from a smaller than a meter to a few hundred meters. Preferably, each gridcell is square or rectangular in plan view and has a thickness that is either constant or variable. However, it is contemplated that other shapes may alternatively be used. Gridcells may be visualized as well-stirred tanks with permeable sides. The contents of a gridcell, therefore, may be considered uniformly distributed within the gridcell and the rates at which fluids flow in or out maybe determined by the permeabilities of the sides of the gridcell and the pressure differences between adjacent gridcells. As such, the mathematical problem is reduced to a calculation of flow between adjacent gridcells.

[0015] As used herein, the term "singular value decomposition" is defined as a mathematical technique for decomposing a rectangular matrix into three factors. Given an M by N matrix A, where M > N, using at least certain types of singular value decomposition, matrix A can be rewritten as:

$$\begin{bmatrix} & & \\ & A & \\ & & \end{bmatrix} = \begin{bmatrix} & & \\ & U & \\ & & \end{bmatrix} \begin{bmatrix} W \end{bmatrix} \begin{bmatrix} V^T \end{bmatrix}$$

where W is a diagonal matrix having "singular values" entries. The columns of matrix U and the rows of matrix V^T (or, equivalently, columns of matrix V) are orthonormal. That is, for example, if U_i is a column of matrix U and U_j is another column of matrix U, then

$$U_i^T U_i = 1$$
$$U_i^T U_j = 0, j \neq i$$

[0016] The equivalent is true for the square matrix V, i.e., the inverse of matrix V is matrix V^T. The columns of matrix U, diagonal entries of matrix W, and rows of matrix V^T are customarily arranged such that $w_1 > w_2 > \ldots > w_N$, where w_i is the diagonal entry in row i of matrix W. The diagonal elements of matrix W are the singular values of matrix A. For purposes of illustrating various embodiments of the invention, it is more convenient to place them in the opposite order, such that $w_1 < w_2 < \ldots < w_N$, and reorder the columns of U and V accordingly. Matrix V may be referred to as the right matrix and the vectors comprised therein as the right singular vectors.

[0017] Given the above definitions, matrix A can be approximated using a subset containing the last few columns of matrix U, entries of matrix W, and rows of matrix V^T. Thus, matrix A can be rewritten as:

$$\begin{bmatrix} & & \\ & A & \\ & & \end{bmatrix} \approx \begin{bmatrix} \\ U_S \\ \\ \end{bmatrix} \begin{bmatrix} W_S \end{bmatrix} \begin{bmatrix} V_S^T \end{bmatrix}.$$

[0018] As used herein, the term "timestep" is defined as an increment of time into which the life of a reservoir is discretized. For at least certain types of timesteps, a reservoir simulator computes changes in each gridcell (flow, pressure, etc.) over a timestep for many timesteps. Typically, conditions are defined only at the beginning and end of a timestep, and nothing is defined at any intermediate time within a timestep. Consequently, conditions within each gridcell may change abruptly from one timestep to the next. Usually, timesteps are chosen to be small enough to limit sizes of these abrupt changes to acceptable limits. The size of the timesteps depends on accuracy considerations and stability constraints. Generally, the smaller the timestep, the more accurate the solution, however, smaller timesteps require more computational work.

[0019] As used herein, the term "identity matrix" is defined as a square matrix of any dimension whose elements are ones on its northwest-to-southeast diagonal and zeroes everywhere else. Any square matrix multiplied by the identity matrix with those dimensions equals itself.

[0020] As used herein, the term "volume constraint" is an equation based on the principle that a gridcell must contain the amount of fluid required to fill the gridcell at a given time. For example, if a reservoir contains liquid hydrocarbon and water phases, then:

Liquid Hydrocarbon Volume + Water Volume = Gridcell Volume.

[0021] As used herein, the term "unknown" is defined as an unknown variable for which the linear algebraic equations assembled in the matrix equation are solved. For a reservoir containing oil and water, the linear algebraic equations are solved for various unknown variables, including pressure and mass. Other quantities may be derived from these variables.

[0022] As used herein, the term "block sparse matrix" is defined as a matrix whose elements are mostly null or zeroes and submatrices as the remainder.

However, all of the submatrices along its diagonal are present, i.e., not null. For example, a block sparse matrix A may be expressed as:

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{bmatrix}.$$

[0023] Each submatrix, e.g., A_{ij}, may be expressed as:

$$A_{ij} = \begin{bmatrix} a_{11}^{(ij)} & a_{12}^{(ij)} & \cdots & a_{1,m_j+1}^{(ij)} \\ a_{21}^{(ij)} & a_{22}^{(ij)} & \cdots & a_{2,m_j+1}^{(ij)} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m_i+1,1}^{(ij)} & a_{m_i+1,2}^{(ij)} & \cdots & a_{m_i+1,m_j+1}^{(ij)} \end{bmatrix}$$

where m_i is the number of mass balance equations in row i and m_j is the number of mass change unknowns in column j. Each submatrix has $m_i + 1$ rows and $m_j + 1$ columns. The first m_i rows contain coefficients of the mass balance equations. The bottom row (row $m_i + 1$) contains coefficients relating to the volume constraint equation. The coefficients in the first m_j columns are configured to be multiplied by mass variables and the coefficients in the right-most column (column $m_j + 1$) are configured to be multiplied by pressure variables. As such, each submatrix has a particular structure.

[0024] The mass variables can take several forms. Generally, the mass variables are changes over a Newton iteration. But, they can also be changes over a timestep or end-of-timestep masses, and not changes. They also can be expressed in terms of other measures of masses, such as saturations or mole/mass fractions. Similarly, pressure variables are generally changes over a Newton iteration, but can also be end-of-timestep pressures. These variable choices are interchangeable in the sense that one set of variables can be easily converted to another. For simplicity, the description that follows uses end-of-timestep masses and pressures.

[0025] For an off diagonal submatrix, where $i \neq j$, A_{ij} may be expressed as:

$$A_{ij} = \begin{bmatrix} f_{11}^{(ij)} & f_{12}^{(ij)} & \dots & f_{1,m_j}^{(ij)} & \varphi_1^{(ij)} \\ f_{21}^{(ij)} & f_{22}^{(ij)} & \dots & f_{2,m_j}^{(ij)} & \varphi_2^{(ij)} \\ \dots & \dots & \dots & \dots & \dots \\ f_{m_i,1}^{(ij)} & f_{m_i,2}^{(ij)} & \dots & f_{m_i,m_j}^{(ij)} & \varphi_{m_i}^{(ij)} \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix}$$

where f and φ coefficients relate to the flows between gridblocks. The f coefficients, which occupy the first m_i columns of A_{ij}, are configured to be multiplied by mass changes. Thus, they may be referred to as mass change terms coefficients. The φ coefficients, which occupy the column m_{i+1} of A_{ij}, are configured to be multiplied by pressure changes. Thus, the φ coefficients may be referred to as pressure change terms coefficients. The f coefficients are used in the singular value decomposition, while the φ coefficients are not.

A_{ij} may also be expressed as:

$$A_{ij} = \begin{bmatrix} F_{ij} & \Phi_{ij} \\ 0 & 0 \end{bmatrix}$$

where $F_{ij} = \begin{bmatrix} f_{11}^{(ij)} & f_{12}^{(ij)} & \dots & f_{1,m_j}^{(ij)} \\ f_{21}^{(ij)} & f_{22}^{(ij)} & \dots & f_{2,m_j}^{(ij)} \\ \dots & \dots & \dots & \dots \\ f_{m_i,1}^{(ij)} & f_{m_i,2}^{(ij)} & \dots & f_{m_i,m_j}^{(ij)} \end{bmatrix}$ and $\Phi_{ij} = \begin{bmatrix} \phi_1^{(ij)} \\ \phi_2^{(ij)} \\ \dots \\ \phi_{m_i}^{(ij)} \end{bmatrix}$.

[0026] The diagonal submatrix A_{ii} may be expressed as

$$A_{ii} = \begin{bmatrix} 1+f_{11}^{(ii)} & f_{12}^{(ii)} & \dots & f_{1,m_i}^{(ii)} & \phi_1^{(ii)} \\ f_{21}^{(ii)} & 1+f_{22}^{(ii)} & \dots & f_{2,m_i}^{(ii)} & \phi_2^{(ii)} \\ \dots & \dots & \dots & \dots & \dots \\ f_{m_i,1}^{(ii)} & f_{m_i,2}^{(ii)} & \dots & 1+f_{m_i,m_i}^{(ii)} & \phi_{m_i}^{(ii)} \\ \widetilde{V}_1^{(i)} & \widetilde{V}_2^{(i)} & \dots & \widetilde{V}_{m_i}^{(i)} & c_i \end{bmatrix}$$

where f and φ coefficients relate to the flows between gridblocks, $\widetilde{V}$'s are fluid partial volumes, and the c relates to compressibility.

A_{ii} may also be expressed as:

$$A_{ii} = \begin{bmatrix} I + F_{ii} & \Phi_{ii} \\ \widetilde{V}_i & c_i \end{bmatrix}$$

where $F_{ii} = \begin{bmatrix} f_{11}^{(ii)} & f_{12}^{(ii)} & \dots & f_{1,mi}^{(ii)} \\ f_{21}^{(ii)} & f_{22}^{(ii)} & \dots & f_{2,m_i}^{(ii)} \\ \dots & \dots & \dots & \dots \\ f_{m_i,1}^{(ii)} & f_{m_i,2}^{(ii)} & \dots & f_{m_i,m_i}^{(ii)} \end{bmatrix}$, $\Phi_{ii} = \begin{bmatrix} \phi_1^{(ii)} \\ \phi_2^{(ii)} \\ \dots \\ \phi_{m_i}^{(ii)} \end{bmatrix}$, $\widetilde{V}_j = \begin{bmatrix} \widetilde{V}_1^{(i)} & \widetilde{V}_2^{(i)} & \dots & \widetilde{V}_{m_j}^{(i)} \end{bmatrix}$ and I is an identity matrix. The entries of $\widetilde{V}_j$ are multiphase partial volumes. They relate the fluid volume in the gridcell to the amounts of the components in it. The c_i term is related to fluid compressibility. It relates the gridcell's fluid volume to its pressure.

