

N-R 法在潮流计算中的应用

3.1 N-R 法的原理

牛顿-拉夫逊方法(简称 N-R 法)是牛顿 17 世纪提出的一种在实数域和复数域上近似求解方程的方法,其实质是局部线性化。由于多数方程不存在求根公式,因此求精确根非常困难,甚至不可能,从而寻找方程的近似根就显得特别重要。该方法使用函数 $f(x)$ 的泰勒级数的前面几项来寻找方程 $f(x) = 0$ 的根。

牛顿-拉夫逊法的基本原理为假设初始解为 $x^{(0)}$, 其与真实解的误差为 $\Delta x^{(0)}$, 真实解即为 $x^{(0)} + \Delta x^{(0)}$, 将非线性方程 $f(x) = 0$ 在 $x^{(0)} + \Delta x^{(0)}$ 处进行泰勒展开并略去高次项, 得到:

$$f(x^{(0)}) + f'(x^{(0)})\Delta x^{(0)} \approx 0 \quad (3-1)$$

即

$$\Delta x^{(0)} \approx -\frac{f(x^{(0)})}{f'(x^{(0)})} \quad (3-2)$$

用得到的修正量 $\Delta x^{(0)}$ 进行修正:

$$x^{(1)} = x^{(0)} + \Delta x^{(0)} \quad (3-3)$$

并用修正后 x 进行反复迭代, 直到满足 $|f(x^{(k)})| < \epsilon_1$ 或 $|\Delta x^{(k)}| < \epsilon_2$ 停止迭代, 从而得到具有一定精度的近似解。

将上述方法从一维非线性方程推广到高维情况即可用来求解系统潮流。对于非线性方程组

$$\begin{cases} f_1(x_1^{(0)} + \Delta x_1, x_2^{(0)} + \Delta x_2, \dots, x_n^{(0)} + \Delta x_n) = 0 \\ f_2(x_1^{(0)} + \Delta x_1, x_2^{(0)} + \Delta x_2, \dots, x_n^{(0)} + \Delta x_n) = 0 \\ \vdots \\ f_n(x_1^{(0)} + \Delta x_1, x_2^{(0)} + \Delta x_2, \dots, x_n^{(0)} + \Delta x_n) = 0 \end{cases} \quad (3-4)$$

有

$$\begin{bmatrix} f_1(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}) \\ f_2(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}) \\ \vdots \\ f_n(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}) \end{bmatrix} = - \begin{bmatrix} \left. \frac{\partial f_1}{\partial x_1} \right|_0 & \left. \frac{\partial f_1}{\partial x_2} \right|_0 & \dots & \left. \frac{\partial f_1}{\partial x_n} \right|_0 \\ \left. \frac{\partial f_2}{\partial x_1} \right|_0 & \left. \frac{\partial f_2}{\partial x_2} \right|_0 & \dots & \left. \frac{\partial f_2}{\partial x_n} \right|_0 \\ \vdots & \vdots & & \vdots \\ \left. \frac{\partial f_n}{\partial x_1} \right|_0 & \left. \frac{\partial f_n}{\partial x_2} \right|_0 & \dots & \left. \frac{\partial f_n}{\partial x_n} \right|_0 \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \vdots \\ \Delta x_n \end{bmatrix} \quad (3-5)$$

