

多智能体一致性算法在微电网的应用

微电网由微电源、负荷、储能系统和控制装置等部分构成。在微电网中多个分布式电源、以及多个负载，以一定的拓扑结构组成一个网络，微电源和储能系统提供微电网能量的来源，以电力电子技术作为控制核心的控制装置负责电能的有效控制及转换。微电网是一个能够实现自我控制、保护和管理的自治系统，即可以与外部电网并网运行也可以孤立运行。它可以根据需要自由的选择由大电网进行接入，有效地促进了电网智能化地程度。

对于微电网的调度，以往采用传统的集中式优化策略，系统需要一个集中控制中心发布指令调度整个系统的所有单元包括发电机、电源等。但那时这种传统的集中式优化调度架构可能不能即使实现完美调度保证系统的完美运行。微电网中引入多智能体系统把分布式电源当成智能体，利用的是多智能体之间的通信所带来的信息交流，以解决传统控制策略智能利用本地信息的局限。这种方法的核心即：每个控制器利用自身的策略，本地进行控制，而各个控制器之间相互合作进而相互协调，即传递和接受杏林智能所传达的包括有功功率、无功功率值等信息，进而改变自身的控制策略，从而实现所处控制区域的总体控制目标。通过这种方法把微电网中的每个分布式电源视为智能体，将分布式电源的功率作为分布式电源的状态信息通过通信的方式进行分享，再利用一致性算法作为功率分配的分配算法。

多智能体一致性算法在微电网的应用与传统的集中式优化算

法相比，一致性算法仅要求每个智能体实时的获取本地与相邻智能体的信息，传输信息量小、优化时间段、也可以获得较为理想的收敛值。另外将其应用于实时功率分配问题也具有通信负担低、动态性能好等优势。

此文章主要分析多智能体一致性算法在微电网中的应用，而算法中数学模型的构建与推导主要与电力知识有关，所以本文中忽略数学模型的构建过程。本文直接给模型公式，而通过算发的角度去解释其中的原理与过程。

1、多智能体一致性算法关于微电网功率分配：

因为微电网中数据采集特点为离散数据，所以本算法采用离散时间一阶一致性算法。并且该算法应用的目的是减少微电网的调节成本去使得微电网中的功率分配趋于合理，所以以微电网的总调节成本作为目标函数。

数学模型如下：

$$\left\{ \begin{array}{l} \min F = \sum_{i=1}^n f_c(\Delta P_{MGi}) \\ \text{s.t.} \quad \Delta P = \sum_{i=1}^n \Delta P_{MGi} = \\ \quad \sum_{i=1}^n (P_{Loadi} - P_{PVi} - P_{WTi} - P_{DEi}^{\min}) \\ \Delta P_{MGi} = \Delta P_{ESi} + \Delta P_{DEi} + \Delta P_{ILi} + \Delta P_{DG-i} \\ \Delta P \Delta P_{MGi} > 0 \\ \Delta P_{MGi}^{\min} < \Delta P_{MGi} < \Delta P_{MGi}^{\max} \end{array} \right.$$

上式中：微电网群总功率指令 ΔP 为微电网群总负荷与总出力及储能装置强迫出力的差额； ΔP_{Mgi} 为第 i 个微电网的功率指令，即需要承担的功率指令即功率调节量； $f_c(\Delta P_{Mgi})$ 为第 i 个微电网的调节成本，是关于微电网功率调节量的函数； P_{Loadi} 为第 i 个微电网的总负荷； ΔP_{Mgi}^{max} 和 ΔP_{Mgi}^{min} 分别为第 i 个微电网的功率调节量上、下限，取决于微电网中各控制单元的实时状态， n 为微电网的个数。

当微电网整体出现功率不平衡时，将由全体可参与调节的微电网共同成电，选取微电网的调节成本作为一致性变量，则能够时各个微电网根据自身资源参与调节，其本质就是根据各微电网调节成本函数不同段的斜率数值分配各个微电网的功率调节量，以达到降低微电网总体调节成本的目的。

