

单位代码： 10293 密 级： \_\_\_\_\_

南京邮电大学

硕士学位论文



论文题目： 基于 NETLOGO 与 MATLAB 的电网多智能体  
建模及仿真研究

学 号 1013051320

姓 名 金 珍

导 师 岳 东

学 科 专 业 控制理论与控制工程

研 究 方 向 电力系统仿真研究

申请学位类别 工学硕士

论文提交日期 二〇一六年三月

# **Modeling and Simulation Study for Power System Multi-agent Based on NETLOGO and MATLAB**

Thesis Submitted to Nanjing University of Posts and  
Telecommunications for the Degree of  
Master of Engineering



By

JIN Zhen

Supervisor: Prof. YUE Dong

March 2016

## 南京邮电大学学位论文原创性声明

本人声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得南京邮电大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

本人学位论文及涉及相关资料若有不实，愿意承担一切相关的法律责任。

研究生签名：\_\_\_\_\_ 日期：\_\_\_\_\_

## 南京邮电大学学位论文使用授权声明

本人授权南京邮电大学可以保留并向国家有关部门或机构送交论文的复印件和电子文档；允许论文被查阅和借阅；可以将学位论文的全部或部分内容编入有关数据库进行检索；可以采用影印、缩印或扫描等复制手段保存、汇编本学位论文。本文电子文档的内容和纸质论文的内容相一致。论文的公布（包括刊登）授权南京邮电大学研究生院办理。

涉密学位论文在解密后适用本授权书。

研究生签名：\_\_\_\_\_ 导师签名：\_\_\_\_\_ 日期：\_\_\_\_\_

## 摘要

为了实现源网荷互动的电力系统功率平衡控制，提出了一种电网多智能体建模、仿真与控制方案。该方案仿真平台由 NETLOGO 与 MATLAB 组成，其中 NETLOGO 承担电力系统智能元件建模以及电网多智能体控制的工作，MATLAB 负责电力系统的各项运算，通过 NETLOGO 和 MATLAB 之间的接口模块实现整个系统网络数据交互。仿真方案中电力元件智能体（Agents）与 MATLAB 通过接口传递交互信息，电力元件智能体将有关参数上传给 MATLAB 进行电力系统各项运算，同时调用 MATLAB 中最新信号下达给电力元件智能体，各个电力元件智能体考虑自身目标作出积极响应。仿真结果表明，基于 NETLOGO 与 MATLAB 联合的电网多智能体建模、仿真与控制方案可视化效果好、计算速度快、易于扩展，能够满足电网多智能体功率平衡控制的要求。

**关键词：** 电网多智能体建模；电网多智能体仿真；电网多智能体控制；智能电力元件；NETLOGO 仿真；MATLAB 仿真

## Abstract

To realize the power balance control in source-grid-load interaction power systems, a power system multi-agent modeling, simulation and control scheme is proposed. The simulation platform composes by NETLOGO and MATLAB. In the adopted modeling, simulation and control scheme, the NETLOGO is adopted for smart power system component modeling and multi-agent power system control. The MATLAB is responsible for power system calculation, such as power flow calculation, marginal electricity price calculation, and optimal power flow calculation. The power system measurement data are transferred by the interface of NETLOGO and MATLAB. Using the interface of NETLOGO and MATLAB, power component agents and MATLAB can transfer system data. The power component agents transmit parameters to the MATLAB and calculate the marginal electricity price in MATLAB, then releasing new data to the power component agents. The power component agents response to the information considering own objection. Simulation results show that the proposed modeling, simulation and control scheme based on NETLOGO and MATLAB has the following characteristics: efficient visual effects, fast computation, easy to expand, and the ability to satisfy the demand of power balance control in source-grid-load interaction power systems.

**Key words: power grid multi-agent modeling; power grid multi-agent simulation; power grid multi-agent control; smart power system component; NETLOGO simulation; MATLAB simulation**

# 目录

第一章 绪论 .....	1
1.1 课题背景 .....	1
1.2 国内外研究现状 .....	1
1.2.1 电网多智能体仿真环境 .....	2
1.2.2 电网多智能体控制 .....	2
1.3 论文主要工作 .....	3
第二章 电力系统与多智能体系统的关联性分析.....	4
2.1 电力系统的复杂特性 .....	4
2.2 多智能体系统理论简介 .....	4
2.3 基于多智能体理论的电网特征分析 .....	5
2.3.1 逆变器同步及功率平衡控制 .....	6
2.3.2 电动汽车电网频率偏差分布式控制.....	9
第三章 电网多智能体的仿真环境与建模 .....	12
3.1 电网多智能体仿真平台组成 .....	12
3.1.1 NETLOGO 软件部分.....	12
3.1.2 MATLAB 软件部分 .....	15
3.1.3 NETLOGO 与 MATLAB 接口部分 .....	18
3.2 电网多智能体的仿真环境 .....	21
3.2.1 电网多智能体仿真概述 .....	21
3.2.2 基于 NETLOGO 的仿真环境搭建.....	21
3.2.3 基于 MATLAB 的仿真环境.....	24
3.2.4 NETLOGO 和 MATLAB 接口环境搭建 .....	26
3.3 电网多智能体元件的建模 .....	28
3.3.1 电网多智能体元件建模概述 .....	28
3.3.2 对响应电价的负荷建模 .....	28
3.3.3 NETLOGO 程序实现.....	35
第四章 电网多智能体协调控制与仿真 .....	40
4.1 基于 Prosumer 模型的优化控制与仿真 .....	41
4.1.1 基于 Prosumer 模型的优化控制策略 .....	41
4.1.2 基于 Prosumer 模型的网荷互动仿真 .....	42
4.2 基于空调模型的优化控制与仿真 .....	48
4.2.1 基于空调模型的优化控制策略.....	48
4.2.2 基于空调模型的网荷互动仿真.....	49
第五章 总结与展望 .....	54
5.1 全文总结 .....	54
5.2 进一步工作的展望 .....	54
参考文献 .....	55
附录 1 攻读硕士学位期间撰写的论文 .....	58
附录 2 攻读硕士学位期间申请的专利 .....	59
附录 3 攻读硕士学位期间参加的科研项目 .....	60
致谢 .....	61

# 第一章 绪论

## 1.1 课题背景

随着我国不断地深化电力市场改革，打破垄断，引入竞争，同时积极发展大用户直购电试点工作，电网功率平衡控制问题不在单一的依靠调节发电机发电量来解决，终端用户也可参与调节，使得电网的功率平衡调节多样化。故用户的需求响应（DR）在一定程度上不仅影响电力市场交易，同时对电网功率平衡控制有一定的调节作用，即负荷在竞价市场中通过不同的价格信号改变自身的用电模式，从而达到调节电网功率平衡的目标<sup>[1]-[2]</sup>。为了实现电网安全稳定运行，负荷的需求响应成为需求侧管理的重点。

在电网功率平衡控制中引入负荷，大大提高了电网系统的可靠性与经济性。但是如何设置负荷参与量则与整个系统的运行状况密切相关，因此有必要对参与功率平衡调节的负荷进行一定的控制。基于多智能体的分布式控制方法能够有效应对负荷自身的多变性和不确定性，因此我们采用分布式控制方法对负荷进行控制。每个负荷具有一定的智能性，能够应对外界扰动，根据调节信号实时改变用电模式从而影响整个电网的用电需求量，以保证电网更加可靠和经济的运行。

## 1.2 国内外研究现状

源网荷互动电网的系统功率平衡控制问题已经成为当前的一个研究热点<sup>[3]-[6]</sup>。由于电力系统的内在复杂性，可以把其看做一个复杂系统。电力系统系统的主要任务是可靠、安全地把电力从发电端输送至用电中心。同时，为了有效应对复杂的外在环境，比如负荷持续的波动、因为故障造成的设备切换、天气及气候条件的大范围变化等，需要实时地收集、传送、分析各种信息来监测电网的运行状态进而加以控制。另外，电力系统管理机制正在进行改革，电力系统的复杂性也随之增加。过去，电网的层级监测和控制是由供电部门的集中控制中心来支撑的。但是近年来，为了最大化降低电力成本，电力系统运行正在逐步市场化。市场化带来的竞争给电力系统的安全运行带来了挑战，特别是如何在保证电网安全的同时最大化电网可用资源的利用率。为了应对这些挑战，全世界的供电部门开展了一体化、系统化的监测、计算和控制系统的研究和建设。在电力市场背景下，集中式的决策中心被具有竞争关系的多

主体决策方式代替，这使得电网运行状态的预测愈加困难。再者，根据电压等级的不同，电网被划分为发电、输电和配电。风电、光伏、燃料电池等新能源正在越来越多的进入到传统垂直管理方式下的配电网内。在这种场景下，传统的电力用户将从原先的电力使用者转变为具有向电网汲取和提供电能的双重功能的一方，电网的运行与控制变得越来越复杂<sup>[7]-[8]</sup>，传统的方法已无法满足电网的要求。

### 1.2.1 电网多智能体仿真环境

MATLAB 是一种以矩阵为基本编程单元的程序设计语言，具有强大的计算功能和编程技术，广泛应用于电力系统仿真研究<sup>[9]</sup>。MATLAB 强大的计算功能和编程技术，便于建立电力系统元件的模型和建立复杂的电力网络仿真模型。NETLOGO 是适用于对某一随时间改变的系统进行仿真。论文采用 NETLOGO 与 MATLAB 联合仿真的方式构建电网多智能体仿真环境<sup>[10]</sup>。其中，NETLOGO 完成电力系统元件通用模块的搭建，MATLAB 进行电力系统的各项计算，求解得到的网络参数通过 MATLAB 和 NETLOGO 之间的接口程序实现交互。

### 1.2.2 电网多智能体控制

多智能体控制已应用在很多工程领域，比如机器人系统，飞行器系统，传感器网络系统等<sup>[11]-[14]</sup>。其基本思想是通过控制单元共同参与局部信息交互的方式传递全局控制信息。近几年，多智能体控制已经应用在电力系统中<sup>[15]-[18]</sup>：多智能体控制应用于光伏单元的功率协调输出中来使其有功功率达到某一特定值<sup>[15]</sup>，通过以边际成本为一致性变量的分布式一致性算法调整多个发电机的输出来实现最优经济调度<sup>[16]-[17]</sup>，以及应用“Leader-Follower”一致性算法实现 DESS 的实时功率分配<sup>[18]</sup>。上述应用多智能体控制策略都需要收集系统的全局信息并传送至“调度中心”或“Leader”来计算所有单元的总功率，因此这些调度策略不属于完全意义上的分布式调度策略，而且没有考虑到源网荷互动电网中电力元件的多样性，如柔性负荷，分散储能设备，分布式电源等。

论文仿真环境中的电网多智能体控制策略，考虑电力元件的多样性，结合电网互动运行的特性构建包括可再生能源、分布式发电、储能、需求响应、常规发电等多种类型发电、用电元件的多智能体电网控制策略。



### 1.3 论文主要工作

论文主要工作是电网元件在多智能体理论体系中的准确表达，构建多智能体电网运行环境，以及提出多智能体协调控制策略。具体内容如下：

（1）概述源网荷互动电网中智能电网元件与多智能体理论体系之间的关系，特别是电网元件智能特性与多智能体系统的关联性，分析了基于多智能体理论对电网运行加以控制的案例。

（2）提出若干个电网多智能体协调控制策略，包括考虑电力元件互动的电网优化控制策略，考虑柔性负荷互动的电网功率平衡控制策略，考虑空调参与电网功率平衡调节的控制策略。

（3）设计了基于 NETLOGO 与 MATLAB 联合的电网多智能体建模、仿真与控制方案。NETLOGO 平台主要仿真电力元件受到激励信号的影响自主改变用电模式的过程；MATLAB 仿真平台主要负责进行电力方面的各项计算，使得该电力系统能够安全稳定的运行；对由 NETLOGO 与 MATLAB 联合的电网多智能控制进行模拟仿真，验证了该建模、仿真与控制方案的可行性。

## 第二章 电力系统与多智能体系统的关联性分析

### 2.1 电力系统的复杂特性

随着科技的进步和社会的发展,新能源等新技术的引入,电力系统的复杂程度越来越高,使得电力系统的运行及调度产生诸多挑战。电力系统的异构特点使得传统的集中式调度方法面临困难,对于如此复杂的系统,集中式调度的工作量繁重且智能性程度不高,每个组成部分的动作需要控制中心统一下达命令,费时费力。风力发电及光伏发电的发电量存在不确定性,如果不能实时的对系统做出调整可能给整个系统带来不稳定因素甚至严重后果。因此需要在传统调度方式的基础上使每个电力系统组成模块具有智能性,模块与模块之间能够进行信息交互从而调整自身行为以达到自身目标最大化,当电网出现功率不平衡时对负荷进行实时调度以及分布式调度,以保证整个电网完全可靠运行。故如何将复杂的电力系统用一个多智能体系统来表达成为了如今研究的热点<sup>[19]</sup>。在多智能体系统中,每个模块可以是发电机、输电线,也可以是负荷,都具有智能性,可以通过与相邻模块的沟通完成参数以及动作的自我调整。而传统的仿真平台系统不能很好地实现模块的智能性,各个模块间也不能实现信息和数据的交换,因此不能很好地对电力系统进行直观的仿真建模。

### 2.2 多智能体系统理论简介

多智能体系统又叫多主体系统、自主体系统或群体系统。目前还没有一个严格的、统一的定义,但中科院系统所洪奔光教授给出了以下的多智能体系统定义较具代表性:多智能体系统,它是由一群具备一定的传感、计算、执行和通信能力的智能体通过通讯等方式关联成为一个网络系统。多智能体系统具有自主性、分布性、协调性,并具有自组织能力、学习能力和推理能力。个体可以根据各自的目标,动态地规划各自的运动行为和状态,在发挥其智能和自主行为的基础上并通过与周围邻居智能体进行合作来完成特定的任务。利用多智能体系统来解决实际应用中的问题,具有很强的鲁棒性和可靠性,并具有较高的问题求解效率。研究多智能体系统的主要目的在于利用多个智能体所组成的多智能体系统之间的协作来完成单个智能体能力范围外的复杂任务。这是当前控制领域的一个新兴课题,具有重要的理论和现实意义和广泛的应用前景。与一般的网络控制系统相比,多智能体系统具有以下优点:拥

有广泛的任务领域；高效率；良好的系统性能；容错性强；鲁棒性强；低廉的经济成本；易于开发；具有分布式的感知与作用等。从目前研究来看，多智能体系统协调控制中主要包括群集运动、编队控制、一致性控制、聚集控制等基本问题。

## 2.3 基于多智能体理论的电网特征分析

有上述介绍可知，多智能体系统由大量个体组成，每个个体通过局部通信所获得的信息来控制自身的行为，进而使整个系统完成某种控制目标。这种控制方式称为多智能体的分布式协同控制。多智能体系统中的个体(agent)具有如下特征：（1）分布式架构：个体分布在环境中，通过某种网络实现信息交互；（2）自主能力：个体具有自身的控制目的并被其驱动；（3）交互能力：个体能够与系统中某些其他个体进行信息交互。对于电网元件来讲，上述三条特征均能满足：首先，各电气元件处于电力网络及信息网络的分布式环境下，并可以通过这两种网络实现信息交互；其次，各电气元件具有自身的控制需求及目标，例如：个体需要保持稳定符合电网要求的电压、电流、频率、同时还需保持电能供需平衡，还有一定的经济性要求等，这些都是电气元件的控制目标，电网元件受上述控制目标驱动，通过所设计的控制策略调节自身状态达到控制目的；最后，通信技术的发展使得电力系统对信息网络具有高度依赖性，使得各电网元件之间的信息交互成为了可能。由此可知，电网元件满足多智能体系统中个体的基本特征，电网系统和多智能体系统在特性上（分布式架构、自主能力、交互能力）是相符合的，同时电力系统通过调节个各元件的状态使得电网运行达标，实现整体控制需求。因此，电力系统本质上就是一类多智能体系统，而系统中的各电网元件则为这个多智能体系统中的各智能体。

上述分析说明，运用多智能体系统理论体系研究电网特性及控制方法是合理的,因此多智能体系统分布式协同控制的诸多优势亦能在电网控制中体现。这些优势主要体现在：（1）局部交互性。每个元件仅需与其相邻的元件进行信息交互，相比传统集中式方法而言大大减少了网络通讯压力，且不再需要集中式控制中心。例如：一个负载仅需要和附近的负载或处于同一微网中的电力设备进行交互来实现电能调度，而不再需要与网络中大量的设备或与其相距很远的电力设备进行信息交互；（2）高灵活性。系统结构上的改变不会对整个系统的功能造成明显不良的影响。例如：在电网中增加一个发电设备，系统将自动的将其融入电网架构并令其快速的投入运行。而传统电网控制方法此时往往需要整个系统进行重新设计；（3）高容错性。当中某一电力设备损坏时，整个系统将自动处理故障并维持运行。为此，运用多智

能体系统理论体系探讨电网元件分布式控制策略具有其独特且强大的优势，有必要对其进行深入研究。

由前述分析可知，运用多智能体系统理论体系研究电网特性及控制方法具有合理性，且多智能体系统分布式协同控制的诸多优势亦能在电网控制中体现，电力系统本质上就是一类多智能体系统，各电网元件即为这个多智能体系统中的各智能体。为在多智能体系统理论体系下开展电网控制研究工作，首先对基于多智能体理论控制电网运行的案例分析。

### 2.3.1 逆变器同步及功率平衡控制

考虑一个孤岛微网，针对其频率同步及功率平衡问题展开研究，在多智能体系统理论体系下，构建微网中逆变器及负荷的频率动态特性模型，并结合多智能体系统分布式协调控制思想，构建逆变器的分布式频率调控策略<sup>[20]</sup>。设孤岛微网具有线性 AC 回路且具有纯虚数导纳矩阵  $Y \in R^{N \times N}$ ，其所对应一个无向连通且具有复数权值的图  $G(V, E, A)$ ，其中点集（或总线集） $V = \{1, 2, \dots, n\}$ ，边集（或分支集） $E \subset V \times V$ ，权值  $a_{ij} = -Y_{ij} = -Y_{ji} \in C$ 。将总线集  $V$  分解为  $V = V_L \cup V_I$ ，其中  $V_L$  与  $V_I$  分别对应微网中的负荷与逆变器集合，并设集合  $V$ 、 $V_L$  与  $V_I$  的基数分别为  $n$ 、 $n_L$  与  $n_I$ ，如图 2.1。