即在潮流计算中的牛顿-拉夫逊法有直角坐标系和极坐标系两种形式, 其中直角坐标系形式为

$$\begin{bmatrix} \Delta P_1 \\ \Delta Q_1 \\ \Delta P_2 \\ \Delta Q_2 \\ \vdots \\ \Delta P_p \\ \Delta U_p^2 \\ \Delta P_n \\ \Delta U_n^2 \end{bmatrix} = \begin{bmatrix} H_{11} & N_{11} & H_{12} & N_{12} & H_{1p} & N_{1p} & H_{1n} & N_{1n} \\ J_{11} & L_{11} & J_{1p} & L_{1p} & J_{1p} & L_{1p} & J_{1n} & L_{1n} \\ H_{21} & N_{21} & H_{22} & N_{22} & H_{2p} & N_{2p} & H_{2n} & N_{2n} \\ J_{21} & L_{21} & J_{22} & L_{22} & J_{2p} & L_{2p} & J_{2n} & L_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ H_{p1} & N_{p1} & H_{p2} & N_{p2} & H_{pp} & N_{pp} & H_{pn} & N_{pn} \\ R_{p1} & S_{p1} & R_{p2} & S_{p2} & R_{pp} & S_{pp} & R_{pn} & S_{pn} \\ H_{n1} & N_{n1} & H_{n2} & N_{n2} & H_{np} & N_{np} & H_{nn} & N_{nn} \\ R_{n1} & S_{n1} & R_{n2} & S_{n2} & R_{np} & S_{np} & R_{nn} & S_{nn} \end{bmatrix} \begin{bmatrix} \Delta e_1 \\ \Delta f_1 \\ \Delta e_2 \\ \Delta f_2 \\ \vdots \\ \Delta e_p \\ \Delta f_p \\ \Delta e_n \\ \Delta f_n \end{bmatrix} \quad (3-6)$$

极坐标系形式为

$$\begin{bmatrix} \Delta P_1 \\ \Delta Q_1 \\ \Delta P_2 \\ \Delta Q_2 \\ \vdots \\ \Delta P_p \\ \vdots \\ \Delta P_n \end{bmatrix} = \begin{bmatrix} H_{11} & N_{11} & H_{12} & N_{12} & H_{1p} & \cdots & H_{1n} \\ J_{11} & L_{11} & J_{12} & L_{12} & J_{1p} & \cdots & J_{1n} \\ H_{21} & N_{21} & H_{22} & N_{22} & H_{2p} & \cdots & H_{2n} \\ J_{21} & L_{21} & J_{22} & L_{22} & J_{2p} & \cdots & J_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ H_{p1} & N_{p1} & H_{p2} & N_{p2} & H_{pp} & \cdots & H_{pn} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ H_{n1} & N_{n1} & H_{n2} & N_{n2} & H_{np} & \cdots & H_{nn} \end{bmatrix} \begin{bmatrix} \Delta \delta_1 \\ \frac{\Delta U_1}{U_1} \\ \Delta \delta_2 \\ \frac{\Delta U_2}{U_2} \\ \vdots \\ \Delta \delta_p \\ \vdots \\ \Delta \delta_n \end{bmatrix} \quad (3-7)$$

直角坐标系的优点在于迭代过程中不需要进行有关三角函数的计算,迭代速度快;而极坐标对 PV 节点的处理比较方便,相比直角坐标而言未知量少,因而求解的维数也要低。

3.2 N-R 法的特点

潮流计算的基本要求主要有三方面,即:

- ① 可靠性、收敛性;
- ② 计算速度和内存要求;
- ③ 方便性和灵活性。

N-R 法的优点是二次收敛,迭代次数少,收敛速度快,可靠性高。

而 N-R 法的缺点是:

- ① 每迭代一次要更新雅可比矩阵,对内存要求高;
- ② 对初始解要求较高。

对于正常运行系统而言,因为各节点的电压相角差不会很大,平启动就是很好的初始解,而且可以迅速收敛得到结果,这也是 N-R 法在稳态潮流计算中应用广泛的直接原因。

3.3 N-R 法的局限性

随着电力系统规模的发展,出现了给定初始条件,而潮流计算得不到有实际意义解的病

态潮流现象,而病态潮流问题实际上就是潮流算法收敛性的问题,常见的病态条件如系统重负荷导等。

N-R 法由于对初始解要求较高,因而在一些情况下无法求解潮流。常用的方法是用对初始解要求不高的 G-S 法先行求解得到相对准确的初始解再用 N-R 法进行求解。另外,为了解决病态潮流问题,非线性规划法也被提出来,其优势是在算法原理上就是收敛的,从而避免潮流计算的发散。