这里我们将调节成本简记为 C ，基于离散时间一阶一致性算法，各智能体一致性变量更新公式如下：

$$C_i^{(k+1)} = \sum_{j=1}^n d_{ij}^{(k)} C_j^{(k)}$$

式中： $C_i^{(k)}$ 为第 i 个微电网第 k 步迭代计算得到的调节成本； $d_{ij}^{(k)}$ 为行随机矩阵在第 k 步迭代时的第 i 行第 j 列元素，行随机矩阵由通信拓扑的拉普拉斯矩阵求得，与通信拓扑的结构相关。

各智能体借助与通信拓扑相关的行随机矩阵获取相邻智能体上一步迭代的状态信息，以更新自身状态。而为了确保功率平

衡，由领导者来引导调节成本调节的方向与幅度，作为领导者的微电网调节成本更新公式如下。

$$C_i^{(k+1)} = \begin{cases} \sum_{j=1}^n d_{ij}^{(k)} C_j^{(k)} + \mu \Delta P_{\text{error}} & \Delta P > 0 \\ \sum_{j=1}^n d_{ij}^{(k)} C_j^{(k)} - \mu \Delta P_{\text{error}} & \Delta P < 0 \end{cases}$$

上式中： μ 为误差调节步长；忽略微电网群系统的线路损耗时， ΔP_{error} 为总功率指令与各微电网功率指令之和的偏差量。 ΔP_{error} 的表达式为：

$$\Delta P_{\text{error}} = \Delta P - \sum_{i=1}^n \Delta P_{\text{MGi}}$$

其中误差调节步长 μ 可人为给定合适的参数 或通过检测 ΔP_{error} 来自适应地调节。微电网调节成本更新公式含义可以做如下解释：以目前总功率指令 $\Delta P > 0$ 为例，若 $\Delta P_{\text{error}} > 0$ 即说明各微电网的功率指令总和仍不足以平衡当前的功率不平衡量，调节成本需要相应增加；反之若 $\Delta P_{\text{error}} < 0$ 则调节成本可减少。

原则上领导者的选取应由调节能力决定，选取调节能力最大的微电网作为领导者可以减少替换领导者的操作，而所谓的调节能力可以由储能容量、发电机容量、可中断负荷容量等反映微电网可调控资源水平的参数来体现。

综上，在迭代过程中各微电网的调节成本为本地及邻居上一步迭代的调节成本的加权平均，网络的通信负担小。当出现部分微电网已达到功率调节量的限值时则应退出通信拓扑，停

止更新，相邻的微电网也应根据新的通信拓扑修改相应的行随机矩阵元素。在本文一性算法收敛过程中，取 $|\Delta P_{\text{error}}| < \varepsilon$ 为收敛条件，其中 ε 为收敛误差。

当微电网之间通过通信交互更新调节成本后，各微电网需要由调节成本变量及相应的调节成本函数来得到各个智能体的功率。

2、多智能体一致性算法关于微电网的有功功率与无功功率分配

微电网中，每个分布式发电源输出有功功率与无功功率是非常重要的两项指标。根据实际工程中使用通讯模块进行周期控制可能存在的延迟、丢包等问题，也可以将多智能体一致性算应用到基于事件触发的多智能体微电网有功功率、无功功率的分配中去。

首先本分配方法中采用的为离散时间一致性算法，公式如下：

$$x_i(k+1) = x_i(k) + \varepsilon \sum_{j \in N_i} a_{ij} (x_j(k) - x_i(k))$$

为了将一致性算法应用于微电网中，再每一采样时刻到来时，将微电网中分布式发电源的有功功率 P 和无功功率 Q 作为智能体本身的状态信息将本地输出的有功功率与无功功率预期邻居进行分享。在智能体获取了邻居的有功功率与公共率的信息后根据下面 P 与 Q 的更新公式计算下一时刻期望输出的有功功率和无功功率。公式如下：

$$P_i(k+1) = P_i(k) + \varepsilon \sum_{j \in N_i} a_{ij} (P_j(k) - P_i(k))$$

$$Q_i(k+1) = Q_i(k) + \varepsilon \sum_{j \in N_i} a_{ij} (Q_j(k) - Q_i(k))$$

其中， $P_j(k)$ 为当前时刻智能体邻居的有功功率值， $Q_j(k)$ 为当前时刻智能体邻居的无功功率值，为了配合垂直控制的相关指示并且便于计算将公式的后半部分作为 P_{ref} 与 Q_{ref} 的值，便能使得整