[0027] As used herein, the term "column matrix" is defined as a matrix containing the submatrices in a column of the block sparse matrix. For example, the j^{th} column matrix of A is

$$A_j = \begin{bmatrix} A_{1j} \\ A_{2j} \\ \dots \\ A_{n1,j} \end{bmatrix}$$

[0028] As used herein, the term "solution block vector" is defined as a block vector comprised of subvectors $X_1 \dots X_n$. For example, a solution block vector X may be expressed as:

$$X = \begin{bmatrix} X_1 \\ X_2 \\ \dots \\ X_n \end{bmatrix}.$$

Each subvector may be expressed as: $X_i = \begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \\ \\ x_{m_i}^{(i)} \\ x_{m_i+1}^{(i)} \end{bmatrix}$ or $X_i = \begin{bmatrix} m_1^{(i)} \\ m_2^{(i)} \\ \ldots \\ m_{m_i}^{(i)} \\ p^{(i)} \end{bmatrix}$ or $X_i = \begin{bmatrix} M_i \\ P_i \end{bmatrix}$

where m or M represents a mass unknown and p or P represents a pressure unknown.

[0029] In some instances, a solution change vector δX may be of interest, as opposed to the solution block vector X. The solution change vector δX may be expressed as:

$$\delta X_i = \begin{bmatrix} \delta x_1^{(i)} \\ \delta x_2^{(i)} \\ \\ \delta x_{m_i}^{(i)} \\ \delta x_{m_i+1}^{(i)} \end{bmatrix} \text{ or } \delta X_i = \begin{bmatrix} \delta m_1^{(i)} \\ \delta m_2^{(i)} \\ \ldots \\ \delta m_{m_i}^{(i)} \\ \delta p^{(i)} \end{bmatrix} \text{ or } \delta X_i = \begin{bmatrix} \delta M_i \\ \delta P_i \end{bmatrix}.$$

[0030] As used herein, the term "right hand side block vector" is defined as the right hand side of the equation $AX = B$. For example, a subvector of the right hand side block vector B may be expressed as:

$$B_i = \begin{bmatrix} b_1^{(i)} \\ b_2^{(i)} \\ \ldots \\ b_{m_i}^{(i)} \\ b_{m_i+1}^{(i)} \end{bmatrix} \text{ or } B_i = \begin{bmatrix} b_{M,1}^{(i)} \\ b_{M,2}^{(i)} \\ \ldots \\ b_{M,m_i}^{(i)} \\ b_V^{(i)} \end{bmatrix} \text{ or } B_i = \begin{bmatrix} B_{M,i} \\ B_{V,i} \end{bmatrix}$$

where the M subscript denotes a mass balance right-hand side, and a V subscript denotes a volume constraint right-hand side.

[0031] As used herein, the term "residual block vector" is defined by $R = B - Ax$. For example, a subvector of the residual block vector R may be expressed as:

$$R_i = \begin{bmatrix} r_1^{(i)} \\ r_2^{(i)} \\ \ldots \\ r_{m_i}^{(i)} \\ r_{m_i+1}^{(i)} \end{bmatrix} \text{ or } R_i = \begin{bmatrix} r_{M,1}^{(i)} \\ r_{M,2}^{(i)} \\ \ldots \\ r_{M,m_i}^{(i)} \\ r_V^{(i)} \end{bmatrix} \text{ or } R_i = \begin{bmatrix} R_{M,i} \\ R_{V,i} \end{bmatrix}$$

where the M subscript denotes a mass balance residual, and a V subscript denotes a volume constraint residual.

[0032] As used herein, the term "mass balance equation" is defined as a mathematical relationship between the contents of a gridcell and flow into and out of the gridcell. It is based on the assumption that material is neither generated nor lost from the system. Each chemical component in the reservoir fluids must satisfy mass balance in each gridcell. For example, for a methane component, the mass of methane in a particular gridcell at the end of a predetermined timestep must satisfy the following equation:

Mass of Methane at New Time = Mass of Methane at Old Time + Mass of Methane Flow In – Mass of Methane Flow Out

A similar relationship applies for any other chemical component present in the reservoir fluids.

[0033] The mass balance equations for the entire reservoir model may be written as:

$$IM + F_M M + F_P P = B_M \quad (1)$$

[0034] The F_M and F_P matrices relate to flow between gridcells. M contains the mass subvectors, P is the vector of pressures, and B_M contains the mass balance right hand side subvectors. If a one-dimensional model having five gridcells is assumed, matrix F_M can be written as a block tridiagonal matrix as follows:

$$F_m = \begin{bmatrix} F_{m11} & F_{m12} & & & \\ F_{m21} & F_{m22} & F_{m23} & & \\ & F_{m32} & F_{m33} & F_{m34} & \\ & & F_{m43} & F_{m44} & F_{m45} \\ & & & F_{m54} & F_{m55} \end{bmatrix}$$

where, in general,

$$F_{Mjj} = -\sum_i F_{Mij}$$

F_P has the same structural form as F_m. We can write Eq. (1) as

$$\begin{bmatrix} M_1 \\ M_2 \\ M_3 \\ M_4 \\ M_5 \end{bmatrix} + \begin{bmatrix} F_{M11} & F_{M12} & & & \\ F_{M21} & F_{M22} & F_{M23} & & \\ & F_{M32} & F_{M33} & F_{M34} & \\ & & F_{M43} & F_{M44} & F_{M45} \\ & & & F_{M54} & F_{M55} \end{bmatrix} \begin{bmatrix} M_1 \\ M_2 \\ M_3 \\ M_4 \\ M_5 \end{bmatrix} + F_P P = B_M$$

[0035] As used herein, the term "Frobenius norm" is defined as the square root of the sum of the squares of the coefficients in a matrix. For example, the Frobenius norm of the matrix, which is written as:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

may be expressed as:

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}^2} \,.$$

The Frobenius norm provides a useful estimate of the largest singular value in the singular value decomposition of the matrix A.

Specific Embodiments

[0036] Various specific embodiments are described below, at least some of which are also recited in the claims.

[0037] In at least one specific embodiment, a method for solving a matrix equation AX=B, wherein A represents a block sparse matrix, B represents a right hand side block vector and X represents a solution block vector, includes: receiving the block sparse matrix and the right hand side block vector; constructing a reduced transformed block sparse matrix from the block sparse matrix; constructing a reduced transformed residual block vector from the block sparse matrix and the right hand side block vector; and solving for the solution block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector.

[0038] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix comprises assembling the mass change terms coefficients from a column of the block sparse matrix into a column matrix.

[0039] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix includes assembling the mass change terms coefficients from a column of the block sparse matrix into a column matrix; and performing a singular value decomposition on the column matrix to generate a left matrix, a diagonal matrix and a right matrix.

[0040] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix includes: assembling the mass change terms coefficients from a column of the block sparse matrix into a column matrix; and performing a singular value decomposition on the column matrix to generate a left matrix, a diagonal matrix and a right matrix, wherein the diagonal matrix comprises one or more singular values arranged in ascending order.

[0041] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix includes: assembling the mass change terms coefficients from a column of the block sparse matrix into a column matrix; performing a singular value decomposition on the column matrix to generate a left matrix, a diagonal matrix and a right matrix; and discarding one or more singular values that are less than a predetermined threshold value to generate a reduced diagonal matrix. The predetermined threshold value is typically set to a quantity between 0.01 and 0.1, but other values can be used.

[0042] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix includes: assembling the mass change terms coefficients from a column of the block sparse matrix into a column matrix; performing a singular value decomposition on the column matrix to generate a left matrix, a diagonal matrix and a right matrix; discarding one or more singular values that are less than a predetermined threshold value to generate a reduced diagonal matrix; and multiplying the left matrix with the reduced diagonal matrix to generate a temporary column matrix.

[0043] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix includes: assembling the mass change terms coefficients from a column of the block sparse matrix into a column matrix; performing a singular value decomposition on the column matrix to generate a left matrix, a diagonal matrix and a right matrix; discarding one or more singular values that are less than a predetermined threshold value to generate a reduced diagonal matrix; multiplying the left matrix with the reduced diagonal matrix to generate a temporary column matrix having a plurality of mass change terms coefficients; and assembling the mass change terms coefficients of the temporary column matrix into one or more temporary mass change terms coefficient submatrices that correspond to the mass change terms coefficients of the column matrix.

[0044] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix includes: assembling the mass change terms coefficients from a column of the block sparse matrix into a column matrix; performing a singular value decomposition on the column matrix to generate a left matrix, a diagonal matrix and a right matrix; discarding one or more singular values that are less than a predetermined threshold value to generate a reduced diagonal matrix; multiplying the left matrix with the reduced diagonal matrix to generate a temporary column matrix having a plurality of mass change terms coefficients; assembling the mass change terms coefficients of the temporary column matrix into one or more temporary mass change terms coefficient submatrices that correspond to the mass change terms coefficients of the column matrix; and premultiplying each temporary mass change terms coefficient submatrix with the transpose of the right matrix to generate a transformed temporary mass change terms coefficient submatrix.

[0045] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix comprises: assembling the mass change terms coefficients from a column of the block sparse matrix into a column matrix; performing a singular value decomposition on the column matrix to generate a left matrix, a diagonal matrix and a right matrix; discarding one or more singular values that are less than a predetermined threshold value to generate a reduced diagonal matrix; multiplying the left matrix with the reduced diagonal matrix to generate a temporary column matrix having a plurality of mass change terms coefficients; assembling the mass change terms coefficients of the temporary column matrix into one or more temporary mass change terms coefficient submatrices that correspond to the mass change terms coefficients of the column matrix; premultiplying each temporary mass change terms coefficient submatrix with the transpose of the right matrix to generate a transformed temporary mass change terms coefficient submatrix; and premultiplying each subvector containing the pressure change terms coefficients in the block sparse matrix by the transpose of the right matrix to generate a transformed pressure change terms coefficients subvector.

[0046] In a specific embodiment of the method identified above, or of a method described elsewhere herein, constructing the reduced transformed block sparse matrix further comprises constructing a transformed block sparse matrix having the same block structure and submatrix form as the block sparse matrix.