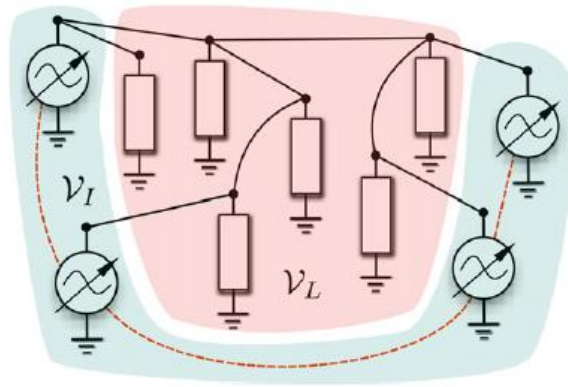


图 2.1 包含负荷与逆变器的孤岛微网，其中红线表示通信线路

对于每一个  $i \in V$ ，定义其有功功率注入（active electrical power injection）为  $P_{e,i} \in R$ ，相量电压  $V_i = E_i e^{j\theta_i}$ ，其中幅值  $E_i > 0$ 、相移  $\theta_i \in S$ 。如此，在节点  $i$  的有功功率注入  $P_{e,i}$  为：

$$P_{e,i}(\theta) = \sum_{j=1}^n E_i E_j |Y_{ij}| \sin(\theta_i - \theta_j) \quad (2.1)$$

对于每一个负荷  $i \in V_L$ ，其需求恒定的有功功率  $P_i^*$ ，且满足如下的功率平衡方程：

$$0 = P_i^* - P_{e,i}(\theta), \quad i \in V_L \quad (2.2)$$

对于每一个逆变器  $i \in V_I$ ，其频率变化遵循如下的下垂控制理论：

$$\omega_i = \omega^* - n_i(P_{e,i} - P_i^*) \quad (2.3)$$

其中， $\omega^*$  为额定频率， $P_i^*$  为标称有功功率注入量，参数  $n_i > 0$  为下垂率。

由此，逆变器  $i \in V_I$  的频率动态特性为：

$$D_i \dot{\theta}_i = P_i^* - P_{e,i}(\theta) + u_i(t), \quad i \in V_I \quad (2.4)$$

其中， $D_i = n_i^{-1}$ ， $u_i$  表示控制输入， $P_i^* \in [0, \bar{P}_i]$  为标称注入， $\bar{P}_i$  为逆变器  $i$  的额定功率极限。同时， $\dot{\theta} = \omega_i - \omega^*$  表示频率相对于额定值的偏差。

综合上述方程，且对于每个负荷  $i \in V_L$  令  $u_i = 0$ ，则可知，若闭环系统存在一同步频率  $\dot{\theta}_i(t) = \omega_{sync} \in R, (i=1,2,...,n)$ ，则  $\omega_{sync}$  可明确地表示为如下的形式：

$$\omega_{sync} = \omega_{avg} = \frac{\sum_{j \in V} P_j^*}{\sum_{j \in V_I} D_j} \quad (2.5)$$

通过旋转变换  $\theta(t) \rightarrow \theta(t) - \omega_{avg} t \pmod{2\pi}$ ，上述负荷及逆变器的频率动态可等价地表述为如下的多智能体系统控制模型：

$$\begin{aligned} 0 &= \tilde{P}_i - P_{e,i}(\theta), \quad i \in V_L \\ D_i \dot{\theta}_i &= \tilde{P}_i - P_{e,i}(\theta), \quad i \in V_I \end{aligned} \quad (2.6)$$

其中，对于  $i \in V_L$ ， $\tilde{P}_i = P_i^*$ ；对于  $i \in V_I$ ， $\tilde{P}_i = P_i^* - D_i \omega_{avg}$ 。

引入邻接矩阵  $B$  来描述元件间的电路连接特性，则上述式子可等价地表述为：

$$\tilde{P} = BA \sin(B^T \theta^*) \quad (2.7)$$

其中， $A = \text{diag}(\{E_i E_j | Y_{ij}\}_{\{i,j\} \in E})$ ， $\tilde{P} = (\tilde{P}_1, \tilde{P}_2, \dots, \tilde{P}_n)$ 。

由此，可证得，定义变量  $\xi = B + \tilde{P}$ ，若令所有的逆变器  $i \in V_I$  的输入量  $u_i = 0$ ，则如下两种陈述等价：

1) 可找到一个弧长  $\gamma \in [0, \pi/2)$  使得闭环系统存在一个局部指数稳定且唯一的同步解

$$\theta^*(t) \in \Delta_G(\gamma);$$

$$2) \Gamma \equiv \|A^{-1}\xi\|_{\infty} < 1。$$

进一步地，若上述两种陈述满足，则同步解为  $\theta^*(t) = \theta_0 + (\omega_{\text{sync}} t \mathbf{1}_n) \pmod{2\pi}$ ，其中  $\theta_0 \in \Delta_G(\gamma)$ ， $\omega_{\text{sync}} = \omega_{\text{avg}}$ 。

上述结果表明，当所有逆变器  $i \in V_l$  的输入量  $u_i = 0$  时，并不能达到功率平衡，为此，还需设计分布式控制器使得  $\omega_{\text{avg}} = 0$ ，即使得微网频率稳定在额定值。为此，对于已获知的多智能体系统控制模型：

$$\begin{aligned} 0 &= \tilde{P}_i - P_{e,i}(\theta), i \in V_L \\ D_i \dot{\theta}_i &= \tilde{P}_i - P_{e,i}(\theta), i \in V_l \end{aligned} \quad (2.8)$$

结合多智能体分布式协调控制理论体系，针对每个逆变器  $i \in V_l$  构建如下的分布式控制策略：

$$u_i = -p_i, k_i \dot{p}_i = D_i \dot{\theta}_i - \sum_{j \in V_l} L_{c,ij} \left( \frac{P_i}{D_i} - \frac{P_j}{D_j} \right) \quad (2.9)$$

其中， $L_c$  表示微网中逆变器之间通信图的 Laplacian 矩阵， $p_i \in R$  为所设定的虚拟功率变量， $k_i > 0$  为参数。由此，可获得如下的闭环系统：

$$\begin{aligned} 0 &= P_i^* - P_{e,i}(\theta), & i \in V_L, \\ D_i \dot{\theta}_i &= P_i^* - p_i - P_{e,i}(\theta), & i \in V_l, \\ k_i \dot{p}_i &= D_i \dot{\theta}_i - \sum_{j \in V_l} L_{c,ij} \left( \frac{p_i}{D_i} - \frac{p_j}{D_j} \right), & i \in V_l. \end{aligned} \quad (2.10)$$

其反馈控制框图见图 2.2，控制架构见图 2.3。

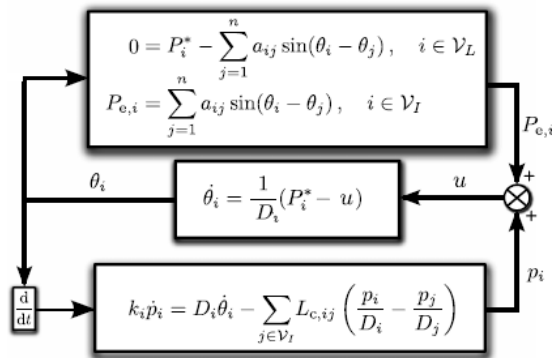


图 2.2 分布式控制策略框图

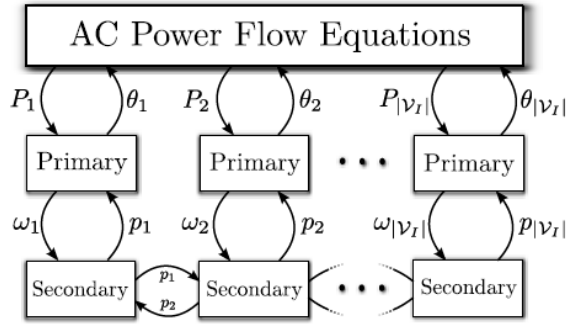


图 2.3 分布式控制策略架构

针对上述分布式控制策略，可证得如下结果：

在通信图连通的前提下，如下两种陈述等价：

1)  $\Gamma \equiv \|A^{-1}\xi\|_{\infty} < 1$ ;

2) 可找到一个弧长  $\gamma \in [0, \pi/2)$  使得闭环系统存在一个局部指数稳定且唯一的平衡点

$(\theta^*, p^*) \in \Delta_G(\gamma)XR^n$ 。

进一步地，若上述两种陈述满足，则在平衡点处，有  $p_i^* = D_i \omega_{avg}$ 。

由此，在多智能体系统架构下，构建了孤岛微网逆变器与负荷的控制模型，完成了电网设备功率平衡及频率调控的多智能体模型表述，并结合具有物理关联特性的多智能体系统分布式协同控制思想，设计了孤岛微网逆变器分布式频率调控策略。

### 2.3.2 电动汽车电网频率偏差分布式控制

考虑一个包含配电变电站、插电式电动汽车及负荷的配电网<sup>[21]</sup>，如图 2.4 所示。

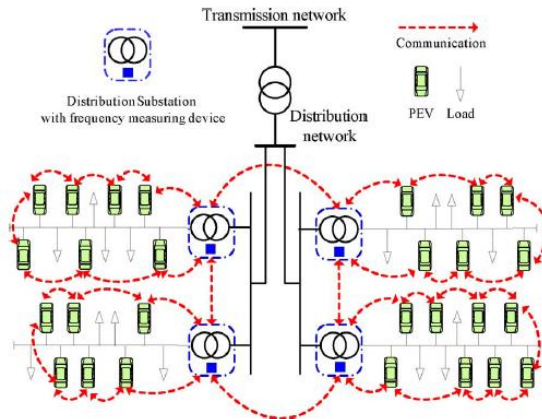


图 2.4 包含配电变电站、插电式电动汽车及负荷的配电网

频率检测设备仅安装在配电变电站处，令每个变电站处的电网频率信号测量值为：

$$u_i(t) = \Delta \omega(t) + \sigma_i(t) d\zeta_i / dt, i=1, \dots, l \quad (2.10)$$

其中， $\Delta \omega$  为电网频率距额定值的偏离量， $u_i$  表示第  $i$  个变电站对于电网频率偏差量的实际测量值， $l$  表示变电站数量， $\zeta_i$  表示 Brownian 运动， $d\zeta_i$  为一均值为 0 方差为 1 的高频白噪声， $\sigma_i$  表示噪声强度，其中  $\beta = \max\{\sigma_i^2\}$  表示噪声方差的上界。由此可见，由于测量误差及外部干扰等因素，使得变电站处所测得的电网频率偏差值距真实值有较大差距，为此，考虑配电网中插电式电动汽车及变电站之间的局部信息交互特性，运用多智能体系统理论体系，运用分布式一致性滤波的思想尽可能的降低由外部扰动所造成的测量误差，以获得较为精确的电网频率偏差值。

运用图  $G=(V, E)$  来描述配电网中各设备间的通信网络拓扑结构，图  $G$  具有非负邻接矩阵  $A=\{a_{ij}\}$ 。若在节点  $i$  和节点  $j$  之间具有无向连通关系，则  $a_{ij}=a_{ji}=1(i \neq j)$ ；反之， $a_{ij}=a_{ji}=0$ 。图中的点集  $V=\{1, \dots, N\}$ ，其中  $i=1, \dots, l$  表示配电网中能够测量频率的配电变电站， $i=l+1, \dots, N$  表示配电网中的插电式电动汽车。定义集  $V_i=\{j \in V: a_{ij}=1\}$  为节点  $i$  的邻居集合。

为了是用一组频率测量信号  $u_i$  来获取尽可能真实的电网频率偏差值，将利用多智能体系统一致性滤波思想。在这种处理思想下，所测得的频率偏差值在相邻电网元件间进行信息交互，并使得所估计的频率偏差值逐步趋近于真实的电网频率偏差  $\Delta \omega$ 。为此，对于配电网中的每个元件，构建如下的分布式估计器：

$$\dot{x}_i = \frac{1}{T_x} \left[ \sum_{j \in V_i} a_{ij} (x_j(t) - x_i(t)) + \sum_{j \in V_i} a_{ij} k_j (u_j(t) - x_i(t)) + k_i (u_i(t) - x_i(t)) \right], (i=1, 2, \dots, N) \quad (2.11)$$

其中， $x_i$  表示设备  $i$  所估计的频率偏差值； $k_j$  为测量信号反馈增益，对于配电变电站来说， $k_j=1(j=1, \dots, l)$ ，这反映了测量值的影响在频率偏差量的估计中被考虑，对于没有频率测量装置的插电式电动汽车来说， $u_j=0, k_j=1(j=l+1, \dots, N)$ ； $T_x$  表示频率获取时间常量，其反映了时变的测量值与估计信号之间的响应时间，包含了信息发送、传递及接受的耗时。

上述分布式估计器可被等价的表述为如下形式：

$$\dot{x} = -\frac{1}{T_x} (\Xi + \Lambda + K)x + \frac{1}{T_x} (A + I_N)Ku \quad (2.12)$$

其中， $x=(x_1, \dots, x_N)^T$ 、 $u=(u_1, \dots, u_N)^T$ 、 $I_N$  为  $N$  阶单位矩阵， $\Xi$  表示图  $G$  的 Laplacian 矩阵，其连通度为：



$$d_i = \sum_{j=1}^N a_{ij}, i=1, \dots, N \quad (2.13)$$

令  $Q = \frac{1}{T_x}(\Xi + \Lambda + K)$ 、 $K = \text{diag}(k_1, \dots, k_N)$ 、 $\Lambda = \text{diag}(\Lambda_{11}, \dots, \Lambda_{NN})$  表示观测点的连通度，

$\lambda_1, \dots, \lambda_N$  为矩阵  $Q$  的特征值，并按递增排列。

进一步，运用多智能体系统一致性理论，可证得电网中每个节点的频率偏差估计值为全局渐进  $\varepsilon$ -稳定的，即，所有节点的频率偏差估计值收敛至圆心为  $x = \Delta\omega$ ，半径为

$\varepsilon = \frac{1}{T_x} \sqrt{\frac{\beta l}{2\lambda_1}} \max(\Lambda_{ii} + k_i)$  的球中。即，所有节点的频率偏差估计值  $x_i$  满足如下关系式：

$$\lim_{t \rightarrow \infty} \sqrt{\sum_{i=1}^N (x_i(t) - \Delta\omega(t))^2} \leq \varepsilon = \frac{1}{T_x} \sqrt{\frac{\beta l}{2\lambda_1}} \max(\Lambda_{ii} + k_i) \quad (2.14)$$

由此，基于多智能体系统理论体系，运用一致性滤波思想，构建了配电网频率检测机制。

### 第三章 电网多智能体的仿真环境与建模

#### 3.1 电网多智能体仿真平台组成

本论文所提出的电网多智能体仿真<sup>[22]-[24]</sup>平台如图 3.1 所示，由 NETLOGO 与 MATLAB 构成，首先，在 MATLAB 中建立电力系统元件模型；接着，在 NETLOGO 中定义代表电力系统元件的智能体通用模块；仿真时，MATLAB 中计算获得电力系统运行状态数据经 MATLAB 与 NETLOGO 之间的数据交换接口模块发送至 NETLOGO，NETLOGO 中智能体通用模块采用控制电力系统元件的控制策略，对电力系统运行状态数据进行仿真处理，并将结果返回至 MATLAB，作用在对应电力系统元件上；本文针对电力系统的源-网-荷互动技术进行仿真，具有仿真过程直观可见，整个过程中可以在 NETLOGO 中很清楚地看到每个智能体通用模块的状态变化情况；并且所设计的方法中模块具有智能性，能自我应对外界情况的变化，做出积极的反应。

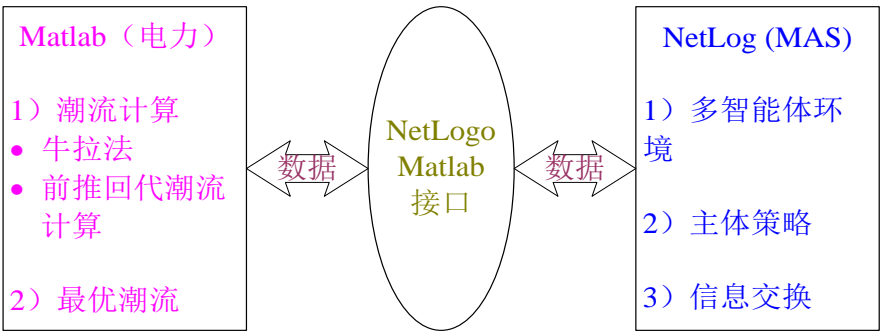


图 3.1 电网多智能体仿真平台组成

##### 3.1.1 NETLOGO 软件部分

NETLOGO 仿真可以规定不同种类主体的个体行为，通过链（link）规定不同主体间的关系，搭建出每个主体有自己独立的行为又同时与其他主体相互联系相互作用的网络。我们可以在模型运行过程中，通过命令中心向模型（海龟、瓦片、链、观察者）发出命令或指令，控制主体的行为，也可以通过滑动条和开关改变模型配置，观察不同情况下模型仿真的情况。

模型的仿真情况，不仅可以在主界面通过 2D 或 3D 模式查看其形象的动态过程，也可以通过监视器和绘图来跟踪我们想要关注的某个主体或主体集合的变化过程。较其他仿真软件而言，NETLOGO 在多智能体方面的应用最大的优点是有很好的演示效果，能够形象的展示

模型仿真的动态过程，并定性定量的分析局部或整体的变化情况。能够使模型在不同场景下自由切换，研究不同情况下模型的不同反应。

NETLOGO 也有不足之处，与 MATLAB 相比，它无法处理较为复杂的数学计算，故采用两者相结合的仿真方案。MATLAB 有许多工具是基于主体的建模人员需要的，包括高级输入能力、统计功能、数据可视化、文档创建等。采用 NETLOGO 与 MATLAB 联合仿真，就可以与 NETLOGO 一起使用这些工具，弥补 NETLOGO 在计算上的不足。

### (1) NETLOGO 模型组成

一个完整模型由以下三个部分组成：命令的執行者主体、模型的控制者按钮、信息的收集者绘图和监视器。

#### (a) 命令的執行者主体

主体：NETLOGO 世界主要是由主体(Agents)组成，主体负责执行操作者下达的指令。每个主体可以同时执行操作者对其下达的不同的指令。NETLOGO 中的主体有海龟、瓦片、链和观察者。其中 NETLOGO 的底层由瓦片组成，而海龟则为一般仿真中的控制对象，它可以在界面中根据控制指令进行变化（如海龟的大小，颜色，位置均可变化）。海龟与海龟之间也可以进行信息通讯，海龟间是否有通讯则取决于海龟与海龟间是否有链连接，有链连接的两个海龟间视为可进行信息通讯。观察者即为操作者。