个系统以离散时间一致性算法的形式进行智能体状态的调整，在调整一段时间后将分布式电源输出的有功功率和无功功率调整至期望值。即：

$$P_{ref} = \varepsilon \sum_{j \in N_i} a_{ij} (P_j(k) - P_i(k))$$

$$Q_{ref} = \varepsilon \sum_{j \in N_i} a_{ij} (Q_j(k) - Q_i(k))$$

即使在一个控制周期过后有功功率和无功功率尚未达到期望输出的值，但只要功率时王均分方向改变的，便能够在一定的周期之后分到一致。另外适当地调节一致性算法地步长 ε 的值，能够得到不同的系统动态响应效果。对于不同容量的分布式电源，为了让每个分布式电源按照比例输出功率，只需要在一致性算发中乘以一个与容量相关的系数，便可以让每个分布式电源按照比例输出功率。

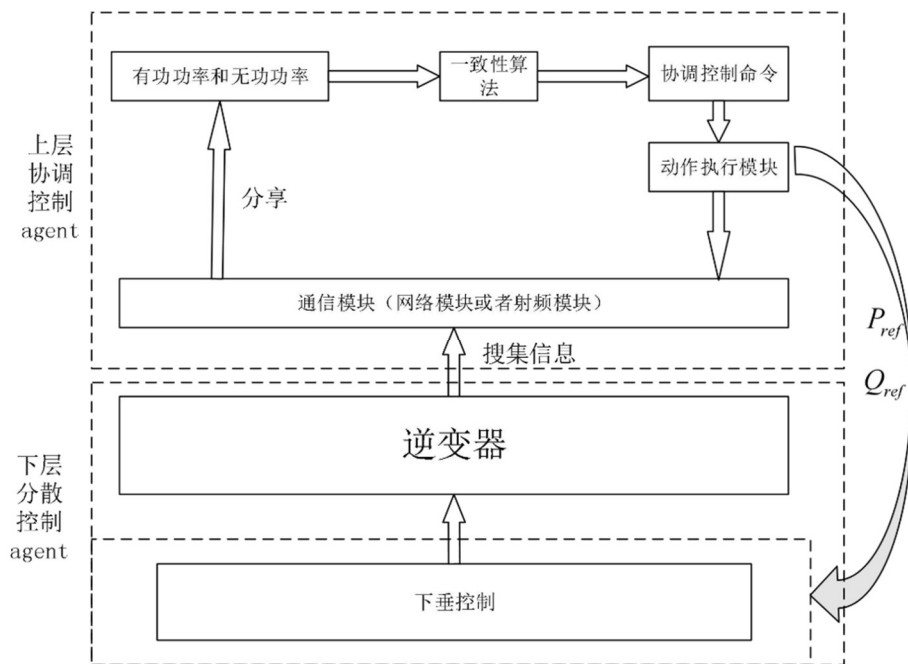


图 基于多智能体的功率分配控制器

上图中下垂控制和离散时间一致性算法控制组成了分层控制的上下两层。其中下垂控制作为底层控制算法，控制着逆变器的输出，而离散时间一致性算法作为上层控制，接受智能体的信息，在这里就是邻居逆变器的输出值 P 和 Q ，然后根据一致性算，最终将影响本地下垂控制的参数，进而改变本地逆变器的输出有功功率和无功功率，经过一段时间的调整，最终让整个系统的功率达到均分。

3、多智能体一致性算法在微电网分布式分层控制策略的应用：

结合微电网分层控制的基本框架，以实现每一层的分布式控制为目标，提出了基于多智能体一致性算法的分布式分层控制策略，以维持微电网系统频率和电压稳定，以及实现有功、无功负荷在分布式电源的灵活分配。