3.4 定雅克比矩阵法潮流计算

1. 直角坐标形式的定雅克比矩阵法

对直角坐标形式的牛顿-拉夫逊法的修正方程,做以下假设:

- ① 考虑到实际电力系统中,总有 $e_i \approx 1$ 、 $f_i \approx 0$,因此假设 $e_i = 1$ 、 $f_i = 0$ 。
- ② 考虑到雅克比矩阵的主要元素是其对角元,而实际影响最大的对角元中的 N_{ii} 、 M_{ii} 、 S_{ii} ,还可以对 H_{ii} 、 L_{ii} 进行简化。此时,考虑 a_i 相对小可以忽略。
- ③ 考虑到 b_i 很小可以忽略,还可以对 N_{ii} 、 M_{ii} 进行简化。

经过上述三个步骤的简化,直角坐标形式的牛顿-拉夫逊法的定雅克比算法的修正方程如下:

$$\begin{bmatrix} \Delta P_1 \\ \Delta Q_1 \\ \Delta P_2 \\ \Delta Q_2 \\ \vdots \\ \Delta P_i \\ \Delta U_i^2 \\ \vdots \\ \Delta P_{n-1} \\ \Delta U_{n-1}^2 \end{bmatrix} = - \begin{bmatrix} -G_{11} & B_{11} & -G_{12} & B_{12} & \cdots & -G_{1i} & B_{1i} & \cdots & -G_{1,n-1} & B_{1,n-1} \\ B_{11} & G_{11} & B_{12} & L_{12} & \cdots & B_{1i} & L_{1i} & \cdots & B_{1,n-1} & L_{1,n-1} \\ -G_{21} & B_{21} & -G_{22} & B_{22} & \cdots & -G_{2i} & B_{2i} & \cdots & -G_{2,n-1} & B_{2,n-1} \\ B_{21} & G_{21} & B_{22} & G_{22} & \cdots & B_{2i} & G_{2i} & \cdots & B_{2,n-1} & G_{2,n-1} \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots \\ -G_{i1} & B_{i1} & -G_{i2} & B_{i2} & \cdots & -G_{ii} & B_{ii} & \cdots & -G_{i,n-1} & B_{i,n-1} \\ 0 & 0 & 0 & 0 & \cdots & 0 & -2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots \\ -G_{n-1,1} & B_{n-1,1} & -G_{n-1,2} & B_{n-1,2} & \cdots & -G_{n-1,i} & B_{n-1,i} & \cdots & -G_{n-1,n-1} & B_{n-1,n-1} \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & -2 \end{bmatrix} \begin{bmatrix} \Delta e_1 \\ \Delta f_1 \\ \Delta e_2 \\ \Delta f_2 \\ \vdots \\ \Delta e_i \\ \Delta f_i \\ \vdots \\ \Delta e_{n-1} \\ \Delta f_{n-1} \end{bmatrix} \quad (3-8)$$

2. 极坐标形式的定雅克比矩阵法

类似直角坐标形式,对于极坐标形式的牛顿-拉夫逊法的修正方程,做以下假设并简化:

- ① 考虑到实际电力系统中,总有 $U_i \approx 1$ 、 $\delta_{ij} \approx 0$,因此假设 $U_i = 1$ 、 $\delta_{ij} = 0$ 。
- ② 在节点电压用极坐标表示的雅克比矩阵中,其主要元素是其对角元,考虑到 $Q_i \approx$

$U_i^2 B_{ii}$, 可以对 H_{ii} 、 L_{ii} 进行简化。此时, 假定 Q_i 相对小可以忽略。

③ 此时若考虑到 P_i 很小可以忽略, 还可以对 N_{ii} 、 K_{ii} 进行简化。

经过上述三个步骤的简化, 极坐标形式的牛顿-拉夫逊法的定雅克比算法的雅克比矩阵各元素如下:

$$H_{ij} = -B_{ij}; \quad L_{ii} = B_{ij}; \quad N_{ij} = -G_{ij}; \quad K_{ii} = G_{ij} \tag{3-9}$$