#### (b) 模型的控制者

NETLOGO 界面中有许多按钮、开关、滑动条和监视器。通过控制这些界面元素让操作者可以控制 NETLOGO 中的各个模型。我们可以在 NETLOGO 界面中设置不同功能的按钮从而对各个模型进行启停等操作。而通过滑动条和开关则可以改变我们对模型中的参数进行快捷修改，从而进行不同参数下的模型仿真，得到各类情况下的仿真结果。NETLOGO 仿真的速度也可以通过滑动条来修改其仿真速度。通过监视器我们可以看到海龟或是瓦片的各项参数配置，观察其变化情况。

#### (c) 信息的收集者

监视器和绘图：其功能是用来显示数据。通过绘图可以显示随着时间推进模型中不同主体的变化趋势，监视器可以显示仿真时间和不同主体中我们关注的的数据。

### (2) NETLOGO 的特点和所提供的功能

#### (a) 跨平台：

NETLOGO 软件是用 Java 编写的，它可以在大部分平台上运行如 Mac, Windows, Linux 等。

#### (b) 环境：

在 NETLOGO 主窗口的顶部是三个标签页：**Interface**(界面)、**Information**(信息)和**Procedures**(例程)，任一时刻只有其中之一可见，但可以通过单击窗口顶部的标签进行切换。

界面页中的大块黑色区域是 2D 视图，它是 NETLOGO 海龟和瓦片世界的可视化表示。通过在视图控制条上单击“3D”按钮打开 3D 视图，这是世界的另外一个可视化表示。界面构建包括按钮、滑动条、开关、选择器、监视器、文本框、注解、输出区，快进滑动条使你可以对模型进行快进和慢放。信息页用来介绍模型、解释如何使用、说明要探索的事情、可能的扩展、NETLOGO 特征等。首次探索模型时很有用。例程页是模型代码存放的地方。对于马上想用的命令，可以在命令中心键入；对于要保存且以后会不断使用的则放在例程页。

### (3) Netlgo 模型实例

我们简单搭建了一个 NETLOGO 生物模型：羊吃草，这是一个捕食者-猎物种群模型。该模型所需完成功能如下：实现海龟移动、进食、繁殖、死亡；实现瓦片长草；给出羊和草数量变化的曲线图。这些功能则需要由以下例程完成：**setup**、**go** (**move-turtles**、**eat-grass**、**reproduce**、**check-death**、**regrow-grass**、**do-plots**)

**setup**：在该例程下，我们首先需要将世界重设为初始、全空状态。所有瓦片变黑。在世界瓦片层上创建一定数量的海龟，同时将海龟的大小，形状，颜色以及在瓦片层上的位置一一进行初始化。然后对瓦片层也进行一个初始化，我们可调节瓦片的颜色，大小以及数量等。每个例程都是由 **to** 开头，以 **end** 结束。

**go**：该例程是实现整个模型的运行，里面包含很多小的例程来完成我们所要达到的功能。

**move-turtles**：该例程让海龟在 360 度内任意选择一个角度，然后往这个方向前进一步。因为 **go** 按钮是永久性的，意味着海龟将不断执行该命令，直到你关掉它（重新点击它）。

**eat-grass**：我们将瓦片颜色设置为绿色，让海龟去判断自己所在的这个瓦片的颜色是什么？如果瓦片颜色是绿色那么就吃掉一个瓦片，瓦片的颜色就会变成黑色，而海龟能量值就会增加一个瓦片所设定的能量值。实现了羊吃草的这个功能。

**reproduce**：每个海龟都有自身的能量值，而我们也会设定一个海龟可以生产一个小海龟所需要达到的能量值。当某一个海龟的能量值达到了可以生产小海龟的时候，世界上就会增加一个小海龟，此时小海龟的能量值就是达到生产的能量值，而原来的那只海龟的能量值就会减去生产所需要达到的那个能量值。完成了海龟繁殖的功能。

**check-death**：当海龟的能量值小于等于零的时候，该海龟在世界上消失。完成了海龟死亡的功能。

**regrow-grass**：让瓦片在 1 到 100 之间随机选择一个数，如果该数值小于 3，则该瓦片的

颜色变成绿色，相当于草生长率是 0.02。该例程完成了草生长的功能。

**do-plots:** 我们要绘制的是绿色瓦片的数量（即草的数量）的曲线图和海龟

数量的曲线图。该绘图功能在绘图编辑框内即可完成，可明显看到羊与草的数量的变化曲线图，如图 3.2 所示，左图为 NETLOGO 世界内动态图截取的某一时刻，其中绿色代表草，而具有羊形状、样色各异的代表羊。在 NETLOGO 的动态图中可形象的看到羊吃草这一过程。而右图则是 NETLOGO 给出的羊与草数量变化曲线，绿色曲线代表的是草数量变化情况，黑色曲线代表的是羊数量变化情况，当程序各项参数配置合理的情况下，草与羊的数量能在某一范围内保持稳定。

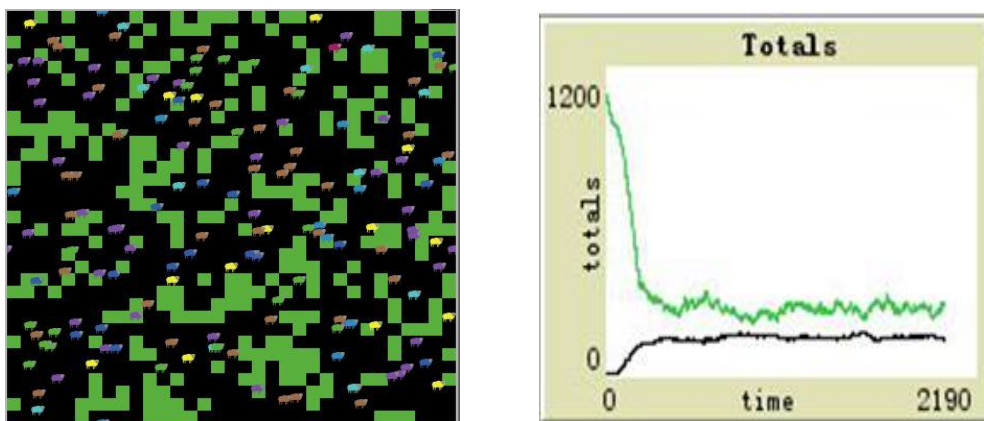


图 3.2 仿真模型及其曲线图

### 3.1.2 MATLAB 软件部分

本文提出的仿真架构中 MATLAB 主要用于电力系统的最优潮流计算，简称 OPF (Optimal Power Flow)。OPF 问题是一个复杂的非线性规划问题，要求在满足特定的电力系统运行和安全约束条件下，通过调整系统中可利用控制手段实现预定目标最优的系统稳定运行状态<sup>[25]</sup>。

#### (1) 最优潮流的模型

本文仿真研究中采用的 OPF 以系统运行成本最小为目标，其数学模型如下：

目标函数：

$$\min. \sum_{i \in S_G} (a_{2i} P_{G_i}^2 + a_{1i} P_{G_i} + a_{0i}) \quad (3.1)$$

式中：  $P_{G_i}$  为第  $i$  台发电机的有功出力；  $a_{0i}$ 、  $a_{1i}$ 、  $a_{2i}$  为其耗量特性曲线参数。

约束条件：

$$P_{G_i} - P_{D_i} - V_i \sum_{j=1}^n V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) = 0 \quad (i \in S_B) \quad (3.2)$$

$$Q_{G_i} - Q_{D_i} + V_i \sum_{j=1}^n V_j (G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij}) = 0 \quad (i \in S_B) \quad (3.3)$$

$$\underline{P_{G_i}} \leq P_{G_i} \leq \overline{P_{G_i}} \quad (i \in S_G) \quad (3.4)$$

$$\underline{Q_{R_i}} \leq Q_{R_i} \leq \overline{Q_{R_i}} \quad (i \in S_R) \quad (3.5)$$

$$\underline{V_i} \leq V_i \leq \overline{V_i} \quad (i \in S_B) \quad (3.6)$$

$$|P_l| = |P_{ij}| = |V_i^2 G_{ij} - V_i V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij})| \leq \overline{P_l} \quad (l \in S_l) \quad (3.7)$$

式 (3.2) -(3.3) 为功率平衡方程；式 (3.4) 为电源有功出力上下限约束，式 (3.5) 为无功出力上下限约束，式 (3.6) 为节点电压上下限约束，式 (3.7) 为线路潮流约束。式中：\$S\_B\$ 为系统所有节点集合，\$S\_G\$ 为所有发电机集合，\$S\_R\$ 为所有无功源集合，\$S\_l\$ 为所有支路集合；\$P\_{G\_i}\$、\$Q\_{G\_i}\$ 为发电机 \$i\$ 的有功和无功出力；\$P\_{D\_i}\$、\$Q\_{D\_i}\$ 为节点 \$i\$ 的有功和无功负荷；\$V\_i\$、\$\theta\_i\$ 为节点 \$i\$ 电压幅值与相角，\$\theta\_{ij} = \theta\_i - \theta\_j\$；\$G\_{ij}\$、\$B\_{ij}\$ 为节点导纳矩阵第 \$i\$ 行第 \$j\$ 列元素的实部和虚部；\$P\_l\$ 为线路 \$l\$ 的有功潮流，设线路 \$l\$ 两端节点为 \$i, j\$。

## (2) 最优潮流的算法

目前已有的求解 OPF 问题的方法有非线性规划法、二次规划法、线性规划法、混合规划法以及今年出现的内点算法和人工智能方法等。其中内点算法因计算速度快，收敛性强同时能满足大规模电网的计算要求等优点受到广泛的应用。本文提出的仿真框架中，对给定的 OPF 模型采用内点算法进行 OPF 计算。

为了便于讨论，将 OPF 模型简化为以下一般非线性模型：

目标函数：

$$\min . f(x) \quad (3.8)$$

约束条件：

$$h(x) = 0 \quad (3.9)$$

$$\underline{g} \leq g(x) \leq \overline{g} \quad (3.10)$$

其中，式 (3.8) 为目标函数，对应于 OPF 模型中式 (3.8) 是一个非线性函数；式 (3.9) 为非线性等式约束条件，对应于 OPF 模型中式 (3.2) -(3.3)，式 (3.10) 为非线性不等式约束。

假设模型中有  $n$  个变量,  $m$  个等式约束,  $r$  个不等式约束。本文采用的内点法的基本思路如下。

首先, 把不等式约束化为等式约束:

$$g(x) + u = \bar{g} \quad (3.11)$$

$$g(x) - l = \underline{g} \quad (3.12)$$

其中松弛变量  $l = [l_1, \dots, l_r]^T$ ,  $u = [u_1, \dots, u_r]^T$ , 应满足

$$l > 0, \quad u > 0 \quad (3.13)$$

这样, 原问题变为优化问题 A:

目标函数:

$$\min . f(x) \quad (3.14)$$

约束条件:

$$h(x) = 0 \quad (3.15)$$

$$g(x) + u = \bar{g} \quad (3.16)$$

$$g(x) - l = \underline{g} \quad (3.17)$$

$$l > 0, \quad u > 0 \quad (3.18)$$

然后把目标函数改造为障碍函数, 该函数在可行域内应近似于原目标函数而在边界时变得很大。因此可得到优化问题 B:

目标函数:

$$\min . f(x) - \mu \sum_{j=1}^r \log(l_j) - \mu \sum_{j=1}^r \log(u_j) \quad (3.19)$$

约束条件:

$$h(x) = 0 \quad (3.20)$$

$$g(x) + u = \bar{g} \quad (3.21)$$

$$g(x) - l = \underline{g} \quad (3.22)$$

其中扰动因子 (或称障碍常数)  $\mu > 0$ 。优化问题 B 只含等式限制, 因此可以直接使用拉格朗日乘子法来求解。

OPF 内点算法流程图如图 3.3 所示, 其中初始化部分包括:

设置松弛变量  $l$ 、 $u$ ，保证  $[l, u]^T > 0$ 。

设置拉格朗日乘子  $z$ 、 $w$ 、 $y$ ，满足  $[z > 0, w < 0, y \neq 0]^T$ 。

设优化问题各变量的初值。

取中心参数  $\sigma \in (0, 1)$ ，给定计算精度  $\varepsilon = 10^{-6}$ ，迭代次数初值  $k = 0$ ，

最大迭代次数  $K_{\max} = 50$ 。

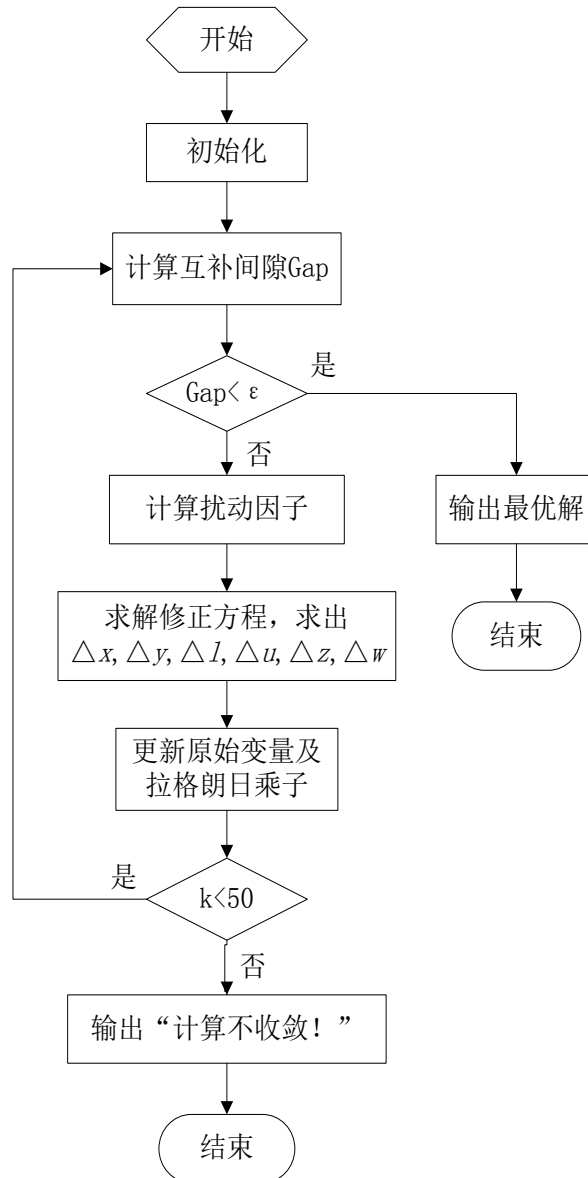


图 3.3 OPF 内点算法流程图

### 3.1.3 NETLOGO 与 MATLAB 接口部分

MATLAB 与 NETLOGO 接口实现 MATLAB 与 NETLOGO 之间的信息通讯。本文



MATLAB 与 NETLOGO 接口程序用 java 语言编写。在接口程序中规定 NETLOGO 调用格式, 从而 NETLOGO 在进行仿真时使用接口中规定的调用命令即可调用 MATLAB。接口命令语句及其主要功能如下:

**eval:** 允许用户在 NETLOGO 中运行任何 MATLAB 命令, 包括 m 文件和在 MATLAB 路径下的程序包。其输入为一个字符串, 代表的是 MATLAB 中固定的命令语言。需要注意的是该命令不能创建变量, 同时该接口也不支持矩阵模型。如果使用 eval 命令, 它发送给 MATLAB 命令比较慢, NETLOGO 将不会等待结果。如果你想要 NETLOGO 等待 MATLAB 完成后在启动, 则你需要在 NETLOGO 中设置一个等待命令。如 MATLAB: eval "someFunction ()"。

**send-string:** 输送一个字符串变量到 MATLAB 中。输入为一个字符串, 并且该变量名称在 MATLAB 中被保存。如 MATLAB: send-string"varName""a"。

**send-double:** 输送一个 "double" 类型的变量到 MATLAB 中。输入为 "double" 类型的变量, 它是 NETLOGO 中的数值, 该变量名保存在 MATLAB 中。如 MATLAB: send-double"varName" 12。

**send-string-list :** 输送一系列的字符串变量到 MATLAB 中。如 MATLAB: send-string"varName" (list "a""Bily")。

**send-double-list:** 输送一系列的 "double" 类型的变量到 MATLAB 中。如 MATLAB: send-double"varName" (list 12 22 23)。

**get-string:** 回复一个保存在 MATLAB 中的字符串到 NETLOGO 中。如 set myString MATLAB:get-string"varName"。

**get-double:** 回复一个保存在 MATLAB 中的 "double" 类型的值到 NETLOGO 中。如 set myNumber MATLAB:get-double"varName"。

**get-string-list:** 回复一系列保存在 MATLAB 中的字符串到 NETLOGO 中。如 set myString MATLAB:get-string-list "varName"。

**get-double-list:** 回复一系列保存在 MATLAB 中的 "double" 类型的值到 NETLOGO 中。如 set myList MATLAB:get-double-list"varName"。

我们给出一个简单的 NETLOGO 与 MATLAB 联合实例。该实例是让 NETLOGO 中建立的海龟的变化情况由 MATLAB 中的函数来控制。

以下是本文实现的 NETLOGO 中调用 MATLAB 的程序, 简单分析其语言特点:

```
ask microplayers[ MATLAB:send-double"sendm" m ]  
MATLAB:eval"co(sendm)"
```

```
ask microplayer 20[ set size MATLAB:get-double"d" set color x]
ask microplayer 23[ set size MATLAB:get-double"d" set color y]
MATLAB:eval"si(sendm)"
ask microplayer 21[ set size MATLAB:get-double"c" set color x]
ask microplayer 24[ set size MATLAB:get-double"c" set color y]
MATLAB:eval"sincos(sendm)"
ask microplayer 22[ set size MATLAB:get-double"z" set color x]
ask microplayer 25[ set size MATLAB:get-double"z" set color y]

set m m + 2 * pi / 50

set x x + 1

set y y + 1
```