由于微电网系统缺少惯性，抗扰动能力较弱，控制环节复杂多样，因此采用微电网分层控制策略。该策略分为一下三层：

①初级控制层：该控制层主要基于传统的分布式电源下垂控制策略进行有功频率及无功电压的控制，实现功率的分配和频率、电压的调节。

②二次控制层：该层主要为消差环节，主要目的在于消除由初级控制层产生的频率和电压的偏差，将频率和电压维持在额定值的附近。

③三次控制层：第三层为调度层，控制各个分布式电源之间及微电网与外界的功率流动。

此控制策略实在初级下垂控制的基础上进行二次和三次调整。主要控制目标实在有功各无功负荷变化的情况下，通过分布式的分层控制方法完成有功、无功功率的合理分配，同时维持电网系统频率、电压的稳定。

由于初级控制层和二次控制层主要涉及的电力等相关知识，所以下面将主要介绍三次控制层的工作方式和原理。

本层利用本不是算法，实时响应负荷变化，根据各分布式电源

的功率可调量进行负荷分配，使微电网运行在更优的工作点。功率分配的具体过程分为以下三步。

①总有功可调量估算

为了可以根据各分布式电源的公共率可调量大小进行负荷响应量的分配，需要先计算出相应比参考值 α_p 。引入总有功可调量的总和，但是该信息无法直接获取计算，需要通过分布式算法进行估算。一个分布式电源有功可调量 $P_{i,adj}$ 为状态量，最终收敛到的结构可以作为各有功可调量的平均值 $P_{avr,adj}$ 的估计量，因此可恶意得到总有功可调量：

$$P_{total,adj} = nP_{avr,adj}$$

②功率分配

在负荷增加 P_D 后，各分布式电源增加出力为 ΔP_i 。设定某台分布式电源作为主电源，或称为主智能体，与其他智能体的区别在于主智能体可以直接通过通讯获得负荷侧的负荷增加量 P_D 。定义分布式电源的响应比 $\alpha_{p,i} = \Delta P_i / \Delta P_{i,adj}$ ，也就是分布式电源 i 对负荷变化的响应量与其有功可调量的比值。

在分配响应功率时，根据有功可调量的大小进行分配更为合理，即可调量大的分布式电源承担更多的负荷变化量。因此这里的目标是使得相应比一致，项映的控制律为：

$$\begin{cases} \dot{\alpha}_{p,i}(t) = u_i(t) \\ u_i(t) = \sum_{j \in N_i} a_{ij} (\alpha_{p,j}(t) - \alpha_{p,i}(t)) \end{cases}$$

在达到一致后有：

$$\alpha_{p,1} = \alpha_{p,2} = \cdots = \alpha_{p,n} = \alpha_p^* = \frac{P_D}{P_{total,adj}}$$

在这个过程中，最终各个分布式电源的相应比达到一致，各个智能体就可以根据相应的相应比计算出有功相应比 $\Delta P_i = \alpha_p^* \Delta P_{i, adj}$ 。

③下垂系数调节（和一致性算法无重要关系）

在得到各个分布式电源的功率增加量 ΔP_i 后，即可得到负荷增大后各个分布式电源的出力 P_i' ；

$$P_i' = P_i + \Delta P_i$$

然后通过调节初级控制层中的下垂控制系数 m_i 来调节各台逆变器的输出功率，即调节 m_i 的值，使得新的下垂系数 m_i' 满足：

$$m_1' P_1' = m_2' P_2' = \cdots = m_n' P_n' = C'$$

由于影响功率分配的因素时下垂系数倒数之间的比值，因此假定 m_1 不变，即 $m_1 = m_1'$ ，则修正后的下垂系数计算公式为：

$$m_i' = \frac{C'}{P_i'}$$

在各个智能体计算出新的下垂系数后，将新的下垂系数发送至初级控制层的下垂控制环节，从而实现功率的合理分配。