[0047] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix includes: assembling the mass change terms coefficients from a column of the block sparse matrix into a column matrix; performing a singular value decomposition on the column matrix to generate a left matrix, a diagonal matrix and a right matrix; discarding one or more singular values that are less than a predetermined threshold value to generate a reduced diagonal matrix; multiplying the left matrix with the reduced diagonal matrix to generate a temporary column matrix having a plurality of mass change terms coefficients; assembling the mass change terms coefficients of the temporary column matrix into one or more temporary mass change terms coefficient submatrices that correspond to the mass change terms coefficients of the column matrix; premultiplying each temporary mass change terms coefficient submatrix with the transpose of the right matrix to generate a transformed temporary mass change terms coefficient submatrix; premultiplying each subvector containing the pressure change terms coefficients in the block sparse matrix by the transpose of the right matrix to generate a transformed pressure change terms coefficients subvector; and postmultiplying each fluid partial volumes subvector in the block sparse matrix by the right matrix to generate a transformed fluid partial volumes vector.

[0048] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix includes: assembling the mass change terms coefficients from a column of the block sparse matrix into a column matrix; performing a singular value decomposition on the column matrix to generate a left matrix, a diagonal matrix and a right matrix;

discarding one or more singular values that are less than a predetermined threshold value to generate a reduced diagonal matrix; multiplying the left matrix with the reduced diagonal matrix to generate a temporary column matrix having a plurality of mass change terms coefficients; assembling the mass change terms coefficients of the temporary column matrix into one or more temporary mass change terms coefficient submatrices that correspond to the mass change terms coefficients of the column matrix; premultiplying each temporary mass change terms coefficient submatrix with the transpose of the right matrix to generate a transformed temporary mass change terms coefficient submatrix; premultiplying each subvector containing the pressure change terms coefficients in the block sparse matrix by the transpose of the right matrix to generate a transformed pressure change terms coefficients subvector; postmultiplying each fluid partial volumes subvector in the block sparse matrix by the right matrix to generate a transformed fluid partial volumes vector; and constructing a transformed block sparse matrix from one or more of the transformed temporary mass change terms coefficient submatrix, the transformed pressure change terms coefficients subvector and the transformed fluid partial volumes vector.

[0049] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix includes: assembling the mass change terms coefficients from a column of the block sparse matrix into a column matrix; performing a singular value decomposition on the column matrix to generate a left matrix, a diagonal matrix and a right matrix; discarding one or more singular values that are less than a predetermined threshold value to generate a reduced diagonal matrix; multiplying the left matrix with the reduced diagonal matrix to generate a temporary column matrix having a plurality of mass change terms coefficients; assembling the mass change terms coefficients of the temporary column matrix into one or more temporary mass change terms coefficient submatrices that correspond to the mass change terms coefficients of the column matrix; premultiplying each temporary mass change terms coefficient submatrix with the transpose of the right matrix to generate a transformed temporary mass change terms coefficient submatrix; premultiplying each subvector containing the pressure change terms coefficients in the block sparse matrix by the transpose of the right matrix to generate a transformed pressure change terms coefficients subvector; postmultiplying each fluid partial volumes subvector in the block sparse matrix by the right matrix to generate a transformed fluid partial volumes vector; constructing a transformed block sparse matrix from one or more of the transformed temporary mass change terms coefficient submatrix, the transformed pressure change terms coefficients subvector and the transformed fluid partial volumes vector; and eliminating one or more transformed fluid partial volumes that correspond to the discarded singular values in the transformed block sparse matrix to generate the reduced transformed block sparse matrix.

[0050] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the Frobenius norm of the column matrix is computed, which is referred to as the column matrix norm or value. If the column matrix value is equal to or less than a predetermined threshold value, the singular value decomposition is not performed on the column matrix, the singular values are set to zero, the right matrix is set equal to the identity matrix. As a result, certain computations that use the right matrix are skipped or simplified because the right matrix is equal to the identity matrix.

[0051] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the reduced transformed block sparse matrix includes one or more reduced transformed diagonal submatrices and one or more reduced transformed off-diagonal submatrices, wherein each reduced transformed diagonal submatrix includes mass change terms coefficients and pressure change terms coefficients only within the bottom r_i+1 rows and right most r_i+1 columns of each transformed diagonal submatrix.

[0052] In a specific embodiment of the method identified above, or of a method described elsewhere herein, the reduced transformed block sparse matrix includes one or more reduced transformed diagonal submatrices and one or more reduced transformed off-diagonal submatrices, wherein each reduced transformed diagonal submatrix includes mass change terms coefficients and pressure change terms coefficients only within the bottom r_i+1 rows and right most r_i+1 columns of each transformed diagonal submatrix and each reduced transformed off-diagonal submatrix includes mass change terms coefficients and pressure change terms coefficients only within the bottom r_i+1 rows and the right most r_j+1 columns of each transformed off-diagonal submatrix.

[0053] In a specific embodiment of the method identified above, or of a method described elsewhere herein, constructing the reduced transformed residual block vector includes constructing a transformed residual block vector.

[0054] In a specific embodiment of the method identified above, or of a method described elsewhere herein, constructing the transformed residual block vector includes constructing a transformed residual block vector having a transformed mass balance residual subvector and a transformed volume constraint residual subvector.

[0055] In a specific embodiment of the method identified above, or of a method described elsewhere herein, constructing the reduced transformed residual block vector includes premultiplying a mass balance residual subvector by the transpose of the right matrix to generate a transformed mass balance residual subvector.

[0056] In a specific embodiment of the method identified above, or of a method described elsewhere herein, solving for the solution block vector includes solving for a reduced transformed solution change block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector.

[0057] In a specific embodiment of the method identified above, or of a method described elsewhere herein, solving for the solution block vector includes: solving for a reduced transformed solution change block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector; and converting the reduced transformed solution change block vector to a solution change block vector.

[0058] In a specific embodiment of the method identified above, or of a method described elsewhere herein, solving for the solution block vector includes: solving for a reduced transformed solution change block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector; converting the reduced transformed solution change block vector to a solution change block vector; and adding the solution change block vector to a current estimate of the solution block vector to update the solution block vector.

[0059] In a specific embodiment of the method identified above, or of a method described elsewhere herein, solving for the solution block vector includes: solving for a reduced transformed solution change block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector; and converting the reduced transformed solution change block vector to a solution change block vector having one or more changes in mass unknowns and one or more changes in pressure unknowns.

[0060] In at least one specific embodiment, a method for solving a matrix equation AX=B, wherein A represents a block sparse matrix, B represents a right hand side block vector and X represents a solution block vector, the method includes: constructing a reduced transformed block sparse matrix from the block sparse matrix; constructing a reduced transformed residual block vector from the block sparse matrix and the right hand side block vector; solving for a reduced transformed solution change block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector; converting the reduced transformed solution change block vector to a solution change block vector having one or more changes in mass unknowns and one or more changes in pressure unknowns; and adding the solution change block vector to a current estimate of the solution block vector to update the solution block vector.

Specific Embodiments In Drawings

[0061] Specific embodiments shown in the drawings will now be described.

[0062] Figure 2 illustrates a flow diagram of a method 200 for solving one or more linear algebraic equations in a matrix equation in accordance with one embodiment of the invention. At step 205, a block sparse matrix A and a block vector B are received. The block sparse matrix A and the block vector B are used for solving for a solution block vector X according to the implicit transport matrix equation $AX = B$.

[0063] At step 210, the f coefficients in column j of the block sparse matrix A are assembled into a block matrix M_j. The f coefficients that are assembled into the block matrix M_j may be expressed as:

$$M_j = \begin{bmatrix} f_{11}^{(jj)} & f_{12}^{(jj)} & \cdots & f_{1,m_j}^{(jj)} \\ f_{21}^{(jj)} & f_{22}^{(jj)} & \cdots & f_{2,m_j}^{(jj)} \\ \cdots & \cdots & \cdots & \cdots \\ f_{m_j,1}^{(jj)} & f_{m_j,2}^{(jj)} & \cdots & f_{m_j,m_j}^{(jj)} \\ f_{11}^{(i_1j)} & f_{12}^{(i_1j)} & \cdots & f_{1,m_j}^{(i_1j)} \\ f_{21}^{(i_1j)} & f_{22}^{(i_1j)} & \cdots & f_{2,m_j}^{(i_1j)} \\ \cdots & \cdots & \cdots & \cdots \\ f_{m_{i_1},1}^{(i_1j)} & f_{m_{i_1},2}^{(i_1j)} & \cdots & f_{m_{i_1},m_j}^{(i_1j)} \\ f_{11}^{(i_2j)} & f_{12}^{(i_2j)} & \cdots & f_{1,m_j}^{(i_2j)} \\ f_{21}^{(i_2j)} & f_{22}^{(i_2j)} & \cdots & f_{2,m_j}^{(i_2j)} \\ \cdots & \cdots & \cdots & \cdots \\ f_{m_{i_2},1}^{(i_2j)} & f_{m_{i_2},2}^{(i_2j)} & \cdots & f_{m_{i_2},m_j}^{(i_2j)} \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \end{bmatrix} \text{ or}$$

$$M_j = \begin{bmatrix} F_{jj} \\ F_{i_1j} \\ F_{i_2j} \\ \cdots \end{bmatrix},$$ where $F_{j,j}$, $F_{i,j}$, $F_{i,j}$ ...are submatrices containing the f coefficients in column j of the block sparse matrix A. Column j represents the first column of interest to be processed.

