

# 一种改进前推回代法的配电网潮流计算

张捷, 徐焰, 张学飞

(国网上海市电力公司市北供电公司, 上海 200072)

**摘要:** 随着电网规模日益扩大, 对配电网的要求不断提高, 配电网潮流计算成为当前研究热点之一。针对传统前推回代法的局限性进行了一些改进。首先, 普通算法在进行迭代计算前必须按一定要求对节点进行编号, 而改进算法通过形成 3 个矩阵能够省略该步骤; 其次, 将变压器负荷换算到高压侧可以使配电网变为简单的单线图, 从而缩小系统规模; 此外, 用节点注入电流替代回代过程中的节点注入功率能省去支路损耗计算的时间; 最后, 通过 PV 节点的无功功率修正方程可以有效地处理含有 PV 节点的配电网, 解决了原方法无法处理配电网系统中 PV 节点的问题。基于 IEEE 33 节点配电网系统和某含有 PV 节点的 10 kV 系统的对比算例表明所提方法的有效性和可行性。

**关键词:** 配电系统; 潮流计算; 改进前推回代法; PV 节点

中图分类号: TM744 文献标志码: A 文章编号: 1003-6954(2020)03-0085-06

DOI:10.16527/j.cnki.cn51-1315/tm.2020.03.017

## Power Flow Calculation of Distribution Network Based on An Improved Forward/Backward Substitution Method

Zhang Jie, Xu Yan, Zhang Xuefei

(State Grid Shanghai Shibei Electric Power Supply Company, Shanghai 200072, China)

**Abstract:** With the increasing scale of power system, the requirements for the distribution network are constantly increasing, and the power flow calculation of distribution network has become a hot research topic. Aiming at the limitations of the traditional forward/backward substitution method, some improvements are proposed. Firstly, the ordinary method must number the nodes according to the certain requirements before performing the iterative calculation, but the improved one can omit this step by forming three matrices. Secondly, converting the transformer load to the high voltage side can make the distribution network a simple one-line diagram, thereby reducing the system scale. In addition, replacing the node injection power with the node injection current can save the branch loss calculation time. Finally, the reactive power correction equation of PV node can effectively process the distribution network with PV nodes, which solves the problem that the original method can not handle the PV nodes in distribution network system. The validity and feasibility of the proposed method are verified by the comparative simulations on IEEE 33 bus distribution network system and a 10 kV system with PV nodes.

**Key words:** distribution system; power flow calculation; pre-back/forward method; PV node

## 0 引言

目前的配电网潮流计算方法主要有改进的牛顿-拉夫逊法<sup>[1]</sup>、改进 PQ 解耦法<sup>[2]</sup>、回路阻抗法<sup>[3-4]</sup>、前推回代法<sup>[5-7]</sup>等。牛顿-拉夫逊法及改进 PQ 解耦法由于其自身原理的缺点, 直接应用于配电网潮流计算时, 收敛性较差且计算误差大。回路阻抗法主要应用于处理环网结构, 收敛性较好, 但其节点和支路编号处理复杂且计算量大。针对配电网辐射状结构的特点, 前推回代法是配电网潮流计

算最优算法之一, 收敛性好且计算速度快。传统的前推回代法在计算前需要对网络的节点、支路进行预编号, 可能导致计算错误, 因此文献 [5-6] 提出了一种自动编号法, 但该方法未考虑日益增加的分布式电源对配电网的影响。文献 [7] 指出传统前推回代法的无功功率取值可能与实际值出入较大, 将导致计算误差增大影响迭代结果, 因此需要将 PV 节点转化为 PQ 节点, 然后对无功和电压方程进行处理; 然而, 将所有 PV 节点转化为 PQ 节点又会导致潮流计算收敛性变差。

因此, 针对已有配电网潮流计算前推回代法的

不足,提出一种改进前推回代配电网潮流计算方法。该方法的贡献在于:提出节点自动编号方法,实现节点的正向遍历和反向遍历,提高计算速度;针对配电网的特殊结构,通过预处理消除近一半节点和支路,提升了算法计算效率;在前述节点自动编号法能区分出PQ节点、PV节点和平衡节点的基础上,判断PV节点是否能转换为PQ节点,并进行相应的特殊处理。

## 1 配电网前推回代法潮流计算原理

各类配电网潮流算法性能通常从以下几个方面进行分析:

1) 算法的收敛速度。改进的牛顿-拉夫逊法将非线性的方程通过一定变换转化为对相应线性方程的反复求解,且是二阶算法,具有平方收敛的特点,能在个位次数的迭代中迅速求解。

