演算量を最小にする縮約アドミタンス行列法

正員 的場 誠一 (株式会社 開発計算センター)

正員 周 小明 (株式会社 開発計算センター)

正員 関田 昌弘 (電源開発 株式会社)

正員 大石 孝穂 (電源開発 株式会社)

A Reduced Admittance Matrix Method Minimizing Operations

Seiichi Matoba, Member (The Kaihatsu Computing Service Center Co.LTD), Xiaming Zhou, Member (The Kaihatsu Computing Service Center Co.LTD), Masahiro Sekita, Member (Electric Power Development Co.LTD), Takaho Ohishi, Member (Electric Power Development Co.LTD)

Two general approaches are feasible to reduce the computation time of network equation described as $Y \cdot V = I$.

The first method is based on the sparsity of admittance matrix and injection current vector. This method is widely spread in power system analysis. Recently, sparse vector method is one of the most remarkable methods. The other hand, second one is based on reduced matrix method. But the bottlenecks of this method are the fact that reducing the dimension of admittance matrix, it becomes dense. Consequently due to a number of generations of fill-in elements, operations of the reduced matrix are rather increasing.

This paper presents a new technique of the reduced admittance matrix method minimizing the number of operations. The proposed algorithm was tested using three model systems. The result shows that the proposed method performs much better than sparse vector methods using MD-ML.

キーワード：縮約アドミタンス行列法、スパースベクトル法、オーダリング、T2、T3、MD-ML

1 まえがき

電力系統解析において、アドミタンス行列を係数とする線形方程式を解くことがよく現れる。例えば、安定度計算の過渡時潮流計算や固有値解析においてである。一般に $Y \cdot V = I$ で記述される線形方程式の計算時間を削減する手法として2つのアプローチがある。

1つはアドミタンス行列と注入電流ベクトルのスパース性をフルに利用する手法である。スパースベクトル法⁽²⁾は、これ等の手法の内、近年最も注目されている手法である。これに対し、もう1つのアプローチとして、ノード縮約法が考えられる。しかし、一般に多数のノード縮約を行うと残りのノードの結合が密になり、そのため、かえって演算量が多くなり計算速度も遅くしてしまうことになる。

本論文では、ネットワーク方程式において演算量を最小にする縮約アドミタンス行列の作成法を提案し、その効果をテスト計算によって確かめる。

2 ネットワーク方程式と縮約アドミタンス行列

(1)式で与えられるネットワーク方程式を考える。

$$Y \cdot V = I \quad (1)$$

ここで、

Y : アドミタンス行列

V : 母線電圧

I : 母線注入電流

である。

アドミタンス行列（以下 Y 行列と記す）は、自端のノードとその接続ノードのみが非ゼロで、他は0であるという強いスパース性を持つ行列である。従って、このスパース性をフルに利用した計算手法が適用されている。また電力系統解析では全てのノードの電圧の計算が必要であることは少なく、その一部（例えば発電機ノードのみ）を必要とすることが多い。

近年では、この要求を満たすために右辺のベクトルのスパース性にも着目した計算手法スパースベクトル法⁽³⁻⁶⁾が提案されている。この手法は、並列計算やPMR⁽⁷⁾ (Partial Matrix Refactorization) にも適用され、にわかに注目を浴びている。スパースベクトル法と、縮約 Y 行列の関係については後述することにする。

(1)式において、浮遊ノードの注入電流ベクトルは0であり、負荷ノードに関しても線形化して Y 行列を修正することにより注入電流ベクトルを0とすることができる。

(1)式の i 番目の右辺ベクトルが0であるとすると、

$$\sum_j y_{ij} r_j = 0 \quad (2)$$

となる。

従って、

$$r_i = - \sum_{j \neq i} \frac{y_{ij}}{y_{ii}} r_j = 0 \quad (3)$$

となり、 i 番ノードは消去できることになる。

このような操作を繰り返して行くと、発電機ノードだけの縮約Y行列を作成することができる。

この操作をまとめると以下のようにになる。オーダリングした行列は、

$$\left[\begin{array}{c|c} Y_{LL} & Y_{LG} \\ \hline Y_{GL} & Y_{GG} \end{array} \right] \left[\begin{array}{c} V_L \\ V_G \end{array} \right] = \left[\begin{array}{c} 0 \\ I_G \end{array} \right] \quad (4)$$