通过以上分析可知, 此时的雅克比矩阵是常数矩阵, 只要在迭代开始阶段形成因子表可以连续使用。由于定雅克比矩阵是一种固定斜率的牛顿-拉夫逊法, 所以只有一阶收敛速度, 但由于每次迭代的时间缩短了, 所以总的收敛速度比标准牛顿-拉夫逊法还是大大加快了。

3. 算例分析

为了验证定雅克比矩阵法的有效性, 本节通过以 ward&Hale-6 节点系统、IEEE14 节点系统、IEEE30 节点系统和 IEEE118 节点系统为例分别进行对比计算, 给出潮流算法的计算结果。

图 3-1 为 ward&Hale-6 节点系统接线图, 该系统包含有 6 个节点、7 条支路, 支路参数见表 3-1。其中节点 1 和 4、节点 3 和 5 之间为变压器支路, 节点 1、3、4、5 为负荷节点, 6 号节点为平衡节点, 2 为 PV 节点。各节点初始参数如表 3-2 所示。如无特别说明, 该系统基准功率 100MVA。

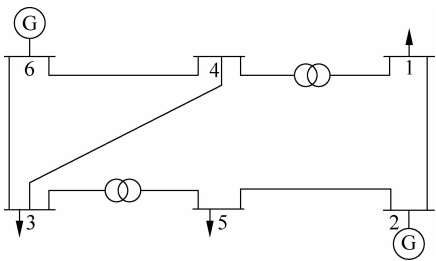


图 3-1 节点系统接线图

表 3-1 支路参数

支路编号	首末节点编号	支路类型	支路电阻	支路电抗	对地导纳	变压器变比
1	1~2	1	0.723	1.05	0	0
2	1~4	2	0	0.133	0	1.1
3	2~5	1	0.282	0.64	0	0
4	3~4	1	0.097	0.407	0	0
5	5~3	2	0	0.3	0	1.025
6	3~6	1	0.123	0.518	0	0
7	4~6	1	0.08	0.37	0	0

注[1]: 非标准变比侧在前;
[2]: 1、2、3 分别表示线路支路、阻抗支路和励磁支路。

表 3-2 6 节点潮流计算节点初试参数

节点编号	电压幅值	电压相角	发电有功	发电无功	负荷有功	负荷无功
1	1	0	0	0	0.55	0.13
2	1.1	0	0.501	0	0	0
3	1	0	0	0	0.5	0.05
4	1	0	0	0	0	0
5	1	0	0	0	0.3	0.18
6	1.05	0	0	0	0	0

采用 C++ 语言编程,编译环境为 VC++ 6.0,在 Inter(R)酷睿 2 双核 CPU2.20GHz 微机上运行(下面章节的运行环境是一样的,不再重复),节点初始潮流计算结果如表 3-3 所示。

表 3-3 6 节点潮流计算结果

节点编号	电压幅值	电压相角	发电有功	发电无功	负荷有功	负荷无功
1	0.8552	-13.815	0	0	0.55	0.13
2	1.1	-0.6095	0.501	0.3478	0	0
3	0.9332	-12.6355	0	0	0.5	0.05
4	0.9525	-9.9117	0	0	0	0
5	0.9008	-13.398	0	0	0.3	0.18
6	1.05	0	0.9652	0.3812	0	0
系统总功率			1.4662	0.729	1.35	0.36

为了节约篇幅,此处仅给出 IEEE14、IEEE30 和 IEEE118 的计算统计结果见表 3-4、表 3-5 和图 3-2。

表 3-4 各种潮流计算方法迭代次数

测试系统	直角 NR	极坐标 NR	直角 GJ	极坐标 FJ	PQ 分解法
IEEE006	5	5	9	14	6
IEEE014	5	4	9	7	9
IEEE030	5	4	8	7	15
IEEE118	10	9	40	43	28