该程序中发送给 MATLAB 的变量名称是 sendm，而该变量中的值则是有 NETLOGO 提供（即为 m 的值）。m 值的变化情况可以看到是执行一步其值则增加  $2 * \pi / 50$ 。而调用的 MATLAB 函数有三个分别是 co(sendm)、si(sendm)、sincos(sendm)。第一个函数是一个 cos 函数，第二个函数是一个 sin 函数，第三个函数则是一个 sin+cos 的函数。最后的结果则是海龟 20 和 23 大小按照 sin 函数变化，海龟 21 和 24 的大小是按照 cos 函数变化，海龟 22 和 25 的大小按照 sin+cos 函数变化。这就完成了 NETLOGO 对 MATLAB 的一个调用如图 3.4 所示。

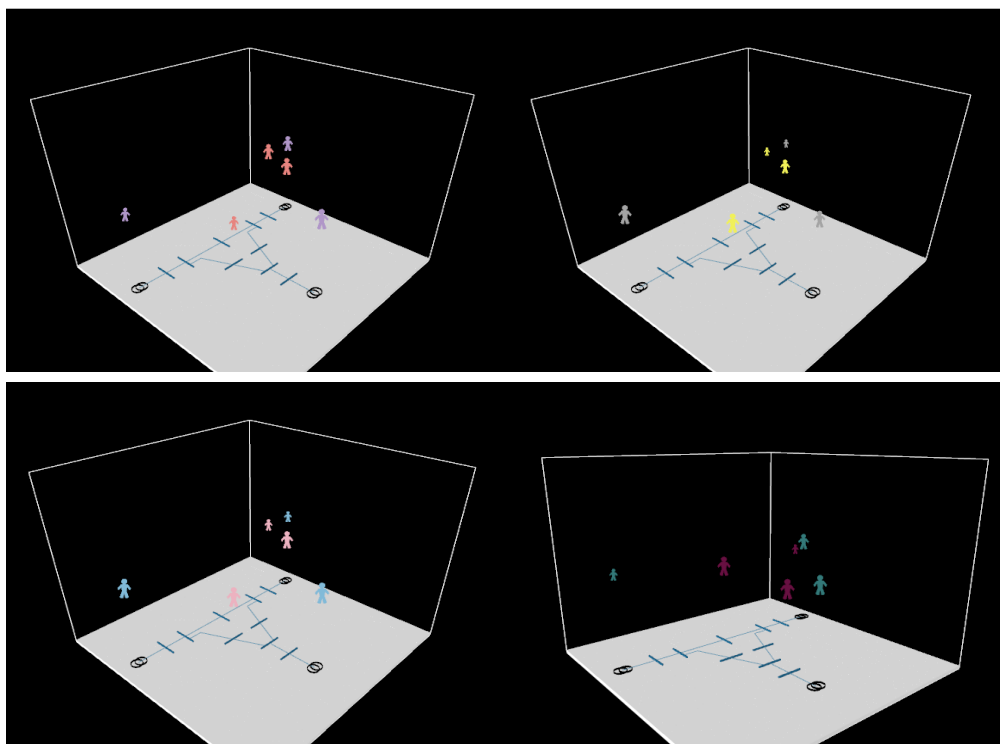


图 3.4 NETLOGO 调用 MATLAB 实例模型图

### 3.2 电网多智能体的仿真环境

### 3.2.1 电网多智能体仿真概述

基于 NETLOGO 与 MATLAB 联合的仿真系统<sup>[26]~[27]</sup>如图 3.5 所示。在 MATLAB 上搭建完整的电力网络，包括各电网拓扑结构、发电机的输出功率，线路潮流，电压等级，频率、负载等。基于确定的电力网络进行相关计算，得到电网运行状态信息量，如频率、电压、电价等。在 NETLOGO 中搭建仿真环境，整个仿真环境分为三层：本地响应层、代理协调层、拓扑结构层。本地响应层是对负荷群进行建模，即对不同种类的负荷分别搭建通信属性子模块、智能属性子模块、物理属性子模块。代理协调层是对 Agent 建模，确定每个 Agent 管辖的负荷群。拓扑结构层是给出本次仿真所采用的网络拓扑结构。

MATLAB 是整个系统的一个虚拟调度中心，调度中心计算频率、电压和电价等网络数据，并给 Agents 下发该时段频率、电压和电价等信息。NETLOGO 中负荷代理（Agents）接收调度中心下发的牵引信号，解析该指令，同时下发给所管辖的负荷群，负荷根据收到牵引信号，以自身目标最大化和为电网服务为目标，调整工作状态和功率需求。负荷响应完成后，将自身功率需求上报给 Agents，由 Agents 上发给调度中心，以便调度中心计算下一时刻的牵引信号量。如此循环重复，直至整个系统达到稳定。

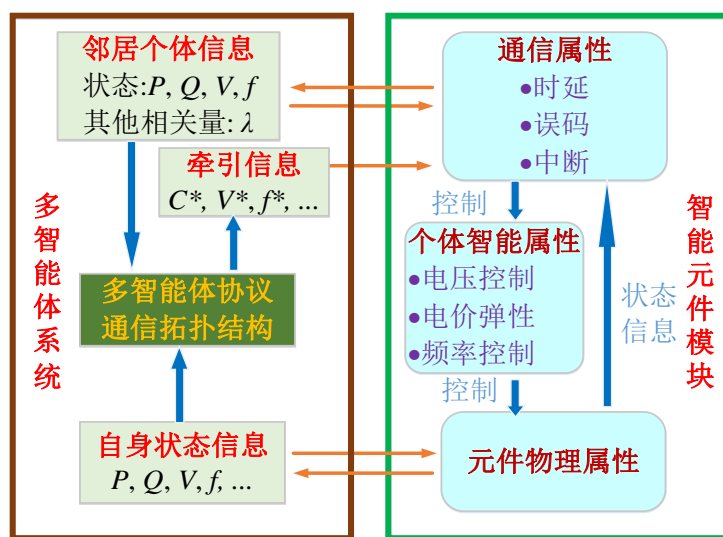


图 3.5 基于 NETLOGO 与 MATLAB 联合的仿真环境总体框架

### 3.2.2 基于 NETLOGO 的仿真环境搭建

本文提出的基于 MATLAB 和 NETLOGO 的电力系统联合仿真建模方法中, 所基于的系

统结构是一种基于 MATLAB 和 NETLOGO 的电力系统仿真系统，NETLOGO 作为该软件系统的前台，也是交互界面，在 NETLOGO 中定义电力系统中的各种元件模块，称之为智能体通用模块，如图 3.6 所示，智能体通用模块包含三个属子模块：通信属性子模块、智能属性子模块、物理属性子模块；其中，每个智能体通用模块都有两组接口，分别为物理接口和通信接口。物理接口是智能体通用模块与物理实体即 MATLAB 连接的接口，负责获取和发送本模块电力元件的信息数据；通信接口是智能体通用模块与相邻智能体通用模块的数据交换接口。NETLOGO 中是由智能体通用模块按通信拓扑组成的网络，仿真的结果可以很直观的反映在 NETLOGO 中；MATLAB 作为该软件的后台，用于搭建完整的电力系统元件模型，包括电力系统运行模型及控制策略算法等，负责电力系统的仿真运算，即模拟电力系统实体的运行，并将结果交给 NETLOGO。

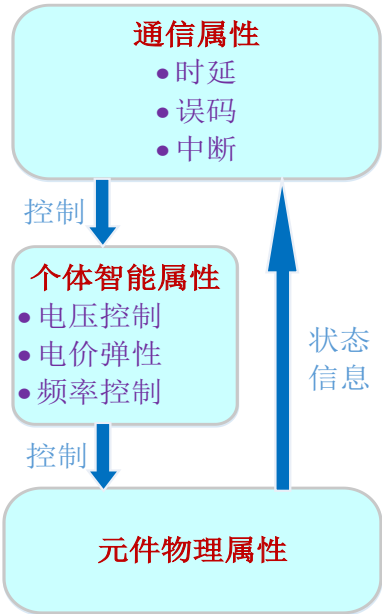


图 3.6 负荷通用模型

(1) 智能属性子模块

如图 3.7 所示，智能属性子模块是使智能体通用模块具有智能性的关键，智能属性子模块用于描述电力系统元件制定决策的过程，其中，智能属性子模块包括知识库子模块、决策库子模块和推理机。其中，知识库子模块是电力系统运行要求规则的集合，库中有智能体的物理特性以及在不同条件下智能体的行为积累，相当于是一个经验库，就电网功率平衡控制而言，知识库中的累积的数据是通过对采集的数据（新能源输出功率波动范围、波动频率、与时间和季节等的关系，以及柔性负荷的用电规律）分析而得到的。决策库子模块是电力系统控制决策规定过程的集合，用 if...then 来描述。推理机用于电力系统根据知识库子模块和

决策库子模块中的规则控制电力系统元件的控制策略。电力系统元件会基于对决策过程自组织行为机制的认识优化调整决策集，各种不同的电力系统元件的决策库必然是不同的。

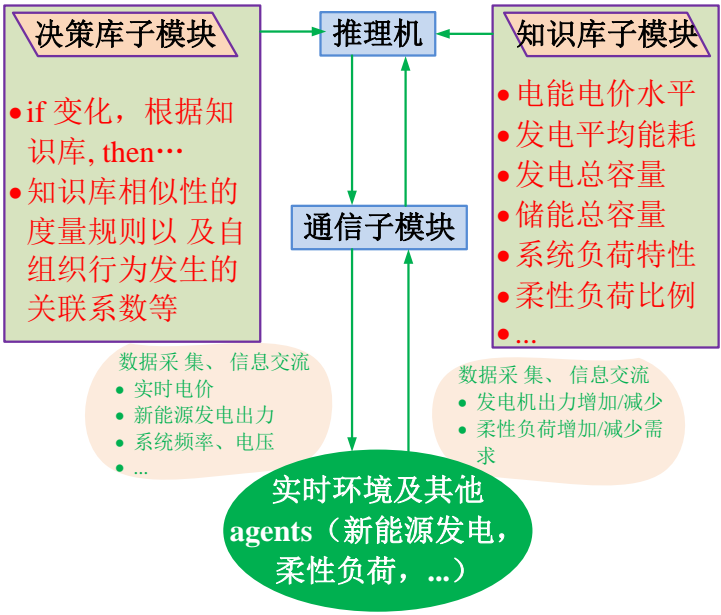


图 3.7 负荷模型的智能属性

(2) 物理属性子模块

物理属性子模块用于定义电力系统元件的运行状态，具体用于定义电力系统元件的运行状态，包括电力系统元件的控制量、控制参数和控制状态变化规律；物理属性子模块采用继承式的类定义，例如电力系统物理元件可分为电源、电网、负荷等，电源中又有常规发电机、风力发电机和光伏发电等，同一父类的几种子类之间的属性定义采用冗余措施，即枚举其特征属性，在区分时使用开关量选取需要的特征属性；物理属性子模块还定义该物理实体的输入输出量，例如常规发电机的输入为频率，输出为功率；负荷的输入为频率、电价，输出为功率；同时，物理属性子模块通过物理接口获取自身的状态信息，即 MATLAB 的仿真计算结果，或是将智能元件执行决策后的状态信息传给 MATLAB 进行计算。

(3) 通信属性子模块

通信属性子模块用于定义电力系统元件之间的通信拓扑结构，同时模拟电力系统元件之间的信息通讯情况，通信属性子模块是电力系统元件进行数据采集、信息交流的通信接口，模拟通信过程中的延时、误码和中断等情形，并且电力系统元件之间交流的形式为问答式。

(a) 延时

Agents 和外界进行信息传输中，可能会存在发送时延、传播时延、转发时延等，造成 Agents 不能获得正确的信息。如图 3.8 所示，Agent j 在当前时段获得的其他 Agents 的信息因为延时

可能并非本时段的，可能是上一个时段或者是更早之前的信息。

为解决信息传输的时延问题，我们让每个 Agents 的数据带带有时标（同一时间段的时标则相同），而 Agents 彼此之间进行信息交互则采用问答机制，如 Agent  $j$  需获得  $t$  时段的信息，则会告知其他 Agents 所要获得  $t$  时段的信息，其他 Agents 再发送  $t$  时段的信息给 Agent  $j$ 。

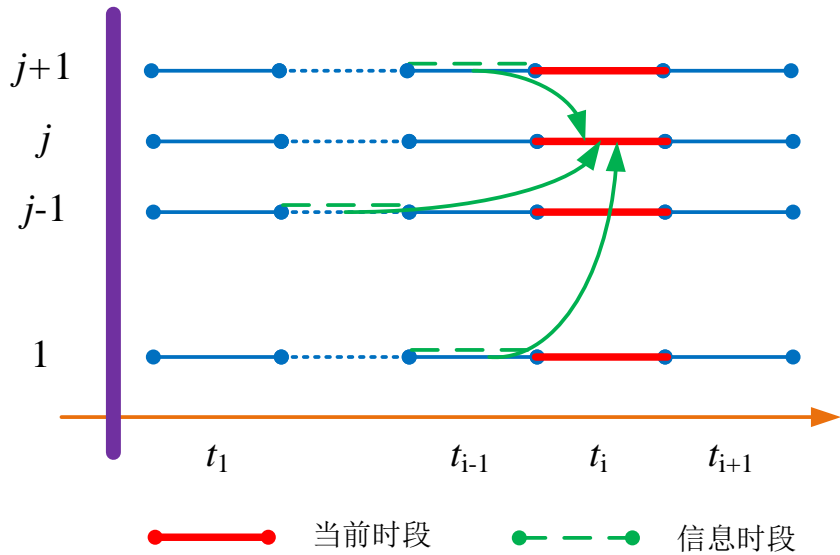


图 3.8 Agent  $j$  获得信息延时图

(b) 误码

在通信网络中，Agents 与外界交互的信息都是由二进制代码表示的字母、数字或控制符号。数据通信系统在正常工作情况下信息传输会出现错误，其错误率即为误码率。信息的单位可以是比特、码元、码字、一组码字，因此，出错率表示方法也不同。接收的错误比特数占传输总比特数的比例即为误比特率，其余还有误码率、误字率、误组率，定义与误比特率相同。

(c) 中断

通信系统中存在较多中断源，如通信中断、输入/输出中断和时基中断等等。为处理中断事件需事先编写好中断程序，中断事件发生时系统调用中断程序。

3.2.3 基于 MATLAB 的仿真环境

本文仿真中采用系统的电价为牵引信号，该电价为实时电价，调度中心根据电力供需状况每五分钟实时制定并发布电价。电价制定的建模框图如图 3.9 所示，根据电力系统网络结构，在 MATLAB 中建立电力系统元件模型，包括电力系统元件之间的电气关联关系，并制定

南京邮电大学硕士研究生学位论文 第三章 电网多智能体的仿真环境与建模

控制策略，同时采用相应的算法对方程进行求解。MATLAB 将求解得到的电价等相关信息量通过 MATLAB 与 NETLOGO 接口下达给所有负荷，所有负荷根据自身特性做出响应，响应量再通过 MATLAB 与 NETLOGO 接口上传给 MATLAB，改变原先网络中的参数，进行下一步求取电价的运算，直至系统稳定。

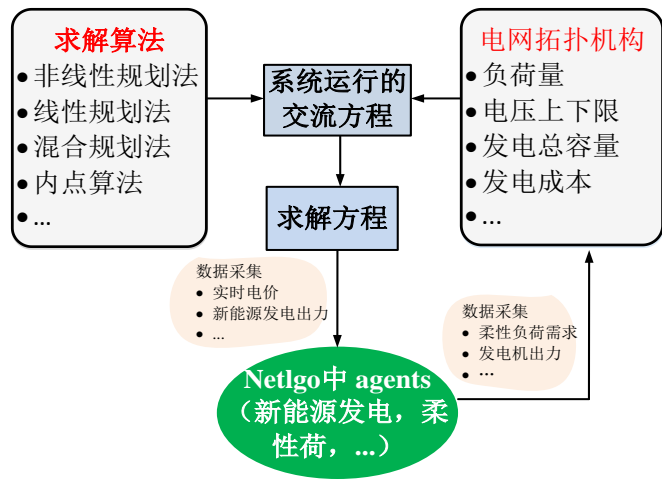


图 3.9 MATLAB 中电价制定建模框图

(1) 电价制定模型

本论文仿真研究中采用的系统运行交流方程以系统的运行成本最小为目标，其模型如图 3.10 所示，由数据采集模块、最优目标模块、物理约束模块组成。其中数据采集模块对负荷、发电机、母线等信息进行采集。最优目标模块设定系统运行的优化目标，而本文设定系统运行目标为发电机运行成本总和最小。物理约束模块给出满足电网系统安全、稳定、可靠同时经济运行的约束条件，例如发电机自身功率平衡约束、有功出力上下限约束，无功出力上下限约束、节点电压上下限约束以及线路潮流约束等。以上三个基本模块构成 MATLAB 中电价制定模型，基于以上模型可求得各个发电机出力以及各个节点电价等数据。

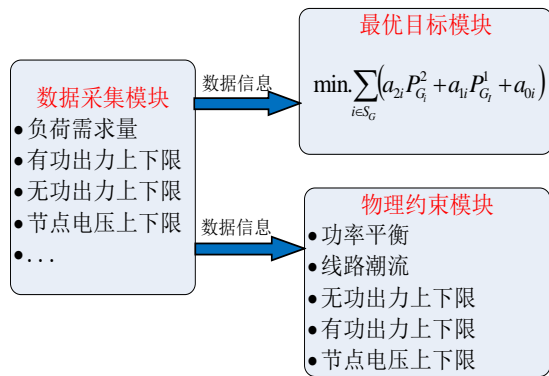


图 3.10 电价制定模型



（2）内点法求解

本文提出的仿真框架中，对给定的电价制定模型采用内点算法进行求解。如图 3.11 所示，首先引入松弛变量将电价制定模型中的不等式约束化为等式约束，使得原模型中所有约束都变为等式约束。然后加入扰动因子（或称障碍常数）把目标函数改造为障碍函数，该函数在可行域内应近似于原目标函数而在边界时变得很大。此时优化问题可以直接用拉格朗日乘子法来求解。