となる。ここで、

L : 負荷、浮遊ノード

G : 発電機ノード

である。

(4)式より、負荷ノードの電圧は、

$$V_L = -Y_{LL}^{-1} \cdot Y_{LG} \cdot V_G \quad (5)$$

で与えられる。これより V_L を消去すれば、

$$Y_G \cdot V_G = I_G \quad (6)$$

となる。

$$Y_G = Y_{GG} - Y_{GL} \cdot Y_{LL}^{-1} \cdot Y_{LG} \quad (7)$$

図 1: 一般の系統構成

Fig.1. General network system

ここで、 Y_G が発電機間縮約Y行列である。図1に示すように、発電機ノードはほとんどの場合端点（ブランチの接続数が1つのノード）であるので、 Y_G は完全に密行列になる。

このようなノード縮約法は一般に発電機の台数が少数の場合は縮約の効果があるが、大規模系統の場合は縮約の効果はなく、かえって非ゼロ要素数の増加を招き計算速度を悪化させることになる。

例えば、ノード数1000、ブランチ数1200、発電機150機の系統について考えてみよう。ノード縮約を行わないでスパース行列のまま処理する手法では、LU分解のフィルイン数をノード数と同数と仮定すると、非ゼロ要素数は、

$$N_1 = 1200 \times 2 + 1000 + 1000 = 4400$$

となる。（日本のようなループが少なく放射状の大規模系統ではオーダリングすることによりフィルイン数はかなり

少なく抑えることが可能となる。しかし、ここではシビアサイドに見積もって、フィルイン数をノード数と同数と仮定する。）

一方ノード縮約後の発電機縮約Y行列の要素数は、

$$N_2 = 150 \times 150 = 22500$$

となり、これはノード縮約をしない行列の4.6倍にもなってしまう。

以上のことから、大規模な行列では、発電機縮約Y行列による計算法は計算速度、記憶容量の面で縮約を行わない計算法より劣っていると考えることができる。

3 演算量を最小にする縮約アドミタンス行列

前章までの考察で、ノード縮約は行列の次元は縮約するが、演算量は増やすことがわかった。本章では、演算量を最小にする縮約Y行列の作成法について考察する。

ノードの縮約プロセスはノードのオーダリングのプロセスと同一視することができる。現在電力系統解析のオーダリング手法としては、Tinney氏が提案した手法T2^[1]が広く一般的に用いられている。

T2のアルゴリズムはノードを消去してできる縮約グラフの中で、そのブランチの接続数が最も少ないノードを次の消去ノードとするものである。1つのノードを消去するということは消去ノードを隣接ノードで表すということである。このことからノードを消去する場合要素数を増やさない接続状態を考えてみることにする。

i) 消去ノードのブランチの接続数が1つの場合

いわゆる端点の場合である。発電機は系統の接続上端点となる。端点を消去する場合フィルインは発生しない。図2では、ノード縮約により非ゼロ要素数は4から1に減っている。

図 2: ブランチの接続数が1のノード

Fig.2. A node connected with one branch

ii) 消去ノードのブランチ接続数が2の場合（ただし、消去ノードの隣接ノード間には接続がないものと仮定する。）

図3ではフィルインブランチが1つ発生しているが、縮約行列の非ゼロ要素数は4となっており、元の行列より3つ減っている。

iii) 消去ノードのブランチ接続数が3の場合（ただし、消去ノードの隣接ノード間には接続がないものと仮定する。）

図 3: ブランチの接続数が2のノード
Fig.3. A node connected with two branches

図 4 では、フィルインブランチが3つ発生しているが、縮約行列の非ゼロ要素数は9であり、元の行列より非ゼロ要素数は1つ減っていることがわかる。

図 4: ブランチの接続数が3のノード
Fig.4. A node connected with three branches

iv) 消去ノードのブランチ接続数が4の場合（ただし、消去ノードの隣接ノード間には接続がないものと仮定する。）

図 5 において、ノード 1 を消去するとフィルインブランチは6本発生し、縮約行列の非ゼロ要素数は、16となる。一方縮約前の行列の非ゼロ要素数は13であり、縮約後の非ゼロ要素数は、3つ増加するのがわかる。

以上の考察から、演算量を削減させるノードのオーダリング手段として、次のアルゴリズムを考えることができる。

提案法 I

- I-(1) オーダリングするノードの集合から発電機ノードを外しておく。
- I-(2) T 2 によって消去ノードを選びそのノードを消去する。
- I-(3) I-(2) で作成した縮約グラフにおいて、
 - I-i) ノードを消去してできる縮約グラフで、ノードの最小の接続ブランチ数が4以上である。