[0064] At step 215, a singular value decomposition is performed on the matrix *Mj* to generate matrices U_j, W_j, and V_j, where U_j represents a first or left matrix containing the left singular vectors, where W_j represents a second or diagonal matrix containing the singular values arranged in ascending (*i.e.*, from small to large) order, which is in reverse of the customary order, and where V_j represents a third or right matrix containing the right singular vectors. The singular value decomposition ensures that $M_j = U_j W_j V_j^T$, where V_j^T is the transpose of right matrix *Vj*. The columns of left matrix U_j and right matrix V_j are arranged in a manner such that each entry of diagonal matrix W_j is configured to multiply the appropriate column of left matrix U_j and the appropriate row of transposed right matrix V_j^T. Left matrix U_j may be expressed as:

$$U_j = \begin{bmatrix} u_{11}^{(jj)} & u_{12}^{(jj)} & \cdots & u_{1,m_j}^{(jj)} \\ u_{21}^{(jj)} & u_{22}^{(jj)} & \cdots & u_{2,m_j}^{(jj)} \\ \cdots & \cdots & \cdots & \cdots \\ u_{m_j,1}^{(jj)} & u_{m_j,2}^{(jj)} & \cdots & u_{m_j,m_j}^{(jj)} \\ u_{11}^{(i_1 j)} & u_{12}^{(i_1 j)} & \cdots & u_{1,m_j}^{(i_1 j)} \\ u_{21}^{(i_1 j)} & u_{22}^{(i_1 j)} & \cdots & u_{2,m_j}^{(i_1 j)} \\ \cdots & \cdots & \cdots & \cdots \\ u_{m_{i_1},1}^{(i_1 j)} & u_{m_{i_1},2}^{(i_1 j)} & \cdots & u_{m_{i_1},m_j}^{(i_1 j)} \\ u_{11}^{(i_2 j)} & u_{12}^{(i_2 j)} & \cdots & u_{1,m_j}^{(i_2 j)} \\ u_{21}^{(i_2 j)} & u_{22}^{(i_2 j)} & \cdots & u_{2,m_j}^{(i_2 j)} \\ \cdots & \cdots & \cdots & \cdots \\ u_{m_{i_2},1}^{(i_2 j)} & u_{m_{i_2},2}^{(i_2 j)} & \cdots & u_{m_{i_2},m_j}^{(i_2 j)} \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

Diagonal matrix W_j may be expressed as:

$$W_j = \begin{bmatrix} w_1^{(j)} & & & \\ & w_2^{(j)} & & \\ & & \cdots & \\ & & & w_{m_j}^{(j)} \end{bmatrix}$$

Right matrix V_j may be expressed as:

$$V_j = \begin{bmatrix} v_{11}^{(j)} & v_{12}^{(j)} & \cdots & v_{1,m_j}^{(j)} \\ v_{21}^{(j)} & v_{22}^{(j)} & \cdots & v_{2,m_j}^{(j)} \\ \cdots & \cdots & \cdots & \cdots \\ v_{m_j,1}^{(j)} & v_{m_j,2}^{(j)} & \cdots & v_{m_j,m_j}^{(j)} \end{bmatrix}$$

[0065] A more detailed description of the singular value decomposition is provided in the definition section above. It should be noted that to enhance the method, the Frobenius norm of the matrix M_j may be computed in block 215. The calculated Frobenius norm may be referred to as the column matrix norm. If the Frobenius norm is greater than a predetermined threshold value, the singular value decomposition is performed on the matrix Mj to generate matrices U_j, W_j, and V_j, as noted above. However, if the Frobenius norm is not larger than the predetermined matrix value, V_j is set equal to the identity matrix, the entries of W_j are set to zero, and the entries of U_j are not computed. That is, the method reduces the computations because the entries of U_j do not influence the outcome. Accordingly, the method is enhanced because certain computations are skipped or simplified.

[0066] At step 220, the singular values in diagonal matrix W_j that are smaller than a predetermined threshold value are discarded to generate reduced diagonal matrix $W_j^{(r)}$. In one embodiment, the singular values are discarded by setting them to zero. The number of remaining (or retained) singular values in diagonal matrix W_j that have not been discarded may be referred to as r_j. As such, reduced diagonal matrix $W_j^{(r)}$ may be expressed as:

$$W_j^{(r)} = \begin{bmatrix} 0 & & & & & \\ & \cdots & & & & \\ & & 0 & & & \\ & & & w_{m_j+1-r_j}^{(j)} & & \\ & & & & \cdots & \\ & & & & & w_{m_j}^{(j)} \end{bmatrix}$$

[0067] At step 225, left matrix U_j is multiplied by reduced diagonal matrix $W_j^{(r)}$ to generate a temporary column matrix $M_j^{(t)}$. The right r_j columns of temporary column matrix $M_j^{(t)}$ contain nonzero entries and the rest of the entries of temporary column matrix $M_j^{(t)}$ are zeroes. Temporary column matrix $M_j^{(t)}$ may be expressed as:

$$M_j^{(t)} = \begin{bmatrix} 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{1,m_j+1-r_j}^{(jj)} & \cdots & w_{m_j}^{(j)} u_{1,m_j}^{(jj)} \\ 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{2,m_j+1-r_j}^{(jj)} & \cdots & w_{m_j}^{(j)} u_{2,m_j}^{(jj)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{m_j,m_j+1-r_j}^{(jj)} & \cdots & w_{m_j}^{(j)} u_{m_j,m_j}^{(jj)} \\ 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{1,m_j+1-r_j}^{(i_1j)} & \cdots & w_{m_j}^{(j)} u_{1,m_j}^{(i_1j)} \\ 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{2,m_j+1-r_j}^{(i_1j)} & \cdots & w_{m_j}^{(j)} u_{2,m_j}^{(i_1j)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{m_{i_1},m_j+1-r_j}^{(i_1j)} & \cdots & w_{m_j}^{(j)} u_{m_{i_1},m_j}^{(i_1j)} \\ 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{1,m_j+1-r_j}^{(i_2j)} & \cdots & w_{m_j}^{(j)} u_{1,m_j}^{(i_2j)} \\ 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{2,m_j+1-r_j}^{(i_2j)} & \cdots & w_{m_j}^{(j)} u_{2,m_j}^{(i_2j)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{m_{i_2},m_j+1-r_j}^{(i_2j)} & \cdots & w_{m_j}^{(j)} u_{m_{i_2},m_j}^{(i_2j)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

[0068] At step 230, the coefficients in temporary column matrix $M_j^{(t)}$ are assembled into blocks corresponding to the original set of submatrices (*e.g.*, F_{ij}) containing the f coefficients in column j of the block sparse matrix A that were assembled into matrix M_j. That is, temporary column matrix $M_j^{(t)}$ may be expressed as:

$$M_j^{(t)} = \begin{bmatrix} F_{jj}^{(t)} \\ F_{i_1 j}^{(t)} \\ F_{i_2 j}^{(t)} \\ \cdots \end{bmatrix},$$

where temporary submatrices $F_{ij}^{(t)}$ may be expressed as:

$$F_{ij}^{(t)} = \begin{bmatrix} 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{1,m_j+1-r_j}^{(ij)} & \cdots & w_{m_j}^{(j)} u_{1,m_j}^{(ij)} \\ 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{2,m_j+1-r_j}^{(ij)} & \cdots & w_{m_j}^{(j)} u_{2,m_j}^{(ij)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & w_{m_j+1-r_j}^{(j)} u_{m_i,m_j+1-r_j}^{(ij)} & \cdots & w_{m_j}^{(j)} u_{m_i,m_j}^{(ij)} \end{bmatrix}.$$

[0069] At step 235, steps 210 through 230 are repeated for the rest of the columns in block sparse matrix A to obtain the rest of temporary column matrices in the block sparse matrix A.

[0070] At step 240, for each i for which V_i is not an identity matrix the temporary submatrix $F_{ij}^{(t)}$ in temporary column matrix $M_j^{(t)}$ is premultiplied by transposed right matrix V_i^T to generate a transformed version of submatrix $F_{ij}^{(t)}$. The product may be expressed as: $\hat{F}_{ij}^{(t)} = V_i^T F_{ij}^{(t)}$. For each i with V_i being an identity matrix, the product is $\hat{F}_{ij}^{(t)} = F_{ij}^{(t)}$.

[0071] At step 245, for each i for which V_i is not an identity matrix the subvector Φ_{ij} containing φ coefficients from submatrix A_{ij} is premultiplied by transposed right matrix V_i^T to generate a transformed version of subvector Φ_{ij}. The product may be expressed as: $\hat{\Phi}_{ij} = V_j^T \Phi_{ij}$. For each i with V_i being an identity matrix, the product is $\hat{\Phi}_{ij} = \Phi_{ij}$.

[0072] At step 250, for each j for which V_j is not an identity matrix the fluid partial volumes vector $\widetilde{V}_j$ is postmultiplied by V_j to generate a transformed version of fluid partial volumes vector $\widetilde{V}_j$. The product may be expressed as: $\hat{\widetilde{V}}_j = \widetilde{V}_j V_j$. For each j with V_j being an identity matrix, the product is $\hat{\widetilde{V}}_j = \widetilde{V}_j$.

[0073] At step 255, a transformed block sparse matrix $\hat{A}$ having the same block structure and submatrix form as block sparse matrix A is constructed. As such, the transformed off diagonal submatrix $\hat{A}_{ij}$, where $i \neq j$, may be expressed as:

$$\hat{A}_{ij} = \begin{bmatrix} \hat{F}_{ij}^{(t)} & \hat{\Phi}_{ij} \\ 0 & 0 \end{bmatrix}$$

The transformed diagonal submatrix $\hat{A}_{ii}$ may be expressed as:

$$\hat{A}_{ii} = \begin{bmatrix} I + \hat{F}_{ii}^{(t)} & \hat{\Phi}_{ii} \\ \hat{\widetilde{V}}_i & c_i \end{bmatrix}$$

or

$$\hat{A}_{ii} = \begin{bmatrix} 1 & \dots & 0 & \hat{f}^{(ii)}_{1,m_i+1-r_i} & \dots & \hat{f}^{(ii)}_{1,m_i} & \hat{\varphi}^{(ii)}_1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 1 & \hat{f}^{(ii)}_{m_i-r_i,m_i+1-r_i} & \dots & \hat{f}^{(ii)}_{m_i-r_i,m_i} & \hat{\varphi}^{(ii)}_{m_i-r_i} \\ 0 & \dots & 0 & 1+\hat{f}^{(ii)}_{m_i+1-r_i,m_i+1-r_i} & \dots & \hat{f}^{(ii)}_{m_i+1-r_i,m_i} & \hat{\varphi}^{(ii)}_{m_i+1-r_i} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & \hat{f}^{(ii)}_{m_i,m_i+1-r_i} & \dots & 1+\hat{f}^{(ii)}_{m_i,m_i} & \hat{\varphi}^{(ii)}_{m_i} \\ \hat{\widetilde{V}}^{(i)}_1 & \dots & \hat{\widetilde{V}}^{(i)}_{m_i-r_i} & \hat{\widetilde{V}}^{(i)}_{m_i+1-r_i} & \dots & \hat{\widetilde{V}}^{(i)}_{m_i} & c_i \end{bmatrix}$$