2) 稳定性。在配电网中,电力系统网络结构、线路参数及各种扰动因素对计算结果影响的程度即为稳定性。由于牛顿拉夫逊算法是二阶算法,且受到配电网R/X比值较高的影响而无法输出准确的计算结果。而前推回代法受到的影响则很小。

3) 算法的复杂程度。用简单原理的算法通常更可靠,由于前推回代法不需要计算节点导纳矩阵,其计算效率高。

综上所述,前推回代法的原理更符合实际配电网结构的要求,其处理环网结构方面的能力较弱;但配电网的主要特点就是运行时呈辐射状,因此并不会受到太大的影响。此外,该方法计算误差小、计算时间少,在系统异常时仍能保证输入有效结果,且收敛性能不会被配电网高R/X数值影响,目前已经广泛被用作计算配电网潮流的主要算法。针对已有方法的不足,下面做出了实用化的改进。

## 2 改进方法

以图1所示10 kV配电网系统为例介绍所提改进前推回代法的原理。图中,节点7为PV节点,各支路阻抗、节点负荷见表1。针对普通前推回代法的不足进行的改进主要包括节点编号处理、迭代节点规模简化处理、PV节点特殊处理3个部分。

### 2.1 节点编号的处理

首先,对节点进行分类,包括根节点、与根节点

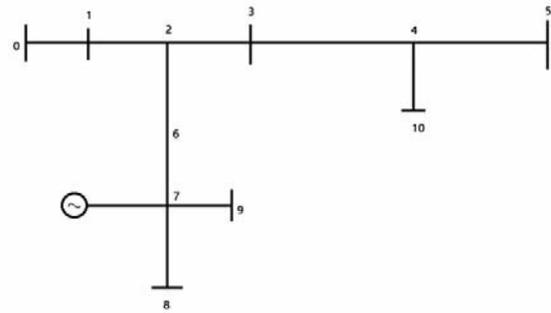


图1 10 kV 配电网简化图

表1 10 kV 配电网相关参数

父节点	子节点	支路阻抗		节点j负荷	
		R	X	P	Q
0	1	3.367	3.685	34.20	30.1
1	2	2.356	2.541	69.30	64.2
2	3	1.328	1.763	69.60	64.3
2	6	1.145	1.280	84.50	76.3
3	4	2.436	2.866	67.60	57.9
4	5	2.237	2.756	91.60	85.9
4	10	2.745	2.965	117.60	95.7
6	7(PV)	3.743	4.251	126.25	0.0
7(PV)	8	3.521	3.966	29.80	24.2
7(PV)	9	4.524	5.040	29.50	26.1

通过支路相连的子节点(根节点为该节点的父节点)、通过不同支路与不同的节点相连的兄弟节点。对节点遍历可形成一个数据表,该表记录了父节点的负荷、节点电压和节点类型(如PQ节点、PV节点)以及其子节点(集)的负荷、节点电压、节点类型等信息和两点之间支路的相关参数(支路阻抗、支路电流或功率),如表1所示。由于前推回代法需要首先对节点进行编号才能开始运算,所改进方法能够实现配电网节点的自动编号。当处理节点较多的配电网时,能提高计算速度;当遇到复杂配电网时不必进行额外运算,能自动识别PV节点。

下面以图1所示系统进行说明。

首先根据表1生成A矩阵和B矩阵。其中,A矩阵为a×b维矩阵,a为网络最末端子节点(叶节点)数目,在该系统中为节点5、8、9、10四个节点,即a=4;矩阵行表示从叶节点经过最短路径到达根节点所经过的节点号序列,形成的矩阵如表2所示。

B矩阵为c×d维矩阵,c为网络中节点数目,d为网络中父节点拥有最多子节点的节点数,其中d为2,形成的B矩阵如表3所示。

表2 A矩阵

叶节点	经过节点				
5	4	3	2	1	0
8	7	6	2	1	0
9	7	6	2	1	0
10	4	3	2	1	0

表3 B矩阵

节点	子节点	
0	1	-
1	2	-
2	3	6
3	4	-
4	5	10
5	-	-
6	7	-
7	8	9
8	-	-
9	-	-
10	-	-

其具体过程为: 1) 根据表1的网络参数图依次生成A矩阵和B矩阵; 2) 利用上述各矩阵配合传统前推回代法进行潮流计算, 在计算过程中生成一个 $c \times 2$ 的矩阵, 在运算完某一节点后自动将矩阵中对应的第2列标为1, 标1的节点将跳过计算, 以避免同一个节点重复计算。