図 5: ブランチの接続数が4のノード
Fig.5. A node connected with four branches

I-ii) 残りのノードがすべて発電機ノードである。
注 1

I-i), I-ii) のどちらかの条件を満たせば、I-(4) に進む。そうでなかったら I-(2) に戻る。

I-(4) 発電機ノードを加えた残りのノードに対して T 2 を用いてオーダリングを行う。

(注 1: 発電機が4台以上ある場合 I-ii) の条件は不要となる。)

提案法 I では消去ノードの隣接ノード間には接続がないものと仮定したが、実際の縮約グラフでは接続があり、これを考慮すればさらに演算量を削減させることが期待できる。そこで、提案法 I をさらに改善した提案法 II を考える。

その前に必要な記号の整理をしておく。

$Adj(m) = \{m_i\}_{i=1 \dots n}$: ノード m の隣接ノードの集合
 $n = |Adj(m)|$: ノード m の隣接ノードの個数
 $\cap$: 積集合
 $\cup$: 和集合

まずノード m を消去することを考える。

第 m 行第 m 列の非ゼロ要素数は消去過程の縮約グラフで、

$$N_1 = 2|Adj(m)| + 1 \quad (8)$$

となる。

また、 m を消去することによって生じるフィルイン数は $m_i (i = 1 \sim n)$ を m の隣接ノードとすると、

$$l_i = \sum_{j \neq i} |\{m_i\} \cap Adj(m_j)| \quad (9)$$

とおくと、

$$N_2 = \sum_{i=1}^n |n - 1 - l_i| \quad (10)$$

で与えられる。

例えば、隣接ノード同士に結合がない場合は、

$$N_2 = n \cdot (n - 1) \quad (11)$$

となる。

従って、ノード消去に伴う縮約行列での非ゼロ要素数の増減数は、

$$N_I = N_2 - N_1 \quad (12)$$

で与えられることになる。図6の場合を例にとれば、

$$Adj(1) = \{2, 3, 4\}$$

$$n = |Adj(1)| = 3$$

$$N_1 = 2 \times n + 1 = 7$$

$$l_1 = |\{2\} \cap \{1\}| + |\{2\} \cap (\{2\} \cup \{1\})| = 1$$

$$l_2 = |\{3\} \cap (\{1\} \cup \{4\})| + |\{3\} \cap (\{1\} \cup \{4\})| = 0$$

$$l_3 = |\{4\} \cap (\{1\} \cup \{4\})| + |\{4\} \cap \{1\}| = 1$$

$$N_2 = |3-1-1| + |3-1-0| + |3-1-1| = 4$$

$$N_I = 4 - 7 = -3$$

となる。

図6: ブランチの接続数が3つのノード

Fig.6. A node connected with three branches

この N_I を指標として提案法IIを考える。

提案法II

- II-(1) オーダリングするノードの集合から発電機ノードを外しておく。
- II-(2) T 3によって消去ノードを選び、そのノードを消去する。注2
- II-(3) II-(2)で作成した縮約グラフにおいて、
 - II-i) 全てのノードで指標 N_I が正である。
 - II-ii) 残りのノードが全て発電機である。
 以上の2つの条件のうち、どちらかを満たせばII-(4)へ進む。そうでなかったら、II-(2)へ戻る。
- II-(4) 発電機ノードを加えた残りのノードに対してT 3を用いてオーダリングを行う。

(注2: T 3は縮約グラフで、ノードを消去した時発生するフィルインブランチの数が最小のノードを次の消去ノードとして選ぶ手法である。)

提案法IIでT 3を用いたのは、II-i)を計算するためである。

提案法I, IIではオーダリング手法として分解パス^[2]の長さを考慮しないT 2やT 3を用いているが、これは、計算に不要な部分は分解パスの長さは長くしても構わないという方針に基づくものである。従って、これらのアルゴリズムの代わりにMD-ML^[10]等分解パスの長さを短くするアル

ゴリズムを選んで良いがその効果は薄いものと考えられる。

全体のノード数を N_D 、発電機の台数を N_G 、縮約するノードの数を N_A とする。 N_A は発電機ノードを加える前までオーダリングしたノード数である。また、ブランチの数を N_B 、フィルインブランチ数を N_{FL} とおく。