表 3-5 各种潮流计算方法计算时间(单位:秒)

测试系统	直角 NR	极坐标 NR	直角 GJ	极坐标 FJ	PQ 分解法
IEEE006	0.001289	0.002023	0.002586	0.003974	0.003487
IEEE014	0.003044	0.006801	0.013185	0.007807	0.009858
IEEE030	0.015504	0.021232	0.036303	0.026713	0.03934
IEEE118	1.707125	0.923279	0.419559	0.450906	0.432161

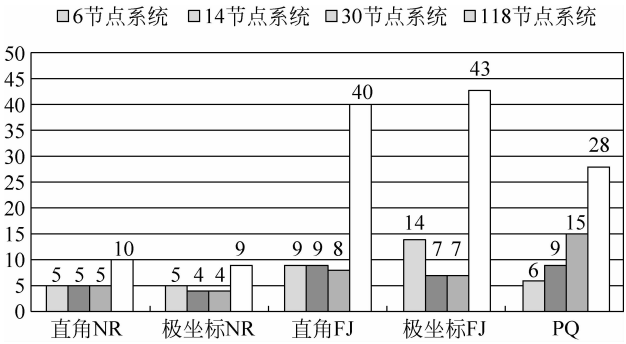


图 3-2 各算法迭代次数比较图

注[3]: NR 系指牛顿-拉夫逊法,FJ 系指定雅克比矩阵法;

[4]: 计算 118 节点时,考虑了加速因子的作用。

可以看出,在求解的迭代次数方面,NR 迭代次数无明显变化,即不同的系统都保持差不多的迭代次数,而定雅克比矩阵法和 PQ 分解法则在求解大系统时,迭代次数明显增多,总体上 $NR < PQ < FJ$ 。但在计算时间方面,求解小规模系统,NR 有一定的优势,但大系统时,PQ 分解法和定雅克比矩阵法优势明显,即在时间方面 $PQ < FJ < NR$,而运行时间是我们主要考虑的因素。

3.5 保留非线性的潮流计算方法

在直角坐标系下,节点功率平衡方程为

$$\Delta P_i = e_i \sum_{j=1}^n (G_{ij}e_j - B_{ij}f_j) - f_i \sum_{j=1}^n (G_{ij}f_j + B_{ij}e_j) - P_{is} = 0 \quad (3-10)$$

$$\Delta Q_i = f_i \sum_{j=1}^n (G_{ij}e_j - B_{ij}f_j) + e_i \sum_{j=1}^n (G_{ij}f_j + B_{ij}e_j) - Q_{is} = 0 \quad (3-11)$$

由式(3-10)和式(3-11)得到在直角坐标系下,潮流方程为二次代数方程组。

1. 潮流计算的矩阵表示

对于二次代数方程组,定义其真解为

$$\mathbf{x} \equiv (x_1, \dots, x_n)^T \quad (3-12)$$

真解对应的方程组的解为

$$\mathbf{y}_s \equiv \mathbf{y}(\mathbf{x}) \equiv (x_1, \dots, x_n)^T \quad (3-13)$$

常数矩阵为

$$\mathbf{A} \equiv \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n^2} \\ a_{21} & a_{22} & \cdots & a_{2n^2} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn^2} \end{bmatrix} \quad (3-14)$$

则二次代数方程组所代表的潮流方程可用式(3-15)表示

$$[\mathbf{y}_s] = [\mathbf{A}] \begin{bmatrix} x_1 x_1 \\ x_1 x_2 \\ \vdots \\ x_i x_j \\ x_i x_{j+1} \\ \vdots \\ x_n x_n \end{bmatrix} \quad (3-15)$$

在式(3-15)中, $\mathbf{y}_s$ 为 PQ 节点的给定 $\mathbf{P}$ 和 $\mathbf{Q}$,以及 PV 节点的 $\mathbf{P}$ 和 $|V|^2$, $\mathbf{A}$ 由节点导纳矩阵中 $\mathbf{G}$ 和 $\mathbf{B}$ 组成, $\mathbf{x}$ 为 e 和 f 。