[0074] At step 260, the transformed partial volumes that correspond to the discarded singular values in transformed block sparse matrix $\hat{A}$ are eliminated. In one embodiment, the transformed partial volumes $\hat{\widetilde{V}}^{(i)}_1, \dots, \hat{\widetilde{V}}^{(i)}_{m_i-r_i}$ that correspond to the discarded singular values in the transformed diagonal submatrix $\hat{A}_{ii}$ are eliminated by multiplying $\hat{\widetilde{V}}^{(i)}_k$ by $\hat{f}^{(ii)}_{k,m_i+1-r_i}, \dots, \hat{f}^{(ii)}_{k,m_i}, \hat{\varphi}^{(ii)}_k$ and subtracting the result from $\hat{\widetilde{V}}^{(i)}_{m_i+1-r_i}, \dots, \hat{\widetilde{V}}^{(i)}_{m_i}, c_i$, where $k = 1, \dots, m_j - r_j$. A similar operation is performed on the transformed off diagonal submatrix $\hat{A}_{ij}$ by multiplying the transformed partial volumes

$\hat{\tilde{V}}_k^{(i)}$ with $\hat{f}_{k,m_j+1-r_j}^{(ij)},\ldots,\hat{f}_{k,m_j}^{(ij)},\hat{\varphi}_k^{(ij)}$ and subtracting the result from the bottom row of the transformed off diagonal submatrix $\hat{A}_{ij}$. The elimination process may be repeated for all rows of transformed block sparse matrix $\hat{A}$. Once the transformed partial volumes that correspond to the discarded singular values in transformed block sparse matrix $\hat{A}$ are eliminated, the transformed diagonal submatrix $\hat{A}_{ii}$ may be expressed as:

$$\hat{A}_{ii}=\begin{bmatrix} 1 & \ldots & 0 & \hat{f}_{1,m_i+1-r_i}^{(ii)} & \ldots & \hat{f}_{1,m_i}^{(ii)} & \hat{\varphi}_1^{(ii)} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & 1 & \hat{f}_{m_i-r_i,m_i+1-r_i}^{(ii)} & \ldots & \hat{f}_{m_i-r_i,m_i}^{(ii)} & \hat{\varphi}_{m_i-r_i}^{(ii)} \\ 0 & \ldots & 0 & 1+\hat{f}_{m_i+1-r_i,m_i+1-r_i}^{(ii)} & \ldots & \hat{f}_{m_i+1-r_i,m_i}^{(ii)} & \hat{\varphi}_{m_i+1-r_i}^{(ii)} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & 0 & \hat{f}_{m_i,m_i+1-r_i}^{(ii)} & \ldots & 1+\hat{f}_{m_i,m_i}^{(ii)} & \hat{\varphi}_{m_i}^{(ii)} \\ 0 & \ldots & 0 & \hat{f}_{m_i+1,m_i+1-r_i}^{(ii)} & \ldots & \hat{f}_{m_i+1,m_i}^{(ii)} & \hat{\varphi}_{m_i+1}^{(ii)} \end{bmatrix}$$

The transformed off diagonal submatrix $\hat{A}_{ij}$, where $i \neq j$, may be expressed as:

$$\hat{A}_{ij}=\begin{bmatrix} 0 & \ldots & 0 & \hat{f}_{1,m_j+1-r_j}^{(ij)} & \ldots & \hat{f}_{1,m_j}^{(ij)} & \hat{\varphi}_1^{(ij)} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & 0 & \hat{f}_{m_i-r_i,m_j+1-r_j}^{(ij)} & \ldots & \hat{f}_{m_i-r_i,m_j}^{(ij)} & \hat{\varphi}_{m_i-r_i}^{(ij)} \\ 0 & \ldots & 0 & \hat{f}_{m_i+1-r_i,m_j+1-r_j}^{(ij)} & \ldots & \hat{f}_{m_i+1-r_i,m_j}^{(ij)} & \hat{\varphi}_{m_i+1-r_i}^{(ij)} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & 0 & \hat{f}_{m_i,m_j+1-r_j}^{(ij)} & \ldots & \hat{f}_{m_i,m_j}^{(ij)} & \hat{\varphi}_{m_i}^{(ij)} \\ 0 & \ldots & 0 & \hat{f}_{m_i+1,m_j+1-r_j}^{(ij)} & \ldots & \hat{f}_{m_i+1,m_j}^{(ij)} & \hat{\varphi}_{m_i+1}^{(ij)} \end{bmatrix}$$

[0075] At step 265, a reduced transformed block sparse matrix $\hat{A}^{(r)}$ is constructed. In the reduced transformed diagonal submatrices $\hat{A}_{ii}^{(r)}$ of the reduced transformed block sparse matrix $\hat{A}^{(r)}$, the bottom r_i+1 rows contain nonzero coefficients in only the right most r_i+1 columns. These coefficients are placed in a matrix of smaller dimension, resulting in the reduced transformed diagonal submatrix $\hat{A}_{ii}^{(r)}$, which may be expressed as:

$$\hat{A}_{ii}^{(r)} = \begin{bmatrix} 1+\hat{f}_{m_i+1-r_i,m_i+1-r_i}^{(ii)} & \cdots & \hat{f}_{m+1-r,m_i}^{(ii)} & \hat{\varphi}_{m_i+1-r_i}^{(ii)} \\ \cdots & \cdots & \cdots & \cdots \\ \hat{f}_{m_i,m_i+1-r_i}^{(ii)} & \cdots & 1+\hat{f}_{m_i,m_i}^{(ii)} & \hat{\varphi}_{m_i}^{(ii)} \\ \hat{f}_{m_i+1,m_i+1-r_i}^{(ii)} & \cdots & \hat{f}_{m_i+1,m_i}^{(ii)} & \hat{\varphi}_{m_i+1}^{(ii)} \end{bmatrix}$$

[0076] The reduced transformed off-diagonal submatrices $\hat{A}_{ij}^{(r)}$ for the reduced transformed block sparse matrix $\hat{A}^{(r)}$ contain only coefficients within the bottom r_i+1 rows and the right most r_j+1 columns. As such, the reduced transformed off diagonal submatrix $\hat{A}_{ij}^{(r)}$ may be expressed as:

$$\hat{A}_{ij}^{(r)} = \begin{bmatrix} \hat{f}_{m_i+1-r_i,m_j+1-r_j}^{(ij)} & \cdots & \hat{f}_{m_i+1-r_i,m_j}^{(ij)} & \hat{\varphi}_{m_i+1-r_i}^{(ij)} \\ \cdots & \cdots & \cdots & \cdots \\ \hat{f}_{m_i,m_j+1-r_j}^{(ij)} & \cdots & \hat{f}_{m_i,m_j}^{(ij)} & \hat{\varphi}_{m_i}^{(ij)} \\ \hat{f}_{m_i+1,m_j+1-r_j}^{(ij)} & \cdots & \hat{f}_{m_i+1,m_j}^{(ij)} & \hat{\varphi}_{m_i+1}^{(ij)} \end{bmatrix}$$

[0077] At step 270, the residual block vector R is initialized by setting it equal to the right-hand side vector B. That is, the initial residual block vector $R^{(0)}$ is set equal to the right-hand side vector B.

[0078] At step 275, a transformed residual block vector $\hat{R}_i$ is constructed. The transformed residual block vector $\hat{R}_i$ may be expressed as:

$$\hat{R}_i = \begin{bmatrix} \hat{R}_{M.i} \\ \hat{R}_{V.i} \end{bmatrix}.$$

The. transformed mass balance residual subvector $\hat{R}_{M.i}$ may be expressed as: $\hat{R}_{M.i} = V_i^T R_{M,i}^{(0)}$. The transformed volume constraint residual subvector $\hat{R}_{V.i}$ may be computed by multiplying the transformed partial volumes $\hat{\hat{V}}_k^{(i)}$ that correspond to the discarded singular values in the transformed block sparse matrix $\hat{A}$ with $\hat{r}_k^{(i)}$ and subtracting the result from the volume constraint residual subvector $R_{V,i}$, where $k = 1, \ldots, m_i - r_i - 1$.

[0079] At step 280, a reduced transformed residual block vector block vector $\hat{R}^{(r)}$ is constructed and expressed as:

$$\hat{R}_i^{(r)} = \begin{bmatrix} \hat{r}_{m_i+1-r_i}^{(i)} \\ \cdots \\ \hat{r}_{m_i}^{(i)} \\ \hat{r}_{m_i+1}^{(i)} \end{bmatrix}$$

where the values of the reduced transformed residual block vector block vector $\hat{R}^{(r)}$ are obtained from the transformed mass balance residual subvector $\hat{R}_{M.i}$.

[0080] At step 282, the matrix equation $\hat{A}^{(r)}\delta\hat{X}^{(r)} = \hat{R}^{(r)}$ is solved for a reduced transformed solution change block vector $\delta\hat{X}^{(r)}$, which contains a reduced set of mass and pressure unknowns.

[0081] At step 284, a partial transformed solution change vector $\delta\hat{X}^{(0)}$ is created by setting each subvector as follows:

$$\delta\hat{X}_i^{(0)} = \begin{bmatrix} 0 \\ \cdots \\ 0 \\ \delta\hat{x}_{m_i+1-r_i}^{(r,i)} \\ \cdots \\ \delta\hat{x}_{m_i}^{(r,i)} \\ \delta\hat{p}^{(i)} \end{bmatrix}$$

The zeroes correspond to the discarded singular values.

[0082] At step 286, a transformed residual block vector $\hat{R}^{(0)}$ is computed using the partial transformed solution change vector $\delta\hat{X}^{(0)}$ and the following equation:

$$\hat{R}^{(0)} = \hat{R} - \hat{A}\delta\hat{X}^{(0)}$$

[0083] At step 288, the values of the transformed solution change vector $\delta\hat{X}_i$ are constructed as follows:

$$\delta\hat{X}_i = \begin{bmatrix} \hat{r}_1^{(0,i)} \\ \cdots \\ \hat{r}_{m_i-r_i}^{(0,i)} \\ \delta\hat{x}_{m_i+1-r_i}^{(r,i)} \\ \cdots \\ \delta\hat{x}_{m_i}^{(r,i)} \\ \delta\hat{p}^{(i)} \end{bmatrix},$$

where the values in the lower half of the transformed solution change vector $\delta\hat{X}_i$ are obtained from the lower half of the partial transformed solution change vector $\delta\hat{X}^{(0)}$ and values in the upper half of the transformed solution change vector $\delta\hat{X}_i$ are obtained from the transformed residual block vector $\hat{R}^{(0)}$ determined at step 286.