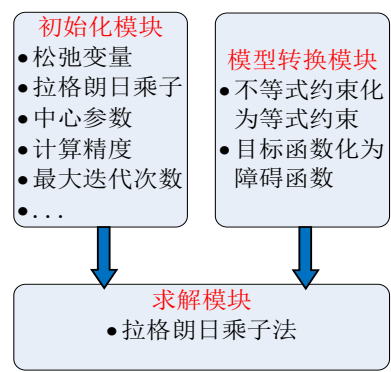


图 3.11 内点法求解模型

3.2.4 NETLOGO 和 MATLAB 接口环境搭建

在以电价为牵引信号的仿真中，从仿真结果可看出电价信息是储存在 Bus 矩阵的第 14 列，Bus 矩阵如表 3.1 所示。根据电网初始环境，MATLAB 进行一次潮流运算得到此时电网电价，NETLOGO 调用该初始电价，将该电价发布给系统内的所有用户，不同的节点上的用户接受到的节点电价均不相同，每个用户根据自身目标限定修改自己的用电模式，从而整个系统的负荷需求受到一定的改变。NETLOGO 通过接口程序再将改变后的系统总需求发送给 MATLAB 进行下一次的潮流运算。从中可知 NETLOGO 与 MATLAB 接口主要负责进行电价与系统总需求信息的交换，NETLOGO 调用接口进行数据交换的部分代码如表 3.2 所示。

表 3.1 三机九节点 bus 矩阵

Bus	Type	Pd (MW)	Qd (MVar)	Gs (pu)	Bs (pu)	Vm (pu)	Vmag (pu)
1	3	0	0	0	0	1	1.1000
2	2	0	0	0	0	1	1.0974
3	2	0	0	0	0	1	1.0866



4	1	0	0	0	0	1	1.0942
5	1	90	30	0	0	1	1.0844
6	1	0	0	0	0	1	1.1000
7	1	100	35	0	0	1	1.0895
8	1	0	0	0	0	1	1.1000
9	1	125	50	0	0	1	1.0717

承接上表

Vang (deg)	BaseKV (MW)	Va (pu)	Vmax (pu)	Vmin (pu)	Lambda (\$/MVA-hr)
0	345	1	1.1000	0.9000	24.7557
4.8931	345	1	1.1000	0.9000	24.0345
3.2490	345	1	1.1000	0.9000	24.0759
-2.4631	345	1	1.1000	0.9000	24.7559
-3.9824	345	1	1.1000	0.9000	24.9985
0.6024	345	1	1.1000	0.9000	24.0759
-1.1968	345	1	1.1000	0.9000	24.2539
0.9051	345	1	1.1000	0.9000	24.0345
-4.6156	345	1	1.1000	0.9000	24.9985

表 3.2 代码描述

MATLAB 与 NETLOGO 之间接口通信

1: MATLAB:send-double"p" p	——NETLOGO 发送功率值给 MATLAB
2: MATLAB:eval "b.bus(i,j)=p"	——改变 MATLAB 中电网负荷
3: MATLAB:eval"d=loadcase(runPri(b))"	——进行电价求解运算
4: MATLAB:eval "c=d.bus(i,j)"	——得到最新电价
5: ask turtle m [ set c MATLAB:get-double "c"]	——负荷代理调用最新电价

3.3 电网多智能体元件的建模

3.3.1 电网多智能体元件建模概述

关于电网多智能体建模总体框架如图 3.12 所示。具体工作主要分为两个部分：协调控制和物理层建模。而物理层建模对象有三个部分：1、电源；2、负荷；3、电网。争对电源建模，我们采用 MATPOWER 工具包中的电源模块，在 MATLAB 对电源进行控制，这里对电源建模不做详细说明；对负荷也分为两类：刚性负荷（负荷量不受影响，保持恒定）、受电价响应的负荷。

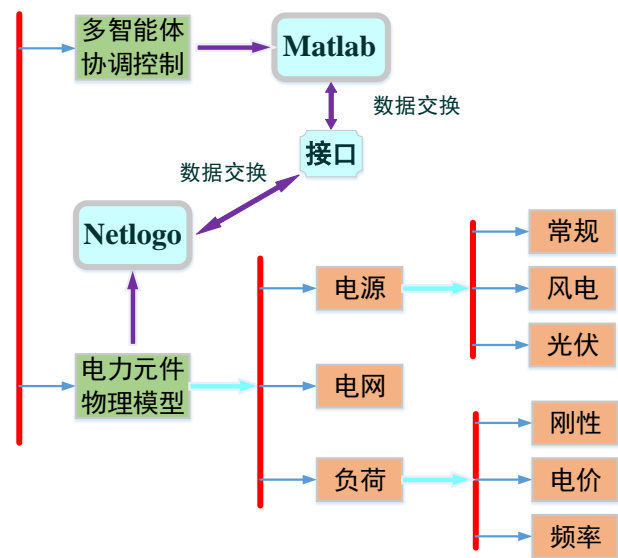


图 3.12 电网多智能体元件建模框架

3.3.2 对响应电价的负荷建模

(1) Prosumer 建模

电力系统中存在多种多样的负荷和电源，本文采用四类典型的负荷和电源：刚性负荷、柔性负荷、分布式电源、储能元件。针对负荷和电源不同特性分别进行建模。负荷耗电和电源发电的电价标么值分别为 $[\rho_k, v_k]$ 。在本文所提的仿真架构中，电价对 $[\rho_k, v_k]$ 即为 Agent 下达给各个负荷的牵引信号<sup>[28]</sup>。对于不同母线上的负荷而言，每对电价可能是不同的。

在本论文中，假定采用的四类典型负荷和电源以追求经济性和舒适度为目标。我们将该种类型的负荷称之为 Prosumer。在现有文献中，负荷的电价响应特性和电源的发电安排是不考虑舒适度的。因此，考虑舒适度后，负荷的需求无法根据传统模型准确预测，而电源的出

力也不能根据传统的发电安排方法得到。负荷和电源追求各自目标的行为可以等效为对应的 2-D 空间：1)对经济性的倾向度  $\mu_k$ ，表现在一方面避免消费成本的最大化，另一方面获得最大化经济收益。2)对舒适度的倾向度  $\varphi_k$ ，表现个体考虑自身欲望和意愿，他们会使用装置或设备来满足他们的生活标准（生理方面）。每个负荷的目标行为特征用  $A_k = [\mu_k, \varphi_k]$  描述， $A_k$  在经济性和舒适度两个方面被赋予不同的值，如图 3.13 所示。

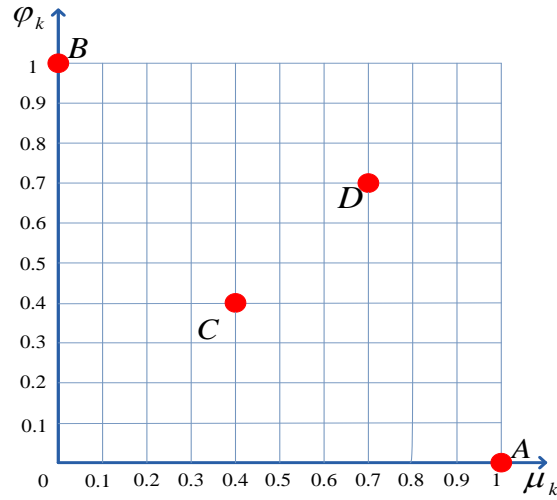


图 3.13 负荷  $k$  标准化的行为空间示意图

负荷从用电中获得的效能可以在经济效益方面和舒适度方面进行量化，这两个方面的价值取决于个体对功率输入输出方面的行为模式。我们定义四类负荷的功率输入输出为：

(a) 刚性负荷：负荷量  $q_k$  不随电价改变；

(b) 柔性负荷：

$$d_k = D_k \cdot [1 - \rho_k \cdot \mu_k \cdot (1 - \varphi_k)] \quad (3.23)$$

其中  $d_k$  为负荷需求量， $D_k$  为负荷参考功率， $\rho_k$  为负荷买电的价格。

(c) 分布式电源：

$$g_k = G_k \cdot [1 - \nu_k \cdot \mu_k \cdot (1 - \varphi_k)] \quad (3.24)$$

其中  $g_k$  为分布式电源的发电量， $G_k$  为分布式电源参考功率， $\nu_k$  为分布式电源卖电的价格。

(d) 储能元件充电时为：

$$d_k = D_k \cdot [1 - \rho_k \cdot \mu_k \cdot (1 - \varphi_k)] \quad (3.25)$$

放电时为：

$$g_k = G_k \cdot [1 - \nu_k \cdot \mu_k \cdot (1 - \varphi_k)] \quad (3.26)$$

负荷的网络输出功率是对参考功率的需求计算而来的。在此模型中负荷的参考功率是恒定的，不涉及技术方面的问题。负荷-电价响应特性由式中的状态参数  $\mu_k$ ， $\varphi_k$  决定，其中所述与价格  $\rho_k$ ， $\nu_k$  相关的需求减少率和生产增长率是  $\mu_k \cdot (1 - \varphi_k)$ ，从而引起功率的弹性变化。此外，与传统的基于固定响应的模型相反，我们的方法中，社会行为被明确建模，综合考虑了由于社会行为的交互所带来的弹性变化。

负荷在空间的状态位置如图 3.13 所示。如果负荷  $k$  在位置  $A_k$  处，表示该负荷只考虑经济利益： $\rho_k(\nu_k)$  的增长会导致功率输出的减少（增加）。位置  $B$  意味着它无法根据价格的改变而改变电能的输出或输入量，它考虑的是舒适度。与这两种情况相比，不在边界上的位置，其价格与一定程度的经济利益和舒适度都相关。例如，负荷  $k$  位于点  $C$ ，从经济性而言，在价格  $\rho_k = 1$  时，功率需求减少 0.3，从舒适度而言，功率消耗将减少 0.3。这就意味着在价格  $\rho_k = 1$  时最终需求将减少  $0.3 \cdot (1 - 0.3)$ 。同理在点  $D$  处，在价格  $\rho_k = 1$  时最终需求将减少  $0.7 \cdot (1 - 0.7)$ 。在价格  $\nu_k = 1$  时最终发电量将减少  $0.7 \cdot (1 - 0.7)$ 。从理论上而言， $\rho_k$  和  $\nu_k$  可以单独设置。但为防止负荷套利，我们假设  $\rho_k = -\nu_k$ 。

作为负荷的收益，经济效益  $B_k$  定义如下：

$$B_k = -\rho_k \cdot E_k = -\rho_k [D_k \cdot [1 - \rho_k \cdot \mu_k \cdot (1 - \varphi_k)] - G_k \cdot [1 - \nu_k \cdot \mu_k \cdot (1 - \varphi_k)]] \quad (3.27)$$

其中  $E_k$  为净输入输出的总和。

负荷舒适度定义如下：

$$C_k = \varphi_k \cdot (1 - \mu_k) \quad (3.28)$$

负荷的整体效用由两个目标函数加权得到总目标函数表示，总目标函数定义如下：

$$R_k = \mu_k \cdot B_k + \varphi_k \cdot C_k \quad (3.29)$$

## (2) 空调建模

制冷空调的运行规则：设定房间的温度值和温度变化的限制范围。当室内温度超过最大

允许温度  $T_{sp} + \frac{\Delta T_{ac}}{2}$  时，制冷空调开启。当室内温度低于  $T_{sp} - \frac{\Delta T_{ac}}{2}$  时，该房间的空调关闭。当室内温度在预设的温度范围内  $\left[ T_{sp} - \frac{\Delta T_{ac}}{2}, T_{sp} + \frac{\Delta T_{ac}}{2} \right]$  时，制冷空调将保持其之前的运行状态。如下式所示：

$$\beta_{ac,t} = \begin{cases} 1 & T_{ac,t} > T_{sp} + \frac{\Delta T_{ac}}{2} \\ 0 & T_{ac,t} < T_{sp} - \frac{\Delta T_{ac}}{2} \\ \beta_{ac,t-1} & T_{sp} - \frac{\Delta T_{ac}}{2} \leq T_{ac,t} \leq T_{sp} + \frac{\Delta T_{ac}}{2} \end{cases} \quad (3.30)$$

式中：  $T_{sp}$  为房间温度设定值，假设用户适合的温度区间为：  $[T_{sp,\min}, T_{sp,\max}]$ ，则  $T_{sp} \in [T_{sp,\min}, T_{sp,\max}]$ ；  $\Delta T_{ac}$  为温度设定的变化范围；  $T_{ac,t}$  在时间  $t$  时室内的温度；  $\beta_{ac,t}$  在时间  $t$  时制暖空调的运行状态（0=OFF；1=ON）。

设定初始室外温度，同时每台空调确定初始设定温度，从而可以得到空调在一个运行周期内空调关闭与运行的时间分别为：

$$\tau_0 = \frac{RC}{\Delta t} \ln \left( \frac{T_{sp} - \frac{\Delta T_{ac}}{2} - T_{out}}{T_{sp} + \frac{\Delta T_{ac}}{2} - T_{out}} \right), \quad \beta = 0 \quad (3.31)$$

$$\tau_1 = \frac{RC}{\Delta t} \ln \left( \frac{PR + T_{sp} + \frac{\Delta T_{ac}}{2} - T_{out}}{PR + T_{sp} - \frac{\Delta T_{ac}}{2} - T_{out}} \right), \quad \beta = 1 \quad (3.32)$$

式中：  $\tau_0$  为空调在一个周期内的关断时间；  $\tau_1$  为空调在一个周期内的开启时间；  $R$  为空调等效热阻；  $C$  为空调等效热容；  $P$  为空调的制冷/制热功率；  $\Delta t$  为仿真时间间隔；  $T_{out}$  为室外温度。

进而可得到线性化的空调等效参数模型，即状态队列模型：

$$T_i^{t+1} = T_i^t + \frac{\Delta t}{\tau_0} \cdot \Delta T_{ac}, \quad \beta = 0 \quad (3.33)$$

$$T_i^{t+1} = T_i^t - \frac{\Delta t}{\tau_1} \cdot \Delta T_{ac}, \quad \beta = 1 \quad (3.34)$$

式中：  $T_i^t$  为室内气体在  $t$  时刻的温度。

图 3.14 为某一台空调机组的运行状态队列图，图中  $\tau_0 = 10 \text{ min}$ ，  $\tau_1 = 5 \text{ min}$ 。我们将其一个

运行周期等分为 15 份，分别代表 15 个状态。将这 15 个状态进行编号，其中状态 1 至 10 表示的状态为空调关闭，而 11 至 15 号表示的状态为空调运行。在没有人为改变空调设定温度或关闭空调时，该空调则会一直在状态 1 到状态 15 依次切换，并循环该周期运行，从而保证室内温度一直处于  $\left[T_{sp} - \frac{\Delta T_{ac}}{2}, T_{sp} + \frac{\Delta T_{ac}}{2}\right]$  范围内。

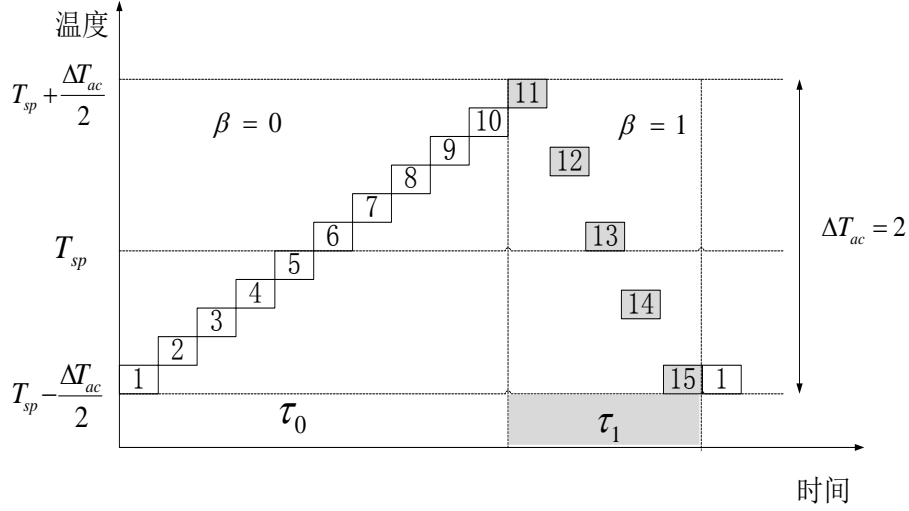


图 3.14 空调状态队列模型示意图

本文进行的空调仿真系统中，每台空调的初始状态随机分布在上文提到的 15 个状态中，即在给定室外温度  $T_{out}$  及温度限值  $\left[T_{sp} - \frac{\Delta T_{ac}}{2}, T_{sp} + \frac{\Delta T_{ac}}{2}\right]$  的情况下每台空调在其一个运行周期内关机时间为 10 min，启动时间为 5 min，而我们已将空调运行周期分为 15 个状态，从而可知每个状态持续的时间为 1 min。由此可得聚合空调负荷的计算公式：

$$P_{sum} = \frac{n_l}{n_c} \cdot N \cdot P \quad (3.35)$$

式中： $n_l$  为该仿真时刻所有运行状态为开启的空调的个数； $n_c$  为空调运行周期的人为划分的总状态数； $N$  为所有仿真的空调总个数。从而就可以通过在某一时刻处于开启状态中的空调个数计算得到此刻的空调聚合功率。

仿真输入信号为电价，随着电价的改变，用户设定的空调温度  $T_{sp}$  随之改变。空调设定温度与电价间的关系如下：

$$T_{sp} = T_0 + \frac{p - p_0}{k} \quad (3.36)$$

式中  $T_0$  为空调初始设定温度， $p_0$  为电价初始值， $p$  为当前时刻电价，系数  $k$  为空调对电价的响应指数，不同的用户响应指数不同。