2. 二次方程组的泰勒级数展开形式

定义二次方程组的估计值为

$$\mathbf{x}_e \equiv (x_{e1}, x_{e2}, \dots, x_{en})^T \quad (3-16)$$

真解和估计值之间的差为

$$\Delta x \equiv (\Delta x_1, \Delta x_2, \dots, \Delta x_n)^T \quad (3-17)$$

$$\Delta x = x - x_e \quad (3-18)$$

定义雅可比矩阵 $\mathbf{J}$ 和二阶偏导数矩阵 $\mathbf{H}$ 为

$$\mathbf{J} \equiv \left[\begin{array}{ccc} \frac{\partial y_1}{\partial x_1} & \dots & \frac{\partial y_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial y_n}{\partial x_1} & \dots & \frac{\partial y_n}{\partial x_n} \end{array} \right]_{x=x_e} \quad (3-19)$$

$$\mathbf{H} = \left[\begin{array}{cccc} \frac{\partial^2 y_1}{\partial x_1 \partial x_1} & \frac{\partial^2 y_1}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 y_1}{\partial x_1 \partial x_n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial^2 y_n}{\partial x_n \partial x_1} & \frac{\partial^2 y_n}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 y_n}{\partial x_n \partial x_n} \end{array} \right] \quad (3-20)$$

式(3-15)的泰勒展开可表示为

$$[\mathbf{y}_s] = [\mathbf{y}(x_e)] + [\mathbf{J}][\Delta \mathbf{x}] + \frac{1}{2}[\mathbf{H}] \begin{bmatrix} x_1 x_1 \\ x_1 x_2 \\ \vdots \\ x_i x_j \\ x_i x_{j+1} \\ \vdots \\ x_n x_n \end{bmatrix} \quad (3-21)$$

式(3-21)是式(3-15)的泰勒展开的准确表达式,因为式(3-15)是二次方程组,其三阶及以上的导数为零。由于

$$[\mathbf{y}(\Delta x)] = \frac{1}{2}[\mathbf{H}] \begin{bmatrix} x_1 x_1 \\ x_1 x_2 \\ \vdots \\ x_i x_j \\ x_i x_{j+1} \\ \vdots \\ x_n x_n \end{bmatrix} \quad (3-22)$$

所以式(3-21)也可以表示为

$$[\mathbf{y}_s] = [\mathbf{y}(x_e)] + [\mathbf{J}][\Delta \mathbf{x}] + [\mathbf{y}(\Delta x)] \quad (3-23)$$

实际上,如果式(3-23)的右边只保留到第二项,则为牛顿法计算表达式。

3. 保留非线性的潮流计算方法

式(3-16)又可以表示为

$$\mathbf{J} \Delta \mathbf{x} = \mathbf{y}_s - \mathbf{y}(x_e) - \mathbf{y}(\Delta \mathbf{x}) \quad (3-24)$$

所以,其迭代方程式可以表示为

$$\mathbf{J} \Delta \mathbf{x}^{(k+1)} = \mathbf{y}_s - \mathbf{y}(x_e) - \mathbf{y}(\Delta \mathbf{x}^{(k)}) \quad (3-25)$$

上标 k 表示第 k 次迭代。

$$\text{定义} \quad \Delta \mathbf{y}^{(k)} = \mathbf{y}_s - \mathbf{y}(\mathbf{x}_e) - \mathbf{y}(\Delta \mathbf{x}^{(k)}) \quad (3-26)$$