综上所述, 通过节点自动编号提升了计算速度。

### 2.2 简化迭代节点的规模

针对配电网通常使用两绕组变压器, 其拓扑结构可等效为一条支路。此外, 为了配合目前已投入运行的线损计算程序, 在实用化处理中, 考虑参与迭代的节点只包含变压器支路的所有高压侧节点, 而不包含低压侧节点。因此, 相较于原方法, 可减少近一半计算节点和计算支路, 能提升计算效率。计算之前将变压器低压侧负荷换算成高压侧负荷, 其换算公式为:

$$P_1 = P_2 + \Delta P_0 \left(\frac{U_1}{U_N}\right)^2 + \Delta P_k \left(\frac{P_2^2 + Q_2^2}{S_N^2}\right) \left(\frac{U_N}{U_1}\right)^2 \quad (1)$$

$$Q_1 = Q_2 + \frac{I_0\%}{100} S_N \left(\frac{U_1}{U_N}\right)^2 + \frac{U_x\%}{100} S_N \left(\frac{P_2^2 + Q_2^2}{S_N^2}\right) \left(\frac{U_N}{U_1}\right)^2 \quad (2)$$

$$U_x\% = \sqrt{(U_s\%)^2 - (U_r\%)^2} = \sqrt{(U_s\%)^2 - \left(\frac{\Delta P_k \times 100}{S_N}\right)^2} \quad (3)$$

式中:  $P_1, Q_1$  分别为变压器高压侧的有功功率和无

功功率;  $P_2, Q_2$  分别为变压器低压侧的有功功率和无功功率;  $\Delta P_0, \Delta P_k$  分别为变压器铭牌参数表中的额定空载损耗和额定短路损耗;  $I_0\%, U_x\%$  分别为空载电流百分比和短路电压百分比;  $U_1, U_N$  和  $S_N$  分别为变压器高压侧的电压、额定电压和额定容量;  $U_s\%, U_r\%$  分别为当变压器中的电流值正好等于额定电流值时, 在电阻和电抗上的电压降百分比。

该实用化处理通过简单等效运算, 将低压侧负荷换算到高压侧, 大大减少了参与计算的节点数, 加快了计算速度。此外, 在算法回代过程中, 将节点注入功率替换为节点注入电流, 避免了在每次迭代时计算支路功率损耗, 提升了计算速度。

### 2.3 PV节点的特殊处理方法

由于传统的配电系统都是只有一个电源, 呈辐射状分布, 随着分布式电源(distributed generation, DG)技术飞速发展, 分布式电源的接入会对传统的配电系统造成了一定的影响。针对此问题, 目前主要的处理方法是将分布式电源视作PV节点, 再将其与并联电容器简化等效为PQ节点, 达到快速计算潮流的目的。

针对前述节点自动编号法能区分出PQ节点、PV节点和平衡节点, 提出采用一种特殊的处理PV节点的方法。该方法简洁且计算速度快, 具有一定的工程意义。其特点是: 当配电系统中出现多个电源共同供电时, 先判断该PV节点是否能被视为PQ节点参与计算, 或当其被视作PQ节点进行计算且误差在两个百分点内时, 则无需用到PV节点特殊处理法; 若不能视作PQ节点来进行计算, 则进行PV节点的特殊处理。其具体步骤如下:

1) 设定PV节点电源的无功出力初始值为0, 使其转换为PQ节点。

2) 通过潮流计算求得PV节点所连上层双亲节点(除平衡节点及根节点外, 其他任一节点都可以找到其父节点)的计算电压。

3) 根据图2, 由PV节点的双亲节点的计算电压、PV节点与其父节点连接支路的阻抗、PV节点的恒定电压及有功功率求得PV节点接入电源的无功出力的修正量为

$$Q_{PV} = \lambda(Q_2 - Q_d) = -\lambda\left(\frac{(U_{n1} - U_{n2})U_2 + P_2R}{x} + Q_d\right) \quad (4)$$

式中:  $\lambda$  为计算步长,  $-1 < \lambda < 1$ , 一般取值为0.1;

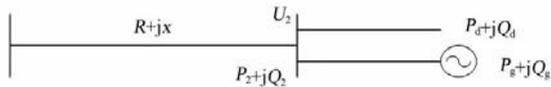


图2 PV节点无功功率迭代计算

$Q_2$ 、 $P_2$ 分别为PV节点的计算无功功率和有功功率;  
 $Q_a$ 为所有与PV节点连接的支路和节点的无功功率;  
 $U_{i1}$ 、 $U_{i2}$ 分别为PV节点及其父节点的节点电压总分量;  
 $U_2$ 为PV节点的输入电压; $R$ 、 $x$ 分别为PV节点与其父节点之间支路上的阻抗。