$T_{sp}$  改变后，通过空调开关断时间求解公式得到最新的空调开关断时间，如图 3.15 所示。

制冷的空调，当设定温度提高时，关闭时间变长，开启时间变短，相应的消耗功率减少。

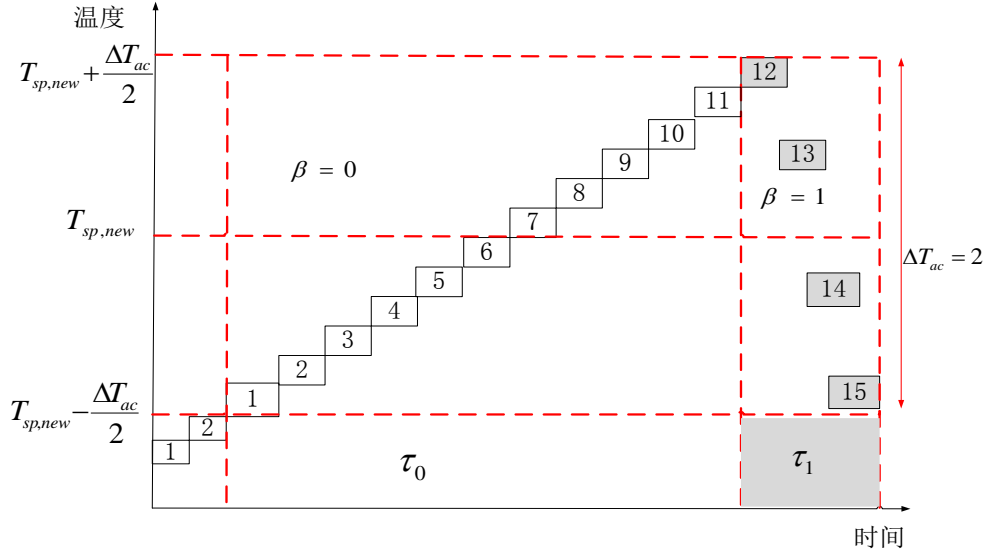


图 3.15 空调状态序列随设定温度的变化图

### (3) 电动汽车建模

#### (a) 基本电动汽车 (EV) 负荷的建模

电动汽车充电模型，三个参数至关重要：额定充电功率，充电插入时间和电池的充电状态 (SOC)。方程 (3.37) 是 EV 的充电功率计算：

$$p_{EV, i} = P_{EV} \cdot S_{EV, i} \cdot w_{EV, i} \cdot c_{EV, i} \quad (3.37)$$

其中，

$p_{EV, i}$ ：EV 在时隙  $i$  的充电功率 (KW)；

$P_{EV}$ ：EV 额定功率 (KW)；

$S_{EV, i}$ ：EV 在时隙  $i$  的连接状态，0 代表 EV 没有与充电接口有物理连接，1 代表已经连接；

$w_{EV, i}$ ：在时隙  $i$  EV 不受控制的充电状态，这取决于电池的 SOC，如式 (3.38)：0 代表 EV 没有充电，1 代表 EV 正在充电；

$c_{EV, i}$ ：EV 在时隙  $i$  的 DR 控制信号，0=OFF, 1=ON。

$$w_{EV, i} = \begin{cases} 0, & SOC_i \geq SOC_{\max} \\ 1, & SOC_i < SOC_{\max} \end{cases} \quad (3.38)$$

在时间时隙 $i$ 时电池  $SOC_i$  为  $SOC$  的在先前时间时隙的函数, 用于驱动和电池的额定容量的能量, 由下面式子确定:

$$SOC_0 = 1 - \frac{E_{dr}}{C_{batt}} \quad (3.39)$$

$$SOC_i = SOC_i + P_{EV} \cdot \frac{\Delta t}{C_{batt}} \quad (3.40)$$

其中,  $SOC_0$  当 EV 插入充电时的电池  $SOC$ ;  $\Delta t$  时隙 $i$ 时长 (分);  $E_{dr}$  已经消耗的能量 (KWh);  $C_{batt}$  电池容量 (KWh)。

电动汽车的电力需求取决于如家庭控制器, 或 DR 控制信号  $c_{EV, i}$ 。0 代表 DR 控制信号将停止 EV 的充电, 而 1 代表 DR 控制信号将允许电动车开始充电。所建立的 EV 模式将延长电动汽车充电完成时间, 如果电动汽车充电是由 DR 信号中断。也就是说, 在 EV 将继续充电直到完全充满。

#### (b)电动汽车充放电时间分布和响应模型

##### 1) 充电接口物理连接

电动汽车是人们作为代步的交通工具, 其主要功能是满足人们的出行活动。当需要出行时, 其充电接口显然是不会有物理连接, 因此其充电接口物理连接的时间分布一定是和出行规律相对应的。一般的, 我们可以认为出行时间段是没有物理连接的, 而非出行时间段是有物理连接的。

如果能确定电动汽车的出行时间规律, 便能得出电动汽车充电接口物理连接的时间分布。我们提出如下方法:

电动汽车分类。不同类型 (按出行用途分类, 如上下班代步车、电动公交车、) 电动汽车具有不同的出行规律, 比如上下班代步车在早上 7-9 点和下午 5-7 点运行, 而在其他时间可认为是停在家或停车场的; 而电动公交车运行时间段是由公交线路时间安排决定, 公交车的运行与在场时间是相对固定的。根据这些规律, 可以确定电动汽车充电接口物理连接的时间段。

##### 2) 充电状态限制的电价弹性

一般的, 电动汽车的  $SOC_{max}$  是用户根据其用车需求和习惯来确定的。但是电动汽车作为电能消费设备, 也具有一般电力负荷的电价弹性属性 ( $SOC_{max} = f(p)$ ), 可表现为电动汽车  $SOC_{max}$  和电价变化的关系。在同一  $SOC$  水平下, 电价越高用户对电动汽车进行充电的意愿



越低。

### 3) 基于电价 DR 控制信号

充电模型中,  $c_{EV,i}$  是 EV 在时隙  $i$  的 DR 控制信号。电动汽车响应时,  $c_{EV,i} = 1$ ; 反之,  $c_{EV,i} = 0$ 。

#### (c) 电动汽车负荷群的聚合

采集电动汽车群中各电动汽车的三个参数 (电动汽车额定充电电力, 插入时间和电池 SOC)。车插入时间是来源于全国家庭旅行调查, 得到典型数据。同类 EV 插入时间是用正态分布具有均值和实际调查数据的方差。

## 3.3.3 NETLOGO 程序实现

### (1) Prosumer 建模在 NETLOGO 中的程序实现

基于上文中对 Prosumer 的建模过程, 我们从负荷的智能属性、物理属性和通信属性等三方面来分析 Prosumer 建模在 NETLOGO 中的程序实现。

(a) Prosumer 的智能属性为 (即负荷-电价响应特性):

1) 刚性负荷: 负荷量  $q_k$  不随电价改变;

2) 柔性负荷:  $d_k = D_k \cdot [1 - \rho_k \cdot \mu_k \cdot (1 - \varphi_k)]$

3) 分布式电源:  $g_k = G_k \cdot [1 - \nu_k \cdot \mu_k \cdot (1 - \varphi_k)]$

4) 储能元件: 充电时为  $d_k = D_k \cdot [1 - \rho_k \cdot \mu_k \cdot (1 - \varphi_k)]$

放电时为  $g_k = G_k \cdot [1 - \nu_k \cdot \mu_k \cdot (1 - \varphi_k)]$

在 NETLOGO 中的代码描述如下所示:

1) 刚性负荷: ask buyer i [set p55 3]; 即让 prosumer i 设置其功率值为 3KW。

2) 柔性负荷: ask buyer i [set p55 (6 \* (1 - r5 \* (xcor / 80) \* (1 - (ycor / 80))))]; 即让 Prosumer i 按照柔性负荷电价-负荷响应的公式计算其功率值, 其中  $D_k$  为 6KW;  $\nu_k = r5$ ;  $\mu_k$ ,  $\varphi_k$  由该 Prosumer i 在 NETLOGO 的 3D 界面中所处的位置决定即  $\mu_k = xcor / 80$ ,  $\varphi_k = ycor / 80$ 。

3) 分布式电源: ask buyer i [set p55 (3 \* (1 + r5 \* (xcor / 80) \* (1 - (ycor / 80))))]; 即让 Prosumer i 按照分布式电源电价-负荷响应的公式计算其功率值, 其中  $G_k$  为 3KW;  $\varphi_k = -r5$ ;

$\mu_k$ ,  $\varphi_k$  由该 Prosumer i 在 NETLOGO 的 3D 界面中所处的位置决定即  $\mu_k = xcor / 80$ ,  $\varphi_k = ycor / 80$ 。

4)储能元件: 充电时: ask buyer i [set p55 (6 \* (1 - r5 \* (xcor / 80) \* (1 - (ycor / 80))))]; 即让 Prosumer i 按照柔性负荷电价-负荷响应的公式计算其功率值, 其中  $D_k$  为 6KW;  $v_k = r5$ ;  $\mu_k$ ,  $\varphi_k$  由该 Prosumer i 在 NETLOGO 的 3D 界面中所处的位置决定即  $\mu_k = xcor / 80$ ,  $\varphi_k = ycor / 80$ 。

放电时: ask buyer i [set p55 (6 \* (1 + r5 \* (xcor / 80) \* (1 - (ycor / 80))))]; 即让 Prosumer i 按照柔性负荷电价-负荷响应的公式计算其功率值, 其中  $G_k$  为 6KW;  $\varphi_k = -r5$ ;  $\mu_k$ ,  $\varphi_k$  由该 Prosumer i 在 NETLOGO 的 3D 界面中所处的位置决定即  $\mu_k = xcor / 80$ ,  $\varphi_k = ycor / 80$ 。

(b) Prosumer 物理属性: 输入为节点电价, 输出为功率需求量;

在 NETLOGO 中代码描述如下:

buyer1s-own [p55 r5...]; 定义负荷代理 1 所管辖的所有负荷自身变量。

buyer2s-own [p77 r7...]; 定义负荷代理 2 所管辖的所有负荷自身变量。

buyer3s-own [p99 r9...]; 定义负荷代理 3 所管辖的所有负荷自身变量。

(c) Prosumer 通信属性: 通过 NETLOGO 与 MATLAB 之间的接口进行电价、系统总需求功率等信息的交互。

在 NETLOGO 中代码描述如下:

while [i < 30]

[set 5p (p5 + [p55] of buyer1 i)

set i (i + 1)]

while [i < 60]

[set p7 (p7 + [p77] of buyer1 i)

set i (i + 1)]

while [i < 90]

[set p9 (p9 + [p99] of buyer1 i)

set i (i + 1) ]

MATLAB:send-double"p5" p5

MATLAB:send-double"p7" p7

; 负荷将功率信息上传给负荷代理

```

MATLAB:send-double "p9" p9 ; 即负荷代理发送功率值给 MATLAB
MATLAB:eval "b.bus(5,3)=p5;b.bus(7,3)=p7;b.bus(9,3)=p9"
MATLAB:eval "d=loadcase(runopf(b))" ; MATLAB 进行 OPF 运算
MATLAB:eval "c5=d.bus(5,14);p5=d.bus(5,3)"
ask microplayer 24 [ set c MATLAB:get-double "c5"
                    set p MATLAB:get-double "p5"]
MATLAB:eval "c7=d.bus(7,14);p7=d.bus(7,3)"
ask microplayer 25 [ set c MATLAB:get-double "c7"
                    set p MATLAB:get-double "p7" ]
MATLAB:eval "c9=d.bus(9,14);p9=d.bus(9,3)"
ask microplayer 26 [ set c MATLAB:get-double "c9"
                    set p MATLAB:get-double "p9" ] ; NETLOGO 调用 MATLAB 中电价

```

和发电机出力信息。

## (2) 空调建模在 NETLOGO 中的程序实现

基于上文中对空调的建模过程，我们从负荷的智能属性、物理属性和通信属性等三方面来分析空调建模在 NETLOGO 中的程序实现。

(a)空调负荷的智能属性由两部分组成：空调自身温度特性和负荷-电价响应特性。空调自身温度特性为：设定房间的温度值和温度变化的限制范围。当室内温度超过最大允许温度  $T_{sp} + \frac{\Delta T_{ac}}{2}$  时，制冷空调开启。当室内温度低于  $T_{sp} - \frac{\Delta T_{ac}}{2}$  时，该房间的空调关闭。当室内温度在预设的温度范围内  $\left[ T_{sp} - \frac{\Delta T_{ac}}{2}, T_{sp} + \frac{\Delta T_{ac}}{2} \right]$  时，制冷空调将保持其之前的运行状态。如下式所示：

$$\beta_{ac,t} = \begin{cases} 1 & T_{ac,t} > T_{sp} + \frac{\Delta T_{ac}}{2} \\ 0 & T_{ac,t} < T_{sp} - \frac{\Delta T_{ac}}{2} \\ \beta_{ac,t-1} & T_{sp} - \frac{\Delta T_{ac}}{2} \leq T_{ac,t} \leq T_{sp} + \frac{\Delta T_{ac}}{2} \end{cases} \quad (3.42)$$

在 NETLOGO 中的代码描述如下：

```

ask aircondition i [ set Tset0 MATLAB:get-double "c"
                    set Iac (Tout - Tset0 - 1) / 0.4
                    set Tmax 27
                    set Tmin 23] ; 设定空调初始设置温度由 MATLAB 随机产生
ask aircondition i [ set Tset Tset0

```

```

ifelse x < [Iac] of aircondition i
[set Tac Tout - 0.4 set color 15 set q 1.5]
[
  if Tac <= (Tset - 1)
    [set Tac Tac + 0.2 set color 9 set q 0]
  if Tac >= (Tset + 1)
    [set Tac Tac - 0.4 set color 15 set q 1.5]
  if Tac > (Tset - 1) and Tac < (Tset + 1)
    [
      ifelse [q] of aircondition i = 0
        [set Tac Tac + 0.2 set color 9 set q 0]
        [set Tac Tac - 0.4 set color 15 set q 1.5]
    ]
]

```

；根据空调运行规则，确定空调开关状态。

负荷-电价响应特性为：设定电价响应阈值，当电价超过该值时，随着电价的增长，制冷空调的设置温度按照规则设定调高。当空调的设置温度响应做出改变后，空调继续按照以上自身温度特性运行。

(b) 空调物理属性：输入为节点电价、室外温度，输出为功率需求量；

在 NETLOGO 中代码描述如下：

airconditions-own [Tset Tset0 Tac Tmax Tmin Iac q]；定义空调负荷自身变量。

(c) 空调通信属性：通过 NETLOGO 与 MATLAB 之间的接口进行电价、系统总需求功率等信息的交互。

在 NETLOGO 中的代码描述如下：

```

if x = (y * 30) [price
  set y y + 1] ; MATLAB 每个 30 分钟发布一下电价
ifelse c5 < 16
[
  while [i < 530]
  [ ask aircondition i [
    set Tset Tset0

```

```

        ifelse x < [Iac] of aircondition i
            [set Tac Tout - 0.4 set color 15 set q 1.5]
            [
                if Tac <= (Tset - 1)
                    [set Tac Tac + 0.2 set color 9 set q 0]
                if Tac >= (Tset + 1)
                    [set Tac Tac - 0.4 set color 15 set q 1.5]
                if Tac > (Tset - 1) and Tac < (Tset + 1)
                    [
                        ifelse [q] of aircondition i = 0
                            [set Tac Tac + 0.2 set color 9 set q 0]
                            [set Tac Tac - 0.4 set color 15 set q 1.5]
                    ]
            ]
        set i i + 1
    ]
]
; 此时电价阈值为 16，根据当前时刻电价决定空调运行状态
MATLAB:send-double"p5" p5
MATLAB:send-double"p7" p7
MATLAB:send-double"p9" p9

```

## 第四章 电网多智能体协调控制与仿真

本文仿真环境由 NETLOGO 与 MATLAB 构成如图 4.1 所示，其中 MATLAB 主要负责电价计算与预测，NETLOGO 承担电力系统智能元件建模以及负荷分布式控制的工作，通过 NETLOGO 和 MATLAB 之间的接口模块实现整个系统网络数据交互。以基于响应电价的负荷仿真为例，MATLAB 根据电力供需状况每五分钟实时制定并发布电价，NETLOGO 中的负荷通过 NETLOGO 和 MATLAB 之间的接口接收来自 MATLAB 的电价信号，根据电价大小来改变功率需求，同时采用强化学习算法优化每个负荷的经济性响应率和舒适度响应率，得到最新负荷功率需求，NETLOGO 将最新的负荷需求上传给 MATLAB，至此完成一个循环过程。

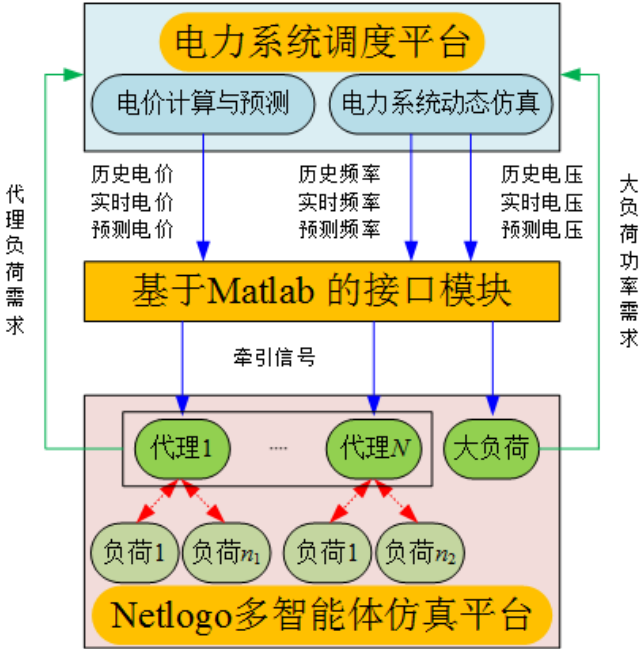


图 4.1 基于 NETLOGO 与 MATLAB 的电网多智能体仿真系统

NETLOGO 与 MATLAB 在仿真过程中的时间配合方式如图 4.2 所示。MATLAB 作为电力系统调度平台每隔 5 分钟计算一次潮流，得到实时电价，同时预测短时电价、计算电网频率和节点电压，并向各负荷代理、大负荷下发该时段电价、频率、电压，需要时同时下发该时段前后的历史和预测电价、频率、电压。这些电价、频率、电压统称为牵引信号（图中蓝色实线），“指导牵引”负荷调整自身用电需求，在最大化自身利益的同时服务于电网，多次连续调整工作状态和功率需求（图中红色虚线）。在每 5 分钟时间段结束之前，负荷代理和独立负荷向电力系统调度平台上报本时间段的功率总量需求，以便计算/预测下一时间段的牵引信号。

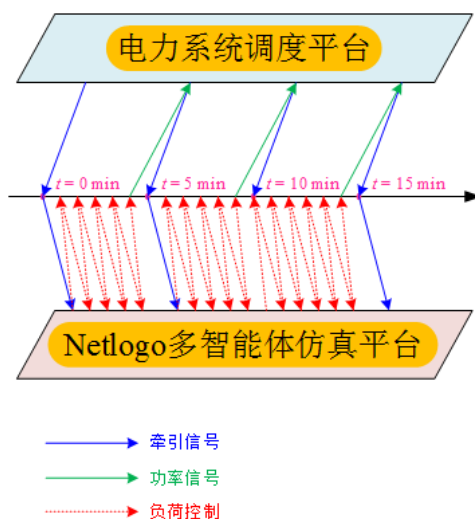


图 4.2 各平台之间的运行时间配合方式

## 4.1 基于 Prosumer 模型的优化控制与仿真

### 4.1.1 基于 Prosumer 模型的优化控制策略

根据第三章中提出的 Prosumer 建模方案, 基于 NETLOGO 的电网多智能体控制具体实施步骤如下<sup>[29]-[39]</sup>:

- 1) 根据电力系统网络结构, 建立基于 MATLAB 与 NETLOGO 的联合仿真平台, 其中, 在 MATLAB 中建立电力系统元件模型, 在 NETLOGO 中定义代表电力系统元件的智能体通用模块, 同时, 搭建 MATLAB 和 NETLOGO 之间的数据交换接口模块实现信息交互;
- 2) 针对各种电力元件类型, 分别根据电力元件基准量、电价, 以及对应电力元件的各目标的目标倾向度, 建立分别对应各种电力元件类型的电力元件-电价响应特性模型;
- 3) 根据对应的电力元件-电价响应特性模型, 分别获得各个电力元件的各个目标的目标函数, 并且分别针对各个电力元件, 将电力元件的各个目标的目标函数进行加权处理, 分别获得对应各个电力元件的总目标函数;
- 4) 将各个电力元件随机分布在 NETLOGO 三维层面上, 构成多个电力元件节点, 并获得各个电力元件的各个目标的初始目标倾向度, 即为各个电力元件的初始策略; 针对 NETLOGO 三维层面中的网络节点, 随机设定电价, 并且根据 NETLOGO 三维层面中的电力元件节点, 建立电力元件代理, 电力元件代理的数量与电力元件节点的数量一致, 电力元件代理与电力元件节点一一对应, 各个电力元件代理管辖对应各个电力元件, 并且各个电力元件代理分别用于其管辖的各个电力元件和 MATLAB 之间的信息传输;

5) 以各个电力元件的初始策略作为负荷基准量, 分别针对各个电力元件的各个目标的目标倾向度, 采用 $+i$  或 $-i$  的方式分别获得各个电力元件对应的策略, 并结合各个电力元件的初始策略构成各个负荷的策略集, 其中,  $i$  为每一步迭代步长。

其中,  $i = 1$ , 在 NETLOGO 三维层面上, 每一个负荷周围包括八个点, 该八个点分别是  $[\mu_k + 1, \varphi_k]$ ,  $[\mu_k - 1, \varphi_k]$ ,  $[\mu_k, \varphi_k + 1]$ ,  $[\mu_k, \varphi_k - 1]$ ,  $[\mu_k + 1, \varphi_k + 1]$ ,  $[\mu_k - 1, \varphi_k - 1]$ ,  $[\mu_k - 1, \varphi_k + 1]$ ,  $[\mu_k + 1, \varphi_k - 1]$ , 即每一个负荷对应八个不同的策略, 分别构成各个电力元件的策略集;

6) 根据对应各个电力元件的总目标函数, 分别获得各个电力元件对应其策略集中各个策略的总目标函数值, 并分别选择获得各个负荷对应其最大总目标函数值的策略, 作为各个电力元件的优选策略;

7) 分别根据各个电力元件的优选策略中的各个目标的目标倾向度, 将各个电力元件分别运动到 NETLOGO 三维层面中相应的位置上, 并更新各个电力元件的各个目标的目标倾向度; 然后根据对应的电力元件-电价响应特性模型, 获得此时各个电力元件的功率, 并且结合电力元件代理针对对应电力元件的管辖, 分别获得各个电力元件代理的总功率;

8) 将各个电力元件代理的总功率通过 MATLAB 与 NETLOGO 之间的数据交换接口模块, 由 NETLOGO 发送至 MATLAB 中, 在 MATLAB 中分别针对各个电力元件代理的总功率进行最优潮流计算, 获得发电机出力和对应各个网络节点的电价, 并将该各个网络节点的电价, 通过 MATLAB 与 NETLOGO 之间的数据交换接口模块返回至 NETLOGO 中, 更新 NETLOGO 三维层面中对应网络节点上的电价;

9) 将 NETLOGO 三维层面中各个网络节点上的电价作为牵引信号, 并分别由各个负电力元件代理将对应网络节点上的电价发布给其管辖的各个电力元件;

10) 根据此时 NETLOGO 三维层面中, 各个电力元件的位置, 以及各个电力元件的各个目标的目标倾向度, 更新各个电力元件的初始策略, 并按 5) 中的方法, 更新各个电力元件对应的策略集, 然后根据对应各个电力元件的总目标函数, 结合各个电力元件对应的电价, 分别获得各个电力元件对应其策略集中各个策略的总目标函数值, 并分别针对各个电力元件, 判断电力元件的初始策略对应的总目标函数值是否大于其策略集中其它策略所对应的总目标函数值, 是则该电力元件停止运动; 否则返回 4)。

#### 4.1.2 基于 Prosumer 模型的网荷互动仿真

我们开发了一个基于多智能体的仿真平台, 如图 4.3 所示。整个仿真平台由三层构成:



电网拓扑结构层、负荷代理层、负荷层。按照电网拓扑结构所示，在网络母线上设置相应量的负荷代理，而一个负荷代理则连接一定量的小负荷，即每个负荷管理一定量的负荷，将调度中心发布的电价下达给自身所管辖的负荷，同时将管辖的负荷群的总需求量上传给调度中心。调度中心的工作由 MATLAB 完成，每五分钟接受负荷代理上传的负荷需求，进行 OPF 运算得到该时段市场出清电价，下发给各个负荷代理进行下一轮仿真<sup>[40]~[42]</sup>。

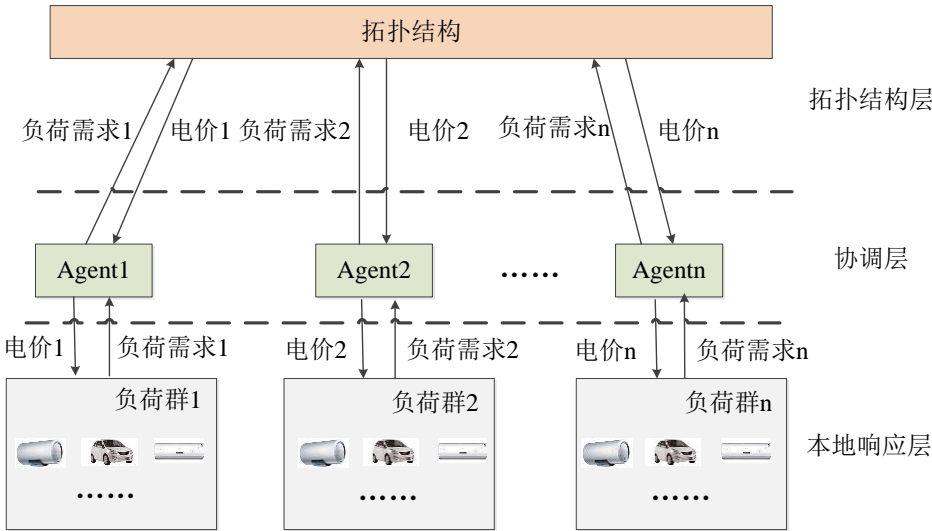


图 4.3 基于市场和负荷心理特性的仿真系统

具体仿真内容如下：以简单的三机九节点网络为例，采用本文提出的优化控制方法解决电力市场仿真中负荷多智能体协调控制的问题，该系统的母线参数如表 4.1 所示。

表 4.1 三机九节点母线参数

Bus	Type	Pd	Qd	Gs	Bs	Vm	Va	BasekV	Vmax	Vmin
1	3	0	0	0	0	1	0	345	1.1	0.9
2	2	0	0	0	0	1	0	345	1.1	0.9
3	2	0	0	0	0	1	0	345	1.1	0.9
4	1	0	0	0	0	1	0	345	1.1	0.9
5	1	80	30	0	0	1	0	345	1.1	0.9
6	1	0	0	0	0	1	0	345	1.1	0.9
7	1	40	35	0	0	1	0	345	1.1	0.9
8	1	0	0	0	0	1	0	345	1.1	0.9
9	1	120	50	0	0	1	0	345	1.1	0.9

NETLOGO 作为该软件系统的前台，也是交互界面，在 NETLOGO 中搭建该系统框架，如图 4.4 所示：底层搭建三机九节点拓扑结构，中层搭建三个负荷代理、三个运营商代理，上层搭建负荷，每个负荷代理分别管辖 30 个小负荷，整个网络共计 90 个负荷。每个负荷以经济利益和舒适度为综合目标，在满足综合目标的条件下对电价做出响应改变对经济性和舒适度的响应率。仿真过程中得到的所有负荷综合目标平均值、三台发电机发电量以及三条负荷母线的负荷需求变化情况。

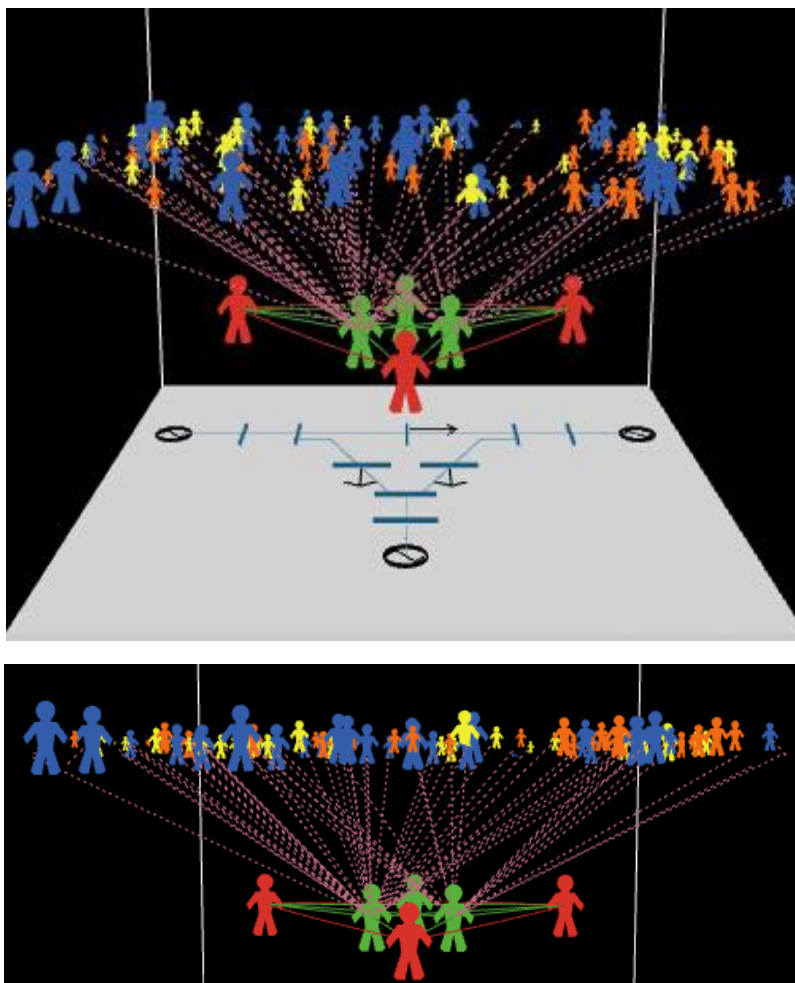


图 4.4 基于 Prosumer 模型的网荷互动仿真平台

首先对负荷代理和负荷进行建模，让 90 个负荷随机分布在 NETLOGO 二维平面上，每个负荷相应的随机产生一个坐标  $(x_k, y_k)$ 。设定每个负荷的  $x_k$  值对应为经济性倾向度  $\mu_k$ ， $y_k$  值对应为舒适度倾向度  $\varphi_k$ 。根据表 4.1 中的电网参数对四类负荷初始化建模如下：

(1) 负荷代理 1：

刚性负荷： $q_k=2\text{MW}$ ；

柔性负荷:  $d_k = 4 \cdot [1 - \rho_k \cdot x_k \cdot (1 - y_k)]$

分布式电源:  $g_k = 2 \cdot [1 - \nu_k \cdot x_k \cdot (1 - y_k)]$

储能元件: 充电时为  $d_k = 4 \cdot [1 - \rho_k \cdot x_k \cdot (1 - y_k)]$

放电时为  $g_k = 2 \cdot [1 - \nu_k \cdot x_k \cdot (1 - y_k)]$

(2) 负荷代理 2:

刚性负荷:  $q_k = 1\text{MW}$

柔性负荷:  $d_k = 2 \cdot [1 - \rho_k \cdot x_k \cdot (1 - y_k)]$

分布式电源:  $g_k = 1 \cdot [1 - \nu_k \cdot x_k \cdot (1 - y_k)]$

储能元件: 充电时为  $d_k = 2 \cdot [1 - \rho_k \cdot x_k \cdot (1 - y_k)]$

放电时为  $g_k = 1 \cdot [1 - \nu_k \cdot x_k \cdot (1 - y_k)]$

(3) 负荷代理 3:

刚性负荷:  $q_k = 3\text{MW}$

柔性负荷:  $d_k = 6 \cdot [1 - \rho_k \cdot x_k \cdot (1 - y_k)]$

分布式电源:  $g_k = 3 \cdot [1 - \nu_k \cdot x_k \cdot (1 - y_k)]$

储能元件: 充电时为  $d_k = 6 \cdot [1 - \rho_k \cdot x_k \cdot (1 - y_k)]$

放电时为  $g_k = 3 \cdot [1 - \nu_k \cdot x_k \cdot (1 - y_k)]$

初始化三个节点电价标准值为  $\rho_1 = 0.2$ ,  $\rho_2 = 0.3$ ,  $\rho_3 = 0.4$ , 为防止用户套利, 我们假定  $\nu_k = -\rho_k$ , 即  $\nu_1 = -0.2$ ,  $\nu_2 = -0.3$ ,  $\nu_3 = -0.4$ 。在此初始电价下, 计算负荷总目标函数值。

(1) 负荷代理 1:

刚性负荷:  $R_k = -0.4 \cdot x_k + y_k^2 \cdot (1 - x_k)$

柔性负荷:  $R_k = -0.8 \cdot x_k \cdot [1 - 0.2 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$

分布式电源:  $R_k = -0.4 \cdot x_k \cdot [1 + 0.2 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$

储能元件: 充电时为  $R_k = -0.8 \cdot x_k \cdot [1 - 0.2 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$ ;

放电时为  $R_k = -0.4 \cdot x_k \cdot [1 + 0.2 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$ 。

## (2) 负荷代理 2:

刚性负荷:  $R_k = -0.3 \cdot x_k + y_k^2 \cdot (1 - x_k)$

柔性负荷:  $R_k = -0.6 \cdot x_k \cdot [1 - 0.3 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$

分布式电源:  $R_k = -0.3 \cdot x_k \cdot [1 + 0.3 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$

储能元件: 充电时为  $R_k = -0.6 \cdot x_k \cdot [1 - 0.3 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$

放电时为  $R_k = -0.3 \cdot x_k \cdot [1 + 0.3 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$

## (3) 负荷代理 3:

刚性负荷:  $R_k = -1.2 \cdot x_k + y_k^2 \cdot (1 - x_k)$

柔性负荷:  $R_k = -2.4 \cdot x_k \cdot [1 - 0.4 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$

分布式电源:  $R_k = -1.2 \cdot x_k \cdot [1 + 0.4 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$

储能元件: 充电时为  $R_k = -2.4 \cdot x_k \cdot [1 - 0.4 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$

放电时为  $R_k = -1.2 \cdot x_k \cdot [1 + 0.4 \cdot x_k \cdot (1 - y_k)] + y_k^2 \cdot (1 - x_k)$

让每一个负荷计算周围的八个点的  $R_k$  值, 该八个点分别是  $[x_k + 1, y_k]$ ,  $[x_k - 1, y_k]$ ,  $[x_k, y_k + 1]$ ,  $[x_k, y_k - 1]$ ,  $[x_k + 1, y_k + 1]$ ,  $[x_k - 1, y_k - 1]$ ,  $[x_k - 1, y_k + 1]$ ,  $[x_k + 1, y_k - 1]$ , 即对应八个不同的策略。由负荷判断这八个策略对应的  $R_k$  值哪一个最大, 选取对应的  $R_k$  值最大的策略, 负荷则运动到该点对应的位置上。

在新策略下求取三个负荷代理的总功率  $P_{L1}$ 、 $P_{L2}$ 、 $P_{L3}$ , 将负荷量输送到 MATLAB。在 MATLAB 中进行 OPF 运算, 得到三个发电机的出力分别为  $P_{G1}$ 、 $P_{G2}$ 、 $P_{G3}$  以及节点电价  $C_1$ 、 $C_2$ 、 $C_3$ 。将发电机出力及电价输送到 NETLOGO 中。

负荷代理将得到的节点电价  $C_1$ 、 $C_2$ 、 $C_3$  换算成对应的  $\rho_k$ 、 $v_k$  发布给每个负荷, 负荷计算在该电价下自身总目标值, 同时计算周围八个点的总目标函数值。若此时的总目标值均大于周围八个点的值, 则该负荷停止运动。否则利用 NETLOGO 中可使动作循环执行的功能, 让负荷 Agents 不断执行步骤三、四, 直至找到最大的  $R_k$  值所对应的策略, 负荷停止运动, 优化结束。

仿真结果如图 4.5 所示，在 NETLOGO 三维的仿真结果中，以负荷所在平面的 x 轴表示经济性倾向度，y 轴表示舒适度倾向度，取值范围为 0-1，每个小负荷由于其自身的目标不同，在接受的电价信号时为满足自身目标最大化会改变自己的用电模式，即改变经济性和舒适度倾向度，从而改变用电量。而在仿真界面中的表现即为小负荷在其平面中做横向和竖向运动，直至找到满足自身目标最大的经济性和舒适度倾向度。从仿真曲线图 4.6 中可以观察到负荷因电价改变以及对经济性、舒适度倾向性的改变导致功率变化情况。因 MATLAB 计算得到的电价在后期基本变化不大，造成负荷及发电机出力变化情况不明显。