オーダリングした行列は、

$$\begin{bmatrix} Y_{L'L'} & Y_{L'G'} \\ Y_{G'L'} & Y_{G'G'} \end{bmatrix} \begin{bmatrix} V_{L'} \\ V_{G'} \end{bmatrix} = \begin{bmatrix} 0 \\ I_{G'} \end{bmatrix} \quad (13)$$

L' : 負荷, 浮遊ノードの部分集合 次元 = N_A

G' : 発電機ノード+負荷, 浮遊ノード 次元 = $N_D - N_A$
となる。ここで、演算量を削減する縮約Y行列は、

$$Y_{G'} = Y_{G'G'} - Y_{G'L'} \cdot Y_{L'L'}^{-1} \cdot Y_{L'G'} \quad (14)$$

となる。

$Y_{G'}$ の次元 $N_D - N_A$ は、発電機間縮約Y行列 Y_G の次元 N_G より多くなっている。 $Y_{G'}$ には、発電機以外の負荷や浮遊ノードが含まれているが、これがフィルイン要素を増やさないための要因であると考えられる。

LU分解後の元のY行列の非ゼロ要素数は、

$$N_{Z0} = N_D + 2N_B + 2N_{FL} \quad (15)$$

で与えられる。

次に、発電機間縮約Y行列の非ゼロ要素数は、完全に密行列になるのでその非ゼロ要素数は、

$$N_{Z1} = N_G \cdot N_G \quad (16)$$

となる。

系統が大きくなると、前述の通り N_{Z1} は N_{Z0} より完全に多くなり、

$$N_{Z1} \gg N_{Z0} \quad (17)$$

となる。

次に、演算量を削減する縮約Y行列の $Y_{G'}$ の非ゼロ要素数 N_{Z2} は、元のY行列の非ゼロ要素数 N_{Z0} よりかなり小さくなっているものと考えられる。すなわち、

$$N_{Z2} \ll N_{Z0} \quad (18)$$

が成立する。これは、縮約Y行列を作成するオーダリングの手順から見て明らかである。

次に、発電機の電圧計算について考えてみよう。ここで、負荷, 浮遊ノードの電圧計算は不要とする。

従来の手法では、(1)式をそのまま解けば良い。注入電流ベクトル I のスパース性を利用したスパースベクトル法を用いればさらに計算量を減らすことが期待できる。

縮約Y行列の発電機電圧計算の演算手順は以下のように行えば良い。行列のLU分解は従来法と全く共通である。前進後退代入については、下の $V_{G'}$, $I_{G'}$ だけに対して行えば良い。これをスパースベクトル法のアルゴリズムでいえば0ベクトル($I_{L'} = 0$)を初めの部分にまとめて持っていたことに相当する。このオーダリングはスパースベクトル法の性能を最大限に引き出すものと考えられる。

ここで、提案法Iの効果を図7-1に示す簡単な系統(ノード数22, ブランチ数22, 発電機数5)を用いて確かめる。

図 7-1: T 2 によるオーダリング
Fig.7-1. Ordering of T2

図 7-2: T 2 による行列の非ゼロ構造
Fig.7-2. Nonzero structure of T2

従来法の T 2 を用いて計算した結果では、図 7-2 のようにフィルインは 2 個発生しており、全体の非ゼロ要素数は 68 になっている。

これに対し、提案法 I では図 8-1 のようなオーダリングを行ない、図 8-2 に示す通りフィルインは 28 個発生し全体の非ゼロ要素数は 92 になっている。しかし、実際に計算に必要とする縮約 Y 行列の非ゼロ要素数はわずか 18 個（図 8-2 の正方形で囲んだ部分）であり、これは従来法の 0.26 倍に過ぎず提案法の効果が大きいことが確認できる。図 8-4 に演算量を最小にする縮約 Y 行列を示す。この Y 行列には発電機以外のノード D が含まれており、これが演算量の増加を防いでいる。一方、図 8-5 に示すように D を消去した発電機間縮約 Y 行列は完全に密行列となっている。