所以最终的迭代表达式为

$$\mathbf{J} \Delta \mathbf{x}^{(k+1)} = \Delta \mathbf{y}^{(k)} \quad (3-27)$$

实际计算式(3-20),需要将矩阵 $\mathbf{J}$ 三角分解,以形成上三角矩阵 $\mathbf{J}'$ 和新的列向量 $\Delta \mathbf{y}'^{(k)}$,表达式如下:

$$\begin{bmatrix} \bullet & \bullet & \bullet \\ 0 & \bullet & \bullet \\ 0 & 0 & \bullet \end{bmatrix} [\Delta \mathbf{x}^{(k+1)}] = [\Delta \mathbf{y}'^{(k)}], \quad \mathbf{J}' = \begin{bmatrix} \bullet & \bullet & \bullet \\ 0 & \bullet & \bullet \\ 0 & 0 & \bullet \end{bmatrix} \quad (3-28)$$

在每次迭代的过程中, $\Delta \mathbf{y}'$ 由于是 $\Delta \mathbf{x}^{(k)}$ 的函数而每次迭代都发生改变,而 $\mathbf{J}'$ 一直等于 $\mathbf{J}'^{(0)}$ 而不发生改变。因此,在潮流计算的初始化阶段,若将 $\mathbf{J}^{(0)}$ 三角分解,得到 $\mathbf{J}'$ 并存储起来,那么在每次迭代的过程中,仅仅需要改变 $\Delta \mathbf{y}'$,相比于牛顿法在每步都要修改雅可比矩阵,该方法计算量和存储量大大减小。保留非线性的潮流计算方法是定雅可比矩阵的潮流计算方法,计算流程如图 3-3 所示。

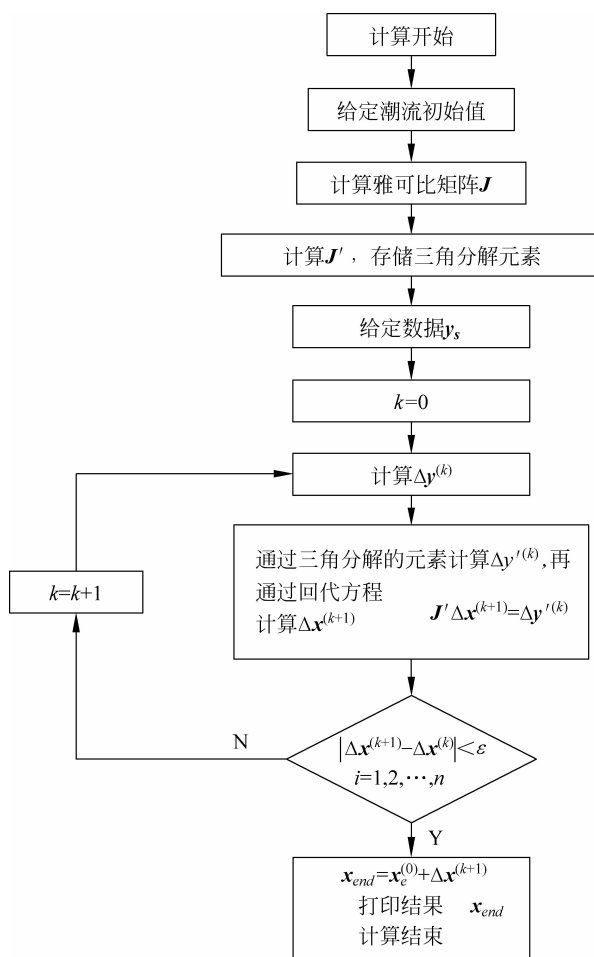


图 3-3 保留非线性的潮流计算流程图

4. 算例

方法	迭代次数		每次迭代时间 $\times 10^{-2}$ sec		每次迭代时间比		总迭代时间比	
	定雅可比	NR	定雅可比	NR	定雅可比	NR	定雅可比	NR
30 节点系统	4	3	17	95	1/5.5	1	1/4.1	1
57 节点系统	6	4	35	263	1/7.6	1	1/5.1	1
118 节点系统	6	4	70	833	1/11.9	1	1/7.9	1

算例取自文献[A fast load flow method retaining nonlinearity. S. IWAMOTO, Y. TAMURA. IEEE TRANS. ON POWER SYSTEM, VOL. PAS-97, No. 5, Sept/Oct 1978.]。