[0084] At 290, the mass changes δM_i are computed according to $\delta M_i = V_i\ X_{m,i}$, where V_i represents the matrix containing the right singular vectors, as previously described with reference to step 215 and $\delta X_{m,i}$ represents a vector containing only the mass unknown entries in $\delta\hat{X}_i$. At step 292, the mass changes δM_i are then used to assemble a solution change block subvector δX_i as follows:

$$\delta X_i = \begin{bmatrix} \delta M_i \\ \delta P_i \end{bmatrix}$$

[0085] At step 294, the solution change subvector δX_i is then used to update the solution subvector X_i as follows $X_i = X_i + \delta X_i$. At step 296, steps 290-294 are repeated for each subvector of X. In this manner, an estimate for the current solution block vector X is computed.

[0086] At step 297, an updated residual block vector R is computed according to $R = B - AX$. At step 298, a determination is made as to whether the updated residual block vector R and the solution block vector X have satisfied a predetermined stopping criteria. If the answer is in the negative, processing returns to step 275. If the answer is in the affirmative, then processing ends.

[0087] Figure 3 illustrates a computer network 300, into which embodiments of the invention may be implemented. The computer network 300 includes a system computer 330, which may be implemented as any conventional personal computer or workstation, such as a UNIX-based workstation. The system computer 330 is in communication with disk storage devices 329, 331, and 333, which may be external hard disk storage devices. It is contemplated that disk storage devices 329, 331, and 333 are conventional hard disk drives, and as such, will be implemented by way of a local area network or by remote access. Of course, while disk storage devices 329, 331, and 333 are illustrated as separate devices, a single disk storage device may be used to store any and all of the program instructions, measurement data, and results as desired.

[0088] In one embodiment, the input data are stored in disk storage device 331. The system computer 330 may retrieve the appropriate data from the disk storage device 331 to solve the implicit reservoir simulation matrix equation according to program instructions that correspond to the methods described herein. The program instructions may be written in a computer programming language, such as C++, Java and the like. The program instructions may be stored in a computer-readable memory, such as program disk storage device 333. Of course, the memory medium storing the program instructions may be of any conventional type used for the storage of computer programs, including hard disk drives, floppy disks, CD-ROMs and other optical media, magnetic tape, and the like.

[0089] According to a preferred embodiment, the system computer 330 presents output primarily onto graphics display 327, or alternatively via printer 328. The system computer 230 may store the results of the methods described above on disk storage 329, for later use and further analysis. The keyboard 326 and the pointing device (e.g., a mouse, trackball, or the like) 225 may be provided with the system computer 330 to enable interactive operation.

[0090] The system computer 330 may be located at a data center remote from the reservoir. While Figure 3 illustrates the disk storage 331 as directly connected to the system computer 330, it is also contemplated that the disk storage device 331 may be accessible through a local area network or by remote access. Furthermore, while disk storage devices 329, 331 are illustrated as separate devices for storing input data and analysis results, the disk storage devices 329, 331 may be implemented within a single disk drive (either together with or separately from program disk storage device 333), or in any other conventional manner as will be fully understood by one of skill in the art having reference to this specification.

WHAT IS CLAIMED IS:

1. A method for solving a matrix equation AX=B, wherein A represents a block sparse matrix, B represents a right hand side block vector and X represents a solution block vector, the method comprising:

receiving the block sparse matrix and the right hand side block vector;

constructing a reduced transformed block sparse matrix from the block sparse matrix;

constructing a reduced transformed residual block vector from the block sparse matrix and the right hand side block vector; wherein constructing the reduced transformed block sparse matrix comprises:

assembling one or more coefficients of one or more mass change terms from a column of the block sparse matrix into a column matrix;

performing a singular value decomposition on the column matrix to generate a left matrix, a diagonal matrix and a right matrix; and

discarding one or more singular values that are less than a predetermined threshold value to generate a reduced diagonal matrix; and

solving for the solution block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector.

2. The method of claim 1, wherein the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and the one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and the one or more coefficients of the mass change terms.

3. The method of claim 1, wherein constructing the reduced transformed block sparse matrix further comprises constructing a transformed block sparse matrix having the same block structure and submatrix form as the block sparse matrix.

4. The method of claim 1, wherein the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change

terms and the one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and the one or more coefficients of the mass change terms, and wherein the diagonal matrix comprises one or more singular values arranged in ascending order.

5. The method of claim 1, wherein the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and the one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and the one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix comprises multiplying the left matrix by the reduced diagonal matrix to generate a temporary column matrix.

6. The method of claim 1, wherein the matrix equation represents a system of fluid flow equations in one or more dimensions having one or more pressure change terms and the one or more mass change terms, wherein the block sparse matrix contains one or more coefficients of the pressure change terms and the one or more coefficients of the mass change terms, and wherein constructing the reduced transformed block sparse matrix comprises:

multiplying the left matrix by the reduced diagonal matrix to generate a temporary column matrix having a plurality of mass change terms coefficients; and

assembling the mass change terms coefficients of the temporary column matrix into one or more temporary mass change terms coefficient submatrices that correspond to the mass change terms coefficients of the column matrix.

7. The method of claim 6, wherein constructing the reduced transformed block sparse matrix comprises premultiplying each temporary mass change terms coefficient submatrix by the transpose of the right matrix to generate a transformed temporary mass change terms coefficient submatrix.

8. The method of claim 7, wherein constructing the reduced transformed block sparse matrix comprises premultiplying each subvector containing the pressure change terms coefficients in the block sparse matrix by the transpose of the right matrix to generate a transformed pressure change terms coefficients subvector.

9. The method of claim 8, wherein constructing the reduced transformed block sparse matrix comprises postmultiplying each fluid partial volumes subvector in the block sparse matrix by the right matrix to generate a transformed fluid partial volumes vector.

10. The method of claim 9, wherein constructing the reduced transformed block sparse matrix comprises constructing a transformed block sparse matrix from one or more of the transformed temporary mass change terms coefficient submatrix, the transformed pressure change terms coefficients subvector and the transformed fluid partial volumes vector.

11. The method of claim 10, wherein constructing the reduced transformed block sparse matrix comprises eliminating one or more transformed fluid partial volumes that correspond to the discarded singular values in the transformed block sparse matrix to generate the reduced transformed block sparse matrix.

12. The method of claim 1, wherein constructing the reduced transformed block sparse matrix comprises:

determining a column matrix norm for the column matrix;

skipping the singular value decomposition if the column matrix norm is equal to or less than the predetermined threshold value;

setting the singular values to zero if the column matrix norm is equal to or less than the predetermined threshold value; and

setting the right matrix to the identity matrix if the column matrix norm is equal to or less than the predetermined threshold value.

13. The method of claim 12, wherein determining the column matrix norm comprises calculating the Frobenius norm of the column matrix.

14. The method of claim 1, wherein the reduced transformed block sparse matrix comprises one or more reduced transformed diagonal submatrices and one or more reduced transformed off-diagonal submatrices, wherein each reduced transformed diagonal submatrix comprises mass change terms coefficients and pressure change terms coefficients only within the bottom r_i+1 rows and right most r_i+1 columns of each transformed diagonal submatrix.

15. The method of claim 14, wherein each reduced transformed off-diagonal submatrix comprises mass change terms coefficients and pressure change terms coefficients only within the bottom r_i+1 rows and the right most r_j+1 columns of each transformed off-diagonal submatrix.

16. The method of claim 1, wherein constructing the reduced transformed residual block vector comprises constructing a transformed residual block vector.

17. The method of claim 1, wherein constructing the transformed residual block vector comprises constructing a transformed residual block vector having a transformed mass balance residual subvector and a transformed volume constraint residual subvector.

18. The method of claim 1, wherein constructing the reduced transformed residual block vector comprises premultiplying a mass balance residual subvector by the transpose of the right matrix to generate a transformed mass balance residual subvector.

19. The method of claim 1, wherein solving for the solution block vector comprises solving for a reduced transformed solution change block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector.

20. The method of claim 1, wherein solving for the solution block vector comprises:

solving for a reduced transformed solution change block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector; and

converting the reduced transformed solution change block vector to a solution change block vector.

21. The method of claim 1, wherein solving for the solution block vector comprises

adding the solution change block vector to a current estimate of the solution block vector to update the solution block vector.

22. The method of claim 1, wherein solving for the solution block vector comprises:

solving for a reduced transformed solution change block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector; and

converting the reduced transformed solution change block vector to a solution change block vector having one or more changes in mass unknowns and one or more changes in pressure unknowns.

23. A method for solving a matrix equation AX=B, wherein A represents a block sparse matrix, B represents a right hand side block vector and X represents a solution block vector, the method comprising:

constructing a reduced transformed block sparse matrix from the block sparse matrix;

constructing a reduced transformed residual block vector from the block sparse matrix and the right hand side block vector;

solving for a reduced transformed solution change block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector;

converting the reduced transformed solution change block vector to a solution change block vector having one or more changes in mass unknowns and one or more changes in pressure unknowns; and

adding the solution change block vector to a current estimate of the solution block vector to update the solution block vector.

24. A method for solving a matrix equation AX=B, wherein A represents a block sparse matrix, B represents a first block vector and X represents a solution block vector, the method comprising:

receiving the block sparse matrix and the first block vector;

constructing a reduced transformed block sparse matrix from the block sparse matrix;

constructing a reduced transformed residual block vector from the block sparse matrix and the first block vector; wherein constructing the reduced transformed block sparse matrix comprises:

assembling at least one coefficient associated with the at least one mass change term from a column of the block sparse matrix into a column matrix;

performing a singular value decomposition on the column matrix to generate a first matrix, a second matrix and a third matrix; and

discarding each singular value less than a predetermined threshold value to generate a reduced diagonal matrix; and

solving for the solution block vector using the reduced transformed block sparse matrix and the reduced transformed residual block vector.

25. The method of claim 24, wherein the matrix equation represents fluid flow equations in a reservoir having at least one pressure change term and at least one mass change term, wherein the block sparse matrix has at least one coefficient associated

with the at least one pressure change term and at least one coefficient associated with the at least one mass change term.