図 7-3: T 2 による分解パス
Fig.7-3. Factorization path of T2

図 8-1: 提案法 I によるオーダリング
Fig.8-1. Ordering of proposed method I

図 8-2: 提案法 I による行列の非ゼロ構造
Fig.8-2. Nonzero structure of proposed method I

また、提案法の効果は、分解パスからみればより一層明確になる。図 7-3 に示す通り従来のオーダリング手法（T 1 ~ T 3, MD-ML 法等）では発電機ノードはそのブランチの接続数が 1 であるので、分解パス上では長さが 1 のレベルにあり、これが演算量を増やす原因となる。これに対し提案法では図 8-3 に示す通り発電機ノードはレベル 1 には 1 つもなくそれより下位のレベルにある。従って、発電機ノードより上のレベルのノードの計算は省略できることになる。これは、ノード縮約されたノードが上位のレベルにあることと等価である。

スパースベクトル法を用いた計算では、従来法では S, T, U, R のノードの計算が不要となる。提案法では、M, L, N

図 8-3: 提案法 I による分解パス
Fig.8-3. Factorization path of proposed method I

図 8-4: 演算量を最小にする
縮約Y行列
Fig.8-4. Reduced Y matrix
minimizing operations

図 8-5: 発電機間
縮約Y行列
Fig.8-5. Reduced
G-G Y matrix

より上位のノードの計算は不要であり提案法の効果がうかがえる。

なお、この例題では提案法IとIIでは同じ結果を与える。

4 テスト計算及び考察

今回提案した手法の効果をテスト計算によって確かめる。

ここでは、発電機ノードの電圧を計算の対象として、以下の4種類の手法についてテスト計算を行なった。

- 1) 従来法 T 2
- 2) スパースベクトル法 MD-ML(Minimum Degree Minimum Length): MD-MLはスパースベクトル法の性能を引き出すオーダリング手法の中で最も優れたものの1つであるといわれている。
- 3) 提案法I
- 4) 提案法II

計算結果を表1～表3に示す。

演算量はLU分解後の前進後退代入の乗除算数で評価した。系統計算では、Y行列は一定のままで、右辺ベクトルがステップ毎に変わり、前進後退代入が計算時間のほとんどを占め、それは乗除算の数にほぼ比例すると考えられるからである。なお、表の演算要素とは乗除算の数である。

表 1: ケース 1 ノード数76 ブランチ数92 発電機数13

Table 1. Case 1

	行列の非0要素数 (フィルイン数)	縮約Y行列 の次元 *	演算要素	演算比 **
1) 従来法 T 2	308 (60)	76	308	1
2) SV 法	308 (60)	42	168	0.55
3) 提案法 I	354 (106)	24	102	0.33
4) 提案法 II	398 (150)	18	84	0.27

* スパースベクトル法では演算対象ノード数

** 演算比 縮約Y行列の非0要素数 / 従来法のY行列の非0要素数

表 2: ケース 2 ノード数307 ブランチ数350 発電機数64

Table 2. Case 2

	行列の非0要素数 (フィルイン数)	縮約Y行列 の次元 *	演算要素	演算比 **
1) 従来法 T 2	1201 (196)	307	1201	1
2) SV 法	1199 (194)	215	811	0.68
3) 提案法 I	1465 (460)	110	484	0.40
4) 提案法 II	1503 (498)	102	456	0.38

* スパースベクトル法では演算対象ノード数

** 演算比 縮約Y行列の非0要素数 / 従来法のY行列の非0要素数

表 3: ケース 3 ノード数983 ブランチ数1091 発電機数142

Table 3. Case 3

	行列の非0要素数 (フィルイン数)	縮約Y行列 の次元 *	演算要素	演算比 **
1) 従来法 T 2	3505 (362)	983	3505	1
2) SV 法	3465 (322)	448	1482	0.42
3) 提案法 I	4105 (962)	211	797	0.23
4) 提案法 II	4159 (1016)	202	774	0.22

* スパースベクトル法では演算対象ノード数

** 演算比 縮約Y行列の非0要素数 / 従来法のY行列の非0要素数

Y行列のLU分解後、提案法のフィルイン数はT 2やMD-MLに比べ、2～3倍になっており、全体の非ゼロ要素数は1.1～1.2倍に増えている。従って、提案法では最初のLU分解で従来法より若干時間がかかることになる。

しかし、このフィルイン要素は計算不要なものであり、前進後退代入で計算量を多くすることはない。

ケース1～ケース3のいずれも提案法が従来法と比較して演算量が少ないことを示しており、演算量は従来法の0.2～0.4倍程度に削減されていることがわかる。提案法Iと提案法IIでは提案法IIの方が縮約Y行列の次元を小さくでき、その分非ゼロ要素数も少なくなっており、性能が勝っていると考えられる。しかし、その効果はほぼ同程度と考えられ、実用では提案法I(簡略法)で十分と思われる。