4) 若前推回代潮流计算收敛,则可得到PV节点的收敛无功;否则根据式(4)修正PV节点接入电源的无功出力,转步骤2)继续下次迭代计算。

### 3 仿真算例

将以10 kV系统和经典IEEE 33节点系统为例,分别计算前推回代算法和改进算法的潮流,通过对比两者的计算结果以验证改进算法的有效性和可行性。

#### 3.1 IEEE 33节点系统仿真

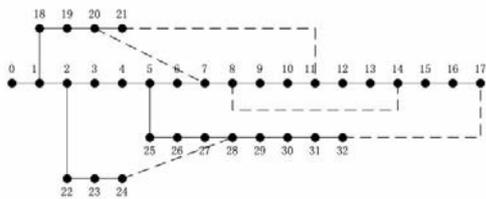


图3 IEEE 33节点系统

IEEE 33节点标准系统拓扑结构如图3所示,用Matlab仿真软件编程并进行潮流计算。分别以原方法与所提改进方法进行潮流计算,并给出网络电压分布与功率分布,其与实际结果的对比误差最大的前十和平均绝对误差值分别如表4—表9所示。

表4 IEEE 33电压误差(原方法)

节点 $i$	误差 / %
16	2.01
17	2.01
15	1.91
14	1.87
13	1.78
32	1.76
12	1.73
24	1.70
31	1.67
11	1.61
平均绝对误差	1.17

表5 IEEE 33有功功率误差(原方法)

节点 $i$	节点 $j$	误差 / %
3	4	-5.46
5	6	-2.98
4	5	-2.86
2	3	-2.78
1	2	-2.67
27	28	1.34
28	29	1.02
29	30	0.61
23	24	0.60
12	13	0.58
3	4	-5.46
平均绝对误差		0.8

表6 IEEE 33无功功率误差(原方法)

节点 $i$	节点 $j$	误差 / %
3	4	-3.53
0	1	2.27
8	9	1.96
7	8	1.89
9	10	1.59
6	7	1.58
27	28	1.52
23	24	1.33
22	23	-1.25
12	13	1.15
3	4	-3.53
平均绝对误差		1.77

表7 IEEE 33电压误差(改进算法)

节点 $i$	误差 / %
30	-0.06
31	-0.06
32	-0.06
28	-0.05
29	-0.05
9	-0.04
10	-0.04
11	-0.04
12	-0.04
13	-0.04
30	-0.06
平均绝对误差	0.02

表8 IEEE 33有功功率误差(改进算法)

节点 <i>i</i>	节点 <i>j</i>	误差/ <i>%</i>
1	2	2.67
2	3	-2.33
28	29	0.85
29	30	0.78
27	28	0.72
26	27	0.68
25	26	0.64
5	25	0.60
31	32	0.39
3	4	0.29
平均绝对误差		0.33

表9 IEEE 33无功功率误差(改进算法)

节点 <i>i</i>	节点 <i>j</i>	误差/ <i>%</i>
4	5	-1.70
2	22	-1.70
1	2	-1.66
6	7	1.60
31	32	-1.51
29	30	1.41
30	31	1.38
2	3	-1.34
3	4	-1.33
7	8	1.11
平均绝对误差		0.69

由表4—表6可知,用普通前推回代算法计算IEEE 33节点系统潮流时,节点电压幅值的误差大部分在1%~3%之间,误差较大的位置集中在节点6到节点17之间的线路上;而功率的误差大部分发生在如节点1、节点2这类有2个以上的子节点的节点上。由表7—表9可知,用所提改进算法计算IEEE 33节点系统潮流时,节点电压幅值的绝对平均误差为0.02%,远小于原方法的1.17%;有功功率分布的绝对平均误差为0.33%,略小于原方法的0.8%;节点电压幅值的绝对平均误差为0.69%,小于原方法的1.77%。因此,在计算相同配置网络下,所提方法具有更小的误差和更好的准确性。

### 3.2 10 kV 系统仿真

10 kV系统以前述配置分别以原方法与所提改进方法进行潮流计算,并给出网络电压分布与功率分布,其与实际结果的对比误差最大的前十和平均

绝对误差值分别如表10—表16所示。

表10 10 kV系统电压误差(原方法)

节点 <i>i</i>	误差/ <i>%</i>
0	0.00
1	0.94
2	3.13
3	4.08
4	2.09
5	3.48
6	5.39
7	4.62
8	2.79
9	5.61
10	3.62
平均绝对误差	3.25