26. The method of claim 24, wherein constructing the reduced transformed block sparse matrix comprises:

determining a column matrix norm for the column matrix;

skipping the singular value decomposition if the column matrix norm is equal to or less than a predetermined threshold value;

setting the singular values to zero if the column matrix norm is equal to or less than the predetermined threshold value; and

setting the third matrix to the identity matrix if the column matrix norm is equal to or less than the predetermined matrix threshold.

27. The method of claim 26, wherein determining the column matrix norm comprises calculating the Frobenius norm of the column matrix.

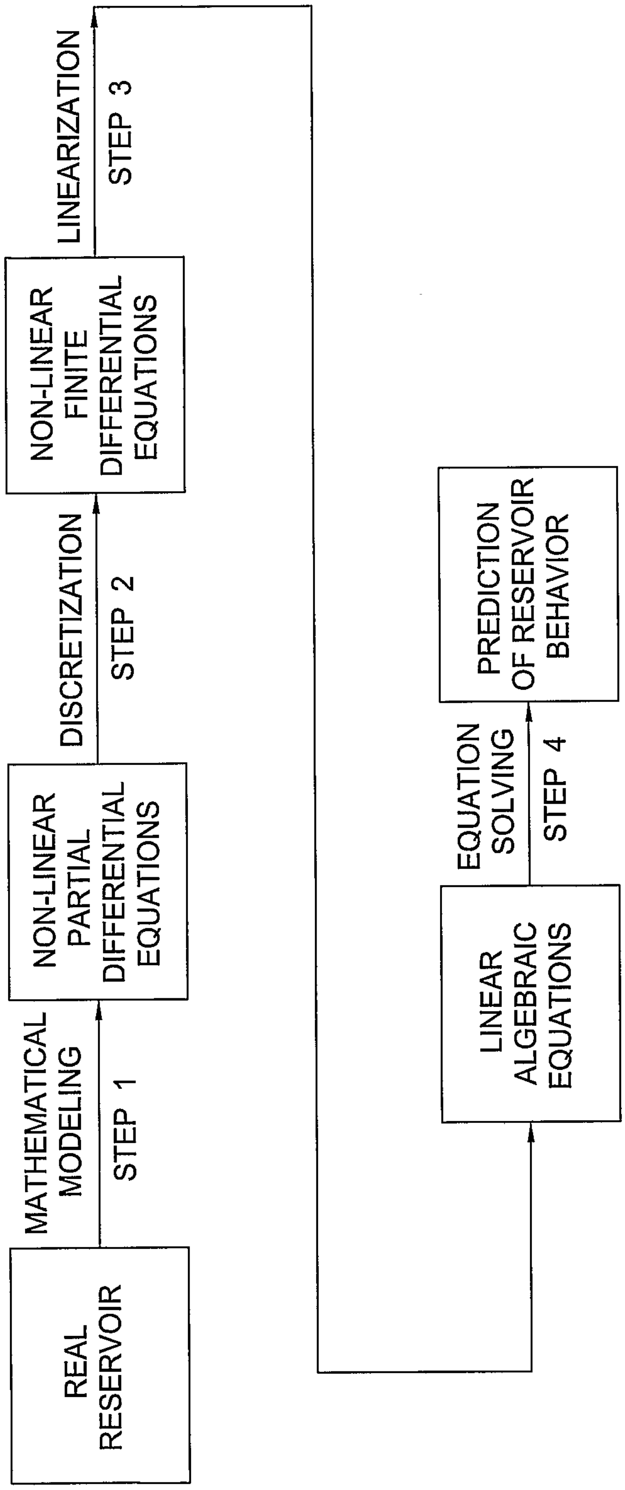
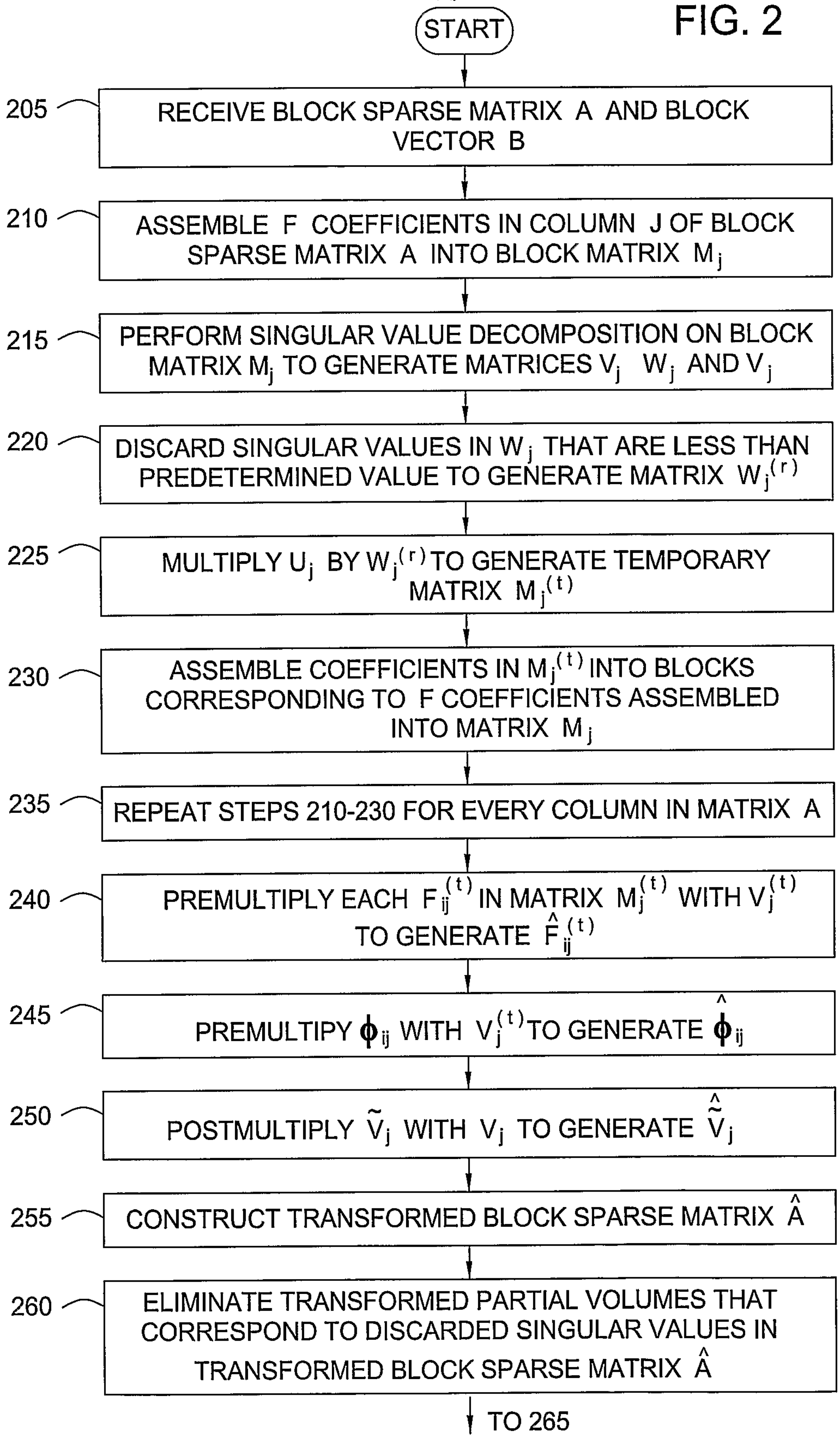

REAL RESERVOIR
MATHEMATICAL MODELING
STEP 1
NON-LINEAR PARTIAL DIFFERENTIAL EQUATIONS
DISCRETIZATION
STEP 2
NON-LINEAR FINITE DIFFERENTIAL EQUATIONS
LINEARIZATION
STEP 3
LINEAR ALGEBRAIC EQUATIONS
EQUATION SOLVING
STEP 4
PREDICTION OF RESERVOIR BEHAVIOR

FIG. 1

FIG. 2


START
205
RECEIVE BLOCK SPARSE MATRIX A AND BLOCK VECTOR B
210
ASSEMBLE F COEFFICIENTS IN COLUMN J OF BLOCK SPARSE MATRIX A INTO BLOCK MATRIX M_j
215
PERFORM SINGULAR VALUE DECOMPOSITION ON BLOCK MATRIX M_j TO GENERATE MATRICES V_j W_j AND V_j
220
DISCARD SINGULAR VALUES IN W_j THAT ARE LESS THAN PREDETERMINED VALUE TO GENERATE MATRIX $W_j^{(r)}$
225
MULTIPLY U_j BY $W_j^{(r)}$ TO GENERATE TEMPORARY MATRIX $M_j^{(t)}$
230
ASSEMBLE COEFFICIENTS IN $M_j^{(t)}$ INTO BLOCKS CORRESPONDING TO F COEFFICIENTS ASSEMBLED INTO MATRIX M_j
235
REPEAT STEPS 210-230 FOR EVERY COLUMN IN MATRIX A
240
PREMULTIPLY EACH $F_{ij}^{(t)}$ IN MATRIX $M_j^{(t)}$ WITH $V_j^{(t)}$ TO GENERATE $\hat{F}_{ij}^{(t)}$
245
PREMULTIPY ϕ_{ij} WITH $V_j^{(t)}$ TO GENERATE $\hat{\phi}_{ij}$
250
POSTMULTIPLY $\tilde{V}_j$ WITH V_j TO GENERATE $\hat{\tilde{V}}_j$
255
CONSTRUCT TRANSFORMED BLOCK SPARSE MATRIX $\hat{A}$
260
ELIMINATE TRANSFORMED PARTIAL VOLUMES THAT CORRESPOND TO DISCARDED SINGULAR VALUES IN TRANSFORMED BLOCK SPARSE MATRIX $\hat{A}$
TO 265