なお、提案法はスパースベクトル法(MD-ML)に比較しても性能が1.5～2倍程度勝っている。これはスパースベクトル法のオーダリング手法では分解パスの長さの平均値を最小にするようにしているが、今回の縮約Y行列は必要な計算部分のパスの総計を最小にしていると考えられるからである。

5 まとめ

今回提案した縮約アドミタンス行列法は行列の次元を縮小するとともに演算量も少なくする。その効果を大規模系統(ケース3)によって確かめたところ、提案法IIでは従来法の0.22倍の演算量になっている。また本手法はスパースベクトル法(MD-ML)に対しても勝っており、演算比はその0.52倍に短縮している。

提案法は、別の観点から見ればスパースベクトル法の性能を最大限に引き出すオーダリング手法とも考えることができる。

提案法は発電機の電圧計算や発電機間縮約インピーダンス行列の計算のみならず、Y行列を用いた計算の高速化に適用でき、時間ループや収束計算のループで右辺ベクトルが毎回変わり、その回数が多い場合特に有効であると考えられる。今後は電力系統動特性解析や固有値解析等への適用が考えられる。

(平成7年9月22日受付、同8年1月26日再受付)

文 献

- (1) W. F. Tinney, J. W. Walker, "Direct Solution of Sparse Network Equation by Optimally Ordered Triangular Factorization", IEEE Proc, **55**, 1967
- (2) W. F. Tinney, V. Brandwajn etc., "Sparse Vector Method", IEEE Trans on PAS, **104**, No.2, 1985
- (3) A. Abur, "A Parallel Scheme for Forward/Backward Substitution in Solving Sparse Linear Equation", IEEE Trans on PS, **3**, No. 4, 1988
- (4) A. Padiha, A. Morelato, "A W-Matrix Methodology for Solving Sparse Network Equations on Multiprocessor Computers", IEEE Trans on PS, **7**, No. 3, 1992
- (5) M. K. Enns, W. F. Tinney, F. L. Alverado "Sparse Matrix Inverse Factors", IEEE Trans on PS, **5**, No.2, 1990
- (6) F. L. Alvarado, D. C. Yu "Partitioned Sparse A^{-1} Methods", IEEE Trans on PS, **5**, No.2, 1990
- (7) S. M. Chan, V. Brandwajn etc. "Partial Matrix Refactorization", IEEE Trans on PWRS, **11**, No.1, 1986
- (8) A. Gomez, L. G. Franquelo "Node Ordering Algorithm for Sparse Vector Method Improvement", IEEE Trans on PS, **3**, No.1, 1988
- (9) A. Gomez, L. G. Franquelo "An Efficient Ordering Algorithm to Improve Sparse Vector Method", IEEE Trans on PS, **3**, No.4, 1988
- (10) R. Betancourt "An Efficient Heuristic Ordering Algorithm for Partial Matrix Refactorization", IEEE Trans on PS, **3**, No. 3, 1988
- (11) C. L. Lee, N. Chen "Sparse Vector Method Improvements via Minimum Inverse Fill-in Ordering", IEEE Trans on PS, **7**, No.3, 1992
- (12) 齊藤 他 "スパースベクトル法を適用した発電機間縮約インピーダンス行列の効率的計算法", 電力技術研究会, PE-95-13, 1995

的 場 誠 一

(正員) 1979年3月電気通信大学電気通信学部卒業。同年4月(株)開発計算センター入社。主として電力系統解析に係る業務に従事。

周 小 明

(正員) 1983年7月北京師範大学数学学部卒業。北京経済学院大学経済情報学部助手を経て、1993年(株)開発計算センター入社。主として電力系統解析に係る業務に従事。

関 田 昌 弘

(正員) 1981年3月東京大学工学部電気工学科卒業。同年電源開発(株)入社。主に電力系統の計画、解析業務に従事。同社工務部系統技術課課長代理。

大 石 孝 穂

(正員) 1983年3月佐賀大学理工学部電気工学科卒業。1991年(株)開発計算センター入社。1993年電源開発(株)出向。現在、電源開発(株)工務部系統技術課において主に電力系統の解析業務に従事。