表11 10 kV系统有功功率误差(原方法)

节点 <i>i</i>	节点 <i>j</i>	误差/ <i>%</i>
0	1	34.44
1	2	27.29
2	3	40.66
2	6	-36.05
3	4	12.42
4	5	-39.78
4	10	1.27
6	7	-17.33
7	8	21.97
7	9	28.28
平均绝对误差		25.95

表12 10 kV系统无功功率误差(原方法)

节点 <i>i</i>	节点 <i>j</i>	误差/ <i>%</i>
0	1	34.21
1	2	2.97
2	3	4.90
2	6	26.82
3	4	-97.38
4	5	-94.76
4	10	-194.08
6	7	100.00
7	8	24.73
7	9	31.26
平均绝对误差		61.11

表 13 10 kV 系统电压误差(改进方法)

节点 $i$	误差 /%
0	0.00
1	1.05
2	1.45
3	1.11
4	-0.35
5	-0.28
6	2.13
7	1.63
8	-1.29
9	-0.47
10	-1.95
平均绝对误差	1.06

表 14 10 kV 系统有功功率误差(改进方法)

节点 $i$	节点 $j$	误差 /%
0	1	3.57
1	2	2.34
2	3	5.28
2	6	-3.35
3	4	3.70
4	5	1.50
4	10	-3.75
6	7	2.51
7	8	-3.16
7	9	3.73
平均绝对误差		3.29

表 15 10 kV 系统无功功率误差(改进方法)

节点 $i$	节点 $j$	误差 /%
0	1	2.45
1	2	-3.78
2	3	4.28
2	6	3.13
3	4	-1.40
4	5	2.51
4	10	3.50
6	7	-1.95
7	8	3.58
7	9	-2.72
平均绝对误差		2.93

由表 10—表 12 可知,对于用普通前推回代算法计算 10 kV 系统潮流时,由于系统中有 PV 节点的存在,有功功率的平均绝对误差达到了 25.95%,无功功率绝对误差达到了 61.11%。因此,原算法已失去计算的有效性和可行性,需要对 PV 进行相应的特殊处理。由表 13—表 15 可知,用所提方法计算 10 kV 系统潮流时,电压幅值、有功功率分布、无

功率分布的绝对平均误差分别为 1.06%、3.29%、2.93%,均在可接受范围内。综上所述,相较于原方法,所提方法不仅具有更好的准确性,特别地,对于存在 PV 节点的系统,所提方法具有较好的应用。

## 4 结 语

对已有配电网潮流算法从几个方面进行对比,从而得出前推回代法符合配电网要求、计算结果精确的结论。然后,针对传统前推回代算法仍存在的局限性进行分析,从 3 个方面进行实用化处理:通过生成 3 个特殊矩阵减少编号时间;简化参与计算节点的数量;对 PV 节点采用无功修正计算并将其转化为 PQ 节点来计算。最后,在 Matlab 下用两个算例进行对比,可知:在 IEEE 33 节点系统中,改进方法的误差明显减小;在 10 kV 系统中,PV 节点的处理能将误差控制在  $\pm 3\%$  左右,比传统前推回代法的误差要小得多。因此,所提改进方法具有一定有效性和可行性。

### 参考文献

- [1] 李伟,刘友仁,陈恳,等.配电网潮流计算的实用算法[J].南昌大学学报(工科版),2011,33(3):303-306.
- [2] 柳永妍,乐健,高鹏,等.基于两点估计法的有源配电网概率潮流计算方法研究[J].电测与仪表,2017,54(5):1-8.
- [3] 刘耀年,岂小梅,李国鹏,等.基于回路阻抗法的配电网潮流计算[J].继电器,2004(8):8-10
- [4] 艾鹏,张雪霞,王晓茹.基于补偿算法改进的隐式 Z<sub>bus</sub>(bus)高斯潮流计算方法[J].电力系统保护与控制,2015,43(21):67-72.
- [5] 李欣然,李龙桂,童莹,等.适用于多电源配电网潮流计算的前推回代方法[J].电力系统及其自动化学报,2017,29(12):121-125.
- [6] 张小敏,林群,吴旭光,等.适用于低压配电网潮流计算的前推回代法[J].水电能源科学,2016,34(9):183-186.
- [7] 闫丽梅,谢一冰,徐建军,等.改进的前推回代法在含分布式电源配电网计算中的应用[J].西安交通大学学报,2013,47(6):117-123.

### 作者简介:

张捷(1974),工程师,从事电网调控运行工作。

(收稿日期:2020-01-01)