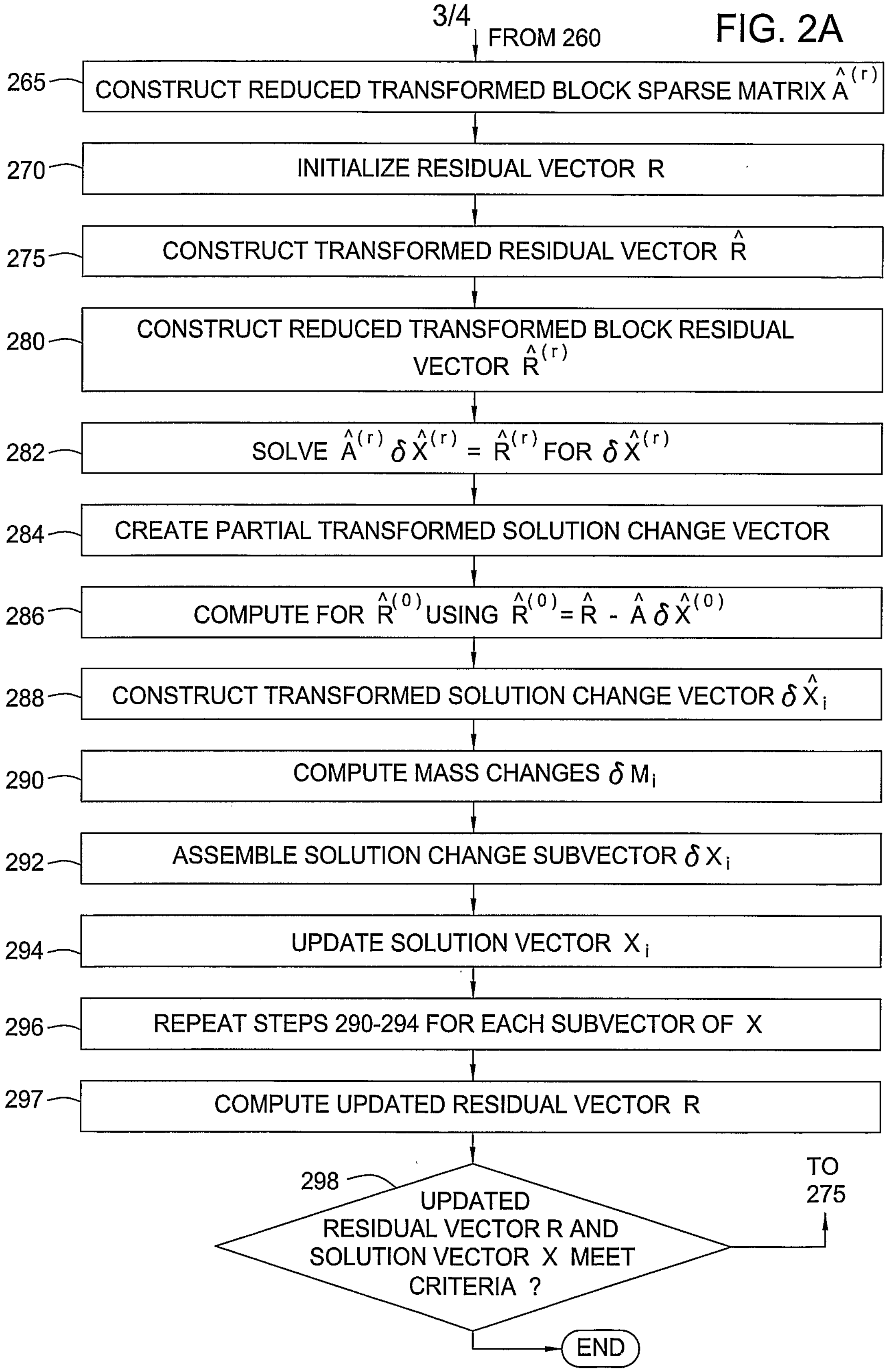

FROM 260
265
CONSTRUCT REDUCED TRANSFORMED BLOCK SPARSE MATRIX $\hat{A}^{(r)}$
270
INITIALIZE RESIDUAL VECTOR R
275
CONSTRUCT TRANSFORMED RESIDUAL VECTOR $\hat{R}$
280
CONSTRUCT REDUCED TRANSFORMED BLOCK RESIDUAL VECTOR $\hat{R}^{(r)}$
282
SOLVE $\hat{A}^{(r)} \delta \hat{X}^{(r)} = \hat{R}^{(r)}$ FOR $\delta \hat{X}^{(r)}$
284
CREATE PARTIAL TRANSFORMED SOLUTION CHANGE VECTOR
286
COMPUTE FOR $\hat{R}^{(0)}$ USING $\hat{R}^{(0)} = \hat{R} - \hat{A} \delta \hat{X}^{(0)}$
288
CONSTRUCT TRANSFORMED SOLUTION CHANGE VECTOR $\delta \hat{X}_i$
290
COMPUTE MASS CHANGES δM_i
292
ASSEMBLE SOLUTION CHANGE SUBVECTOR δX_i
294
UPDATE SOLUTION VECTOR X_i
296
REPEAT STEPS 290-294 FOR EACH SUBVECTOR OF X
297
COMPUTE UPDATED RESIDUAL VECTOR R
298
UPDATED RESIDUAL VECTOR R AND SOLUTION VECTOR X MEET CRITERIA ?
TO 275
END

FIG. 2A

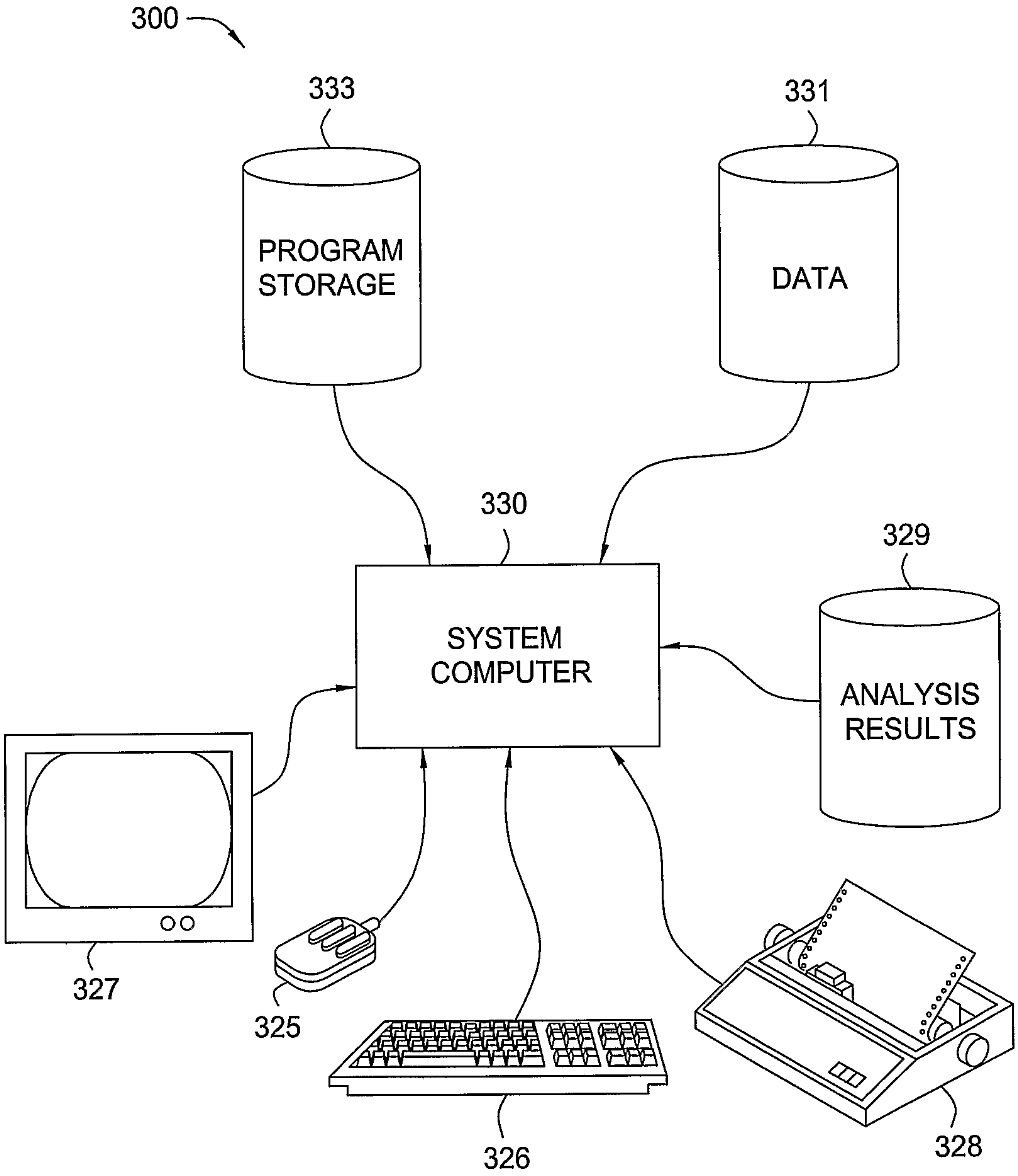

300
333
PROGRAM STORAGE
331
DATA
330
SYSTEM COMPUTER
329
ANALYSIS RESULTS
327
325
326
328

FIG. 3

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US05/12629

A. CLASSIFICATION OF SUBJECT MATTER
IPC(7) : G01V 01/00, 01/28; G06F 19/00; G06F 17/10, 07/60; G06G 7/48
US CL : 703/2, 703/10, 702/14, 702/16
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
U.S. : 703/2, 703/10, 702/14, 702/16

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	No Reference	

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" *document of particular relevance;* the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
22 August 2005 (22.08.2005)	13 SEP 2005
Name and mailing address of the ISA/US Mail Stop PCT, Attn: ISA/US Commissioner for Patents P.O. Box 1450 Alexandria, Virginia 22313-1450 Facsimile No. (703) 305-3230	Authorized officer for Michelle L. [signature] Leo Picard Telephone No. (703) 305-3900

Form PCT/ISA/210 (second sheet) (January 2004)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US05/12629

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. [X] Claims Nos.: 1-27
because they relate to subject matter not required to be searched by this Authority, namely:
(1) The claims are directed to an abstract mathematical algorithm, and (2) the claims do not produce a concrete, useful and tangible result.

2. [ ] Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. [ ] Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

1. [ ] As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. [ ] As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.
3. [ ] As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. [ ] No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest [ ] The additional search fees were accompanied by the applicant's protest.
[ ] No protest accompanied the payment of additional search fees.

Continuation of Item 4 of the first sheet:
Title is longer than 7 words.

New Title: "METHOD FOR SOLVING IMPLICIT RESERVOIR SIMULATION MATRIX"