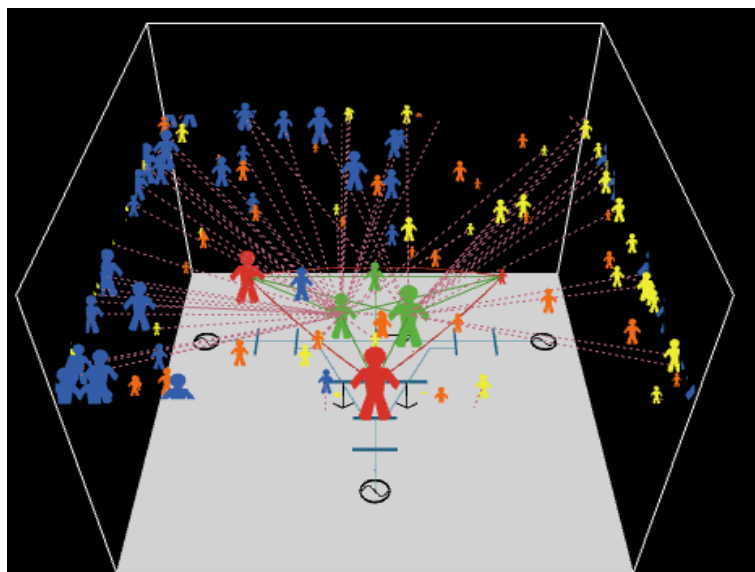


图 4.5 NETLOGO 三维的仿真界面图

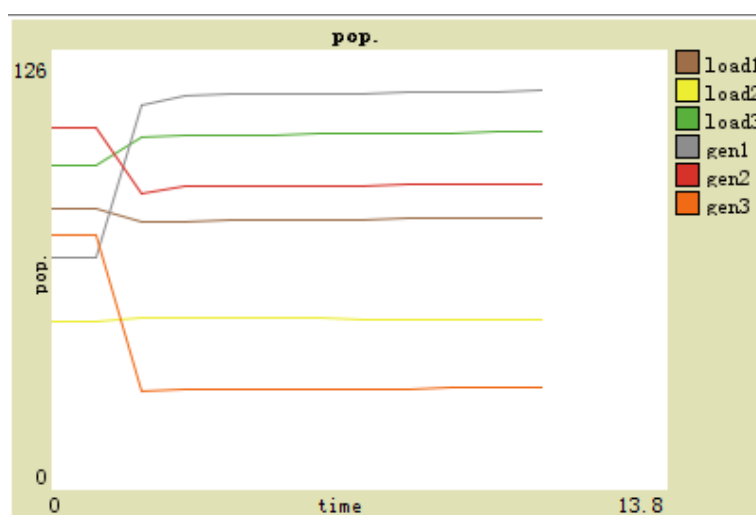


图 4.6 NETLOGO 三维的仿真曲线图

仿真过程中得到的潮流计算的微增率和发电机出力变化情况如图 4.7 所示。从图中可以看出计算一次潮流得到的微增率略有不同。本文 MATLAB 计算潮流时将网损也考虑在内，故

最终三个节点的微增率略有不同。而发电机出力也随着负荷需求量的改变而变化，以保证系统稳定运行。NETLOGO 进行下一次仿真过程需调用系统电价，本文中该系统电价即采用潮流计算得到的节点微增率。

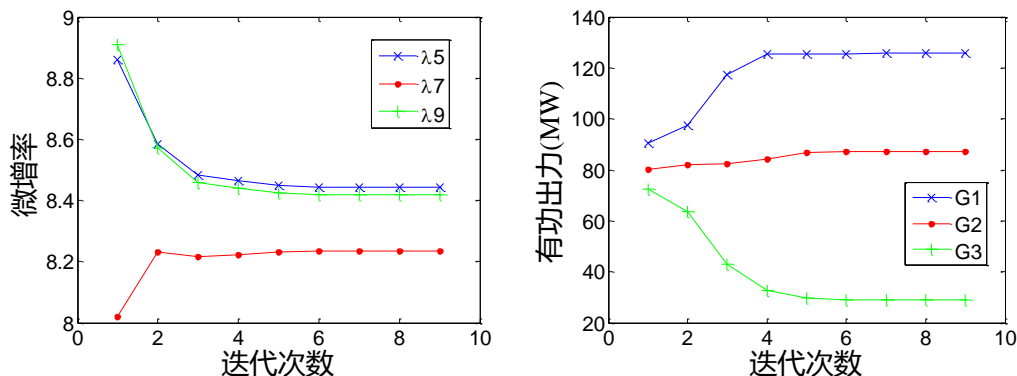


图 4.7 负荷节点微增率及发电机出力

## 4.2 基于空调模型的优化控制与仿真

### 4.2.1 基于空调模型的优化控制策略

根据第三章中提出的空调建模方案，基于 NETLOGO 与 MATLAB 的电网多智能体控制的具体实施步骤如下<sup>[43]-[47]</sup>：

1) 根据电力系统网络结构，建立基于 MATLAB 与 NETLOGO 的联合仿真平台，其中，在 MATLAB 中建立电力系统元件模型，在 NETLOGO 中定义代表电力系统元件的智能体通用模块，同时，搭建 MATLAB 和 NETLOGO 之间的数据交换接口模块实现信息交互；

2) 针对空调负荷特性，根据空调温度设定值、室外温度、空调等效热阻、空调等效热容、空调的制冷/制热功率，建立分别对应各种负荷空调的空调特性模型；同时根据空调温度设定值与电价之间的关系，搭建空调响应电价模型；

3) 将各个空调负荷随机分布在 NETLOGO 三维层面上，针对 NETLOGO 三维层面中的网络节点，随机设定初始电价，并且根据 NETLOGO 三维层面中的负荷节点，建立负荷代理，负荷代理的数量与负荷节点的数量一致，负荷代理与负荷节点一一对应，各个负荷代理管辖对应各个空调聚合体（每个空调聚合体由相应数量的单个空调负荷加和而得），并且各个负荷代理分别用于其管辖的各个空调聚合体和 MATLAB 之间的信息传输；

4) 根据各类空调负荷的分布式特性，设定不同类型的空调开启时间段。各个负荷在其开启时段按照自身空调运行特性工作。每个空调聚合体对所聚合的单个空调的功率进行求和得

到聚合体的总功率，同时每个负荷代理对所管辖的空调聚合体的功率进行求和得到该节点的负荷需求量；

5) NETLOGO 中的各个负荷代理每 5 分钟将各自节点的负荷需求量过 MATLAB 与 NETLOGO 之间的数据交换接口模块，由 NETLOGO 发送至 MATLAB 中，在 MATLAB 中分别针对各个负荷代理的总功率进行最优潮流计算，获得发电机出力和对应各个网络节点的电价，并将该各个网络节点的电价，通过 MATLAB 与 NETLOGO 之间的数据交换接口模块返回至 NETLOGO 中，更新 NETLOGO 三维层面中对应网络节点上的电价；

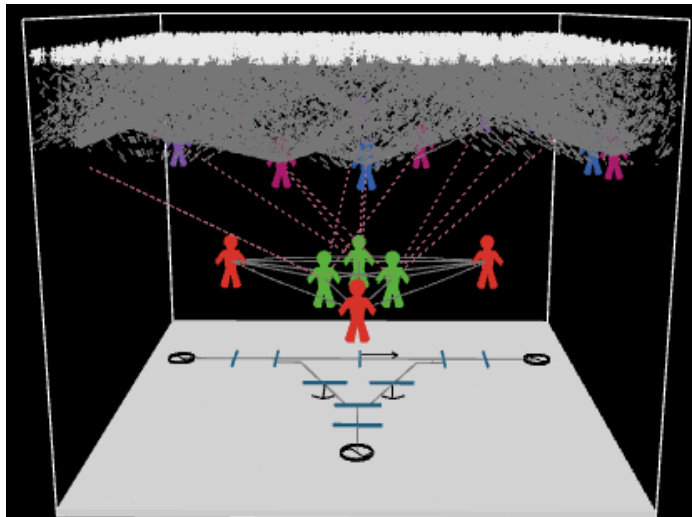
6) 将 NETLOGO 三维层面中各个网络节点上的电价作为牵引信号，并分别由各个负荷代理将对应网络节点上的电价发布给其管辖的各个空调聚合体；

7) 单个空调接收到来自空调聚合体下达的最新电价，根据自身空调温度设定值与电价之间的响应模型，修改空调温度设定值。单个空调依据最新的空调温度设定值修改自身空调运行特性，同时每个空调按照此时的空调运行特性工作；

8) MATLAB 每五分钟下达一次电价，NETLOGO 中的空调负荷根据最新电价修改空调温度设定值及其运行特性，重复步骤 5) -7)。仿真一天 24 小时内空调群与电网互动。

#### 4.2.2 基于空调模型的网荷互动仿真

该仿真算例搭建的平台与基于 Prosumer 模型的网荷互动仿真案例中的平台基本相同，在其三层结构的基础上，在最上层增加一次单个空调的响应层，如图 4.8 所示。四层结构分别为：网络的拓扑结构层；协调层；空调聚合层；单个空调的响应层。仿真时我们假设市场电价由 MATLAB 进行最优潮流计算而得，不同类型的空调根据自身运行特性工作，不同用户对空调的使用需求不同会产生空调使用行为的不同，从而来观察源网荷互动的情况<sup>[47]-[52]</sup>。



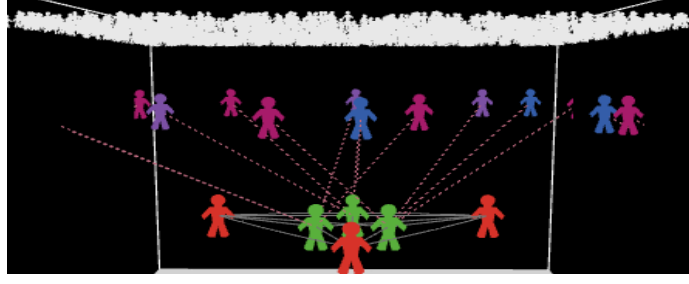


图 4.8 基于空调模型的网荷互动仿真平台

仿真时在 NETLOGOD 界面的中层建立 3 个负荷代理（绿色）以及三台发电机（红色），如图 4.8 所示。每个负荷代理分别管辖 3 个空调聚合体，不同空调聚合体由不同数量的空调聚合而成。整个网络共计有 1500 台空调，同一负荷代理上的负荷电价相同。我们对空调进行建模，空调属性如下：

智能属性：

$$\beta_{ac,t} = \begin{cases} 1 & T_{ac,t} > T_{sp} + \frac{\Delta T_{ac}}{2} \\ 0 & T_{ac,t} < T_{sp} - \frac{\Delta T_{ac}}{2} \\ \beta_{ac,t-1} & T_{sp} - \frac{\Delta T_{ac}}{2} \leq T_{ac,t} \leq T_{sp} + \frac{\Delta T_{ac}}{2} \end{cases} \quad (5.1)$$

空调运行特性：

$$T_i^{t+1} = T_i^t + \frac{\Delta t}{\tau_0} \cdot \Delta T_{ac}, \quad \beta = 0 \quad (5.2)$$

$$T_i^{t+1} = T_i^t - \frac{\Delta t}{\tau_1} \cdot \Delta T_{ac}, \quad \beta = 1 \quad (5.3)$$

空调温度设定值响应电价模型：

$$T_{sp} = T_0 + \frac{P_k - P_0}{k} \quad (5.4)$$

空调聚合模型：

$$P_{sum} = \frac{n_1}{n_c} \cdot N \cdot P \quad (5.5)$$

物理属性：输入为节点电价  $\rho_k$ ，输出为功率需求量  $P_{sum}$ ；

通信属性：通过 NETLOGO 与 MATLAB 之间的接口进行电价、系统总需求功率等信息的交互。

在 NETLOGO 仿真平台中设定空调负荷的基本参数信息，见表 4.2。将空调负荷分为三类：商业空调、居民空调、工业空调。三类空调在 24 小时内的开启时段如图 4.9 所示。不同



南京邮电大学硕士研究生学位论文

第四章 电网多智能体协调控制与仿真

类型用户的用电习惯不同，导致其对于空调温度设定值的接受范围也不同，同时空调的开关段时间不同，三类空调可接受的温度设定值如表 4.3 所示，三类空调在 24 小时内的开启时段如图 4.9 所示。

表 4.2 空调负荷的基本参数

参数	$R$	$C$	$P$	$T_{out}$	$\Delta T_{ac}$
数值	2℃/kW	10kWh/℃	2kW	37℃	2℃

表 4.3 三类空调可接受的温度设定值范围

类型	$T_{sp,min}$	$T_{sp,max}$
商业用户	16	28
居民用户	20	28
工业用户	19	24

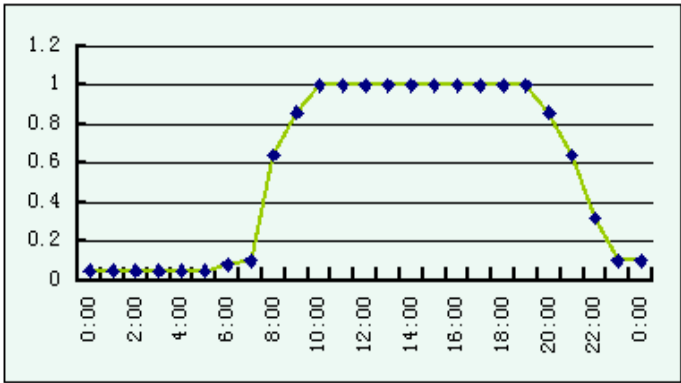


图 4.9 (a) 商业空调用电时段分布图

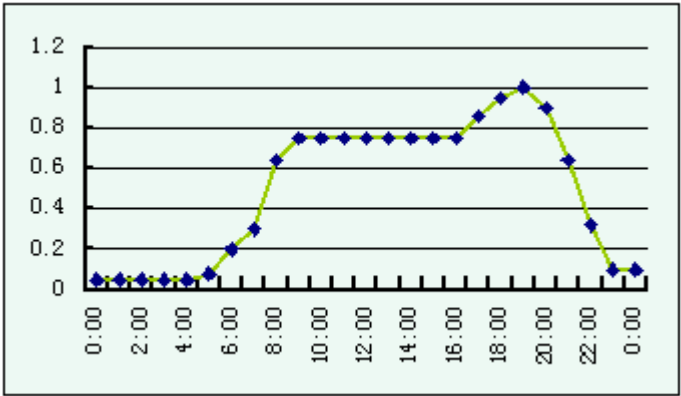


图 4.9 (b) 居民空调用电时段分布图

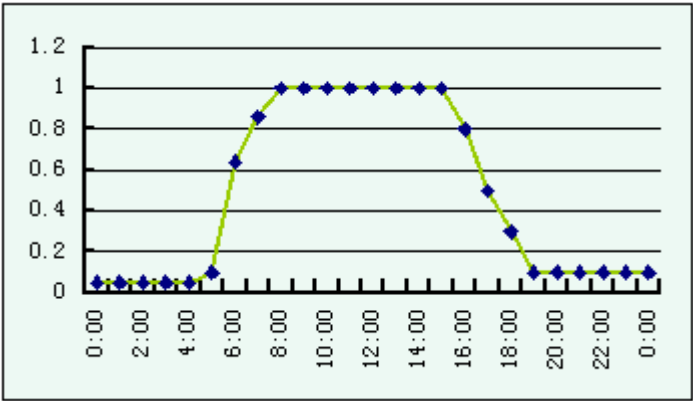


图 4.9 (c) 工业空调用电时段分布图

如图 4.10 所示，NETLOGO 三维的仿真结果中，空调用户层中的每台空调在该层面上按照自身运行特性工作，空调在开与关两个状态下变换（白色表示空调状态为关闭，红色表示空调状态为开启），空调状态在不断变化的同时空调温度设定值也随电价改变，即空调的运行特性也做出相应的改变，从而导致空调聚合群的需求量也在不断地变化。MATLAB 根据整个系统的功率平衡情况，发布具有牵引价值的电价指导空调做出响应，循环整个仿真过程，观察 24 小时内空调的响应情况。

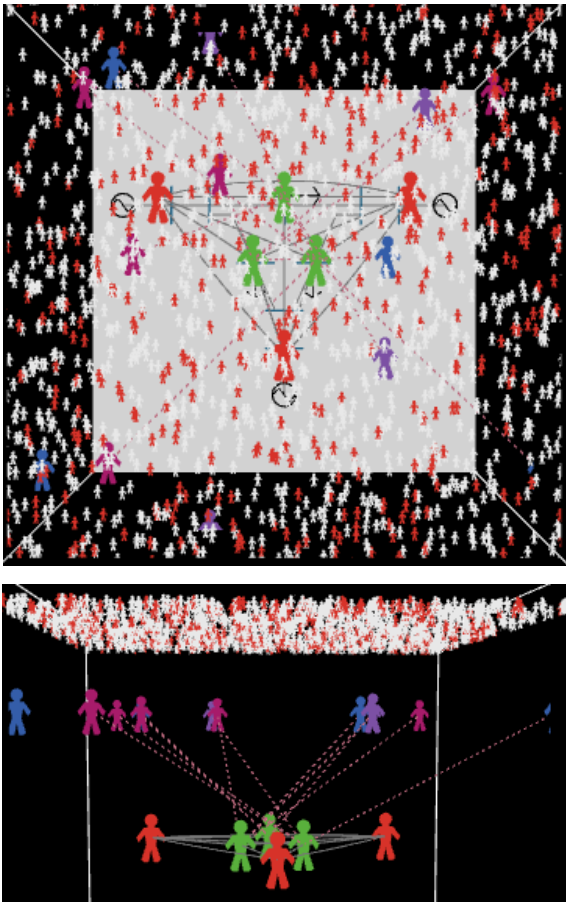


图 4.10 NETLOGO 三维的仿真结果

我们对 1500 台空调进行仿真，其中负荷节点 5 上有商业空调 50 台、居民空调 300 台、工业空调 150 台；负荷节点 7 上有商业空调 100 台、居民空调 180 台、工业空调 120 台；负荷节点 9 上有商业空调 60 台、居民空调 360 台、工业空调 180 台，其功率变化曲线图如图 4.11 所示。图中 ac1、ac2、ac3 分别为负荷节点 5、7、9 上的负荷需求量在 24 小时内的变化情况，ac4、ac5、ac6 分别为某一个商业空调聚合体、居民空调聚合体、工业空调聚合体的负荷需求量在 24 小时内的变化情况。从图中可以看出由于单个空调工作呈现周期变化导致空调聚合体的功率有一定的波动，从而导致整个负荷节点上的负荷需求量变化波动比较大。同时根据功率曲线可以明显看出空调在不同时段开启数量不同，夜间开启数量较少，而白天空调开启数量明显急剧上升，充分体现了空调在不同时段的开启数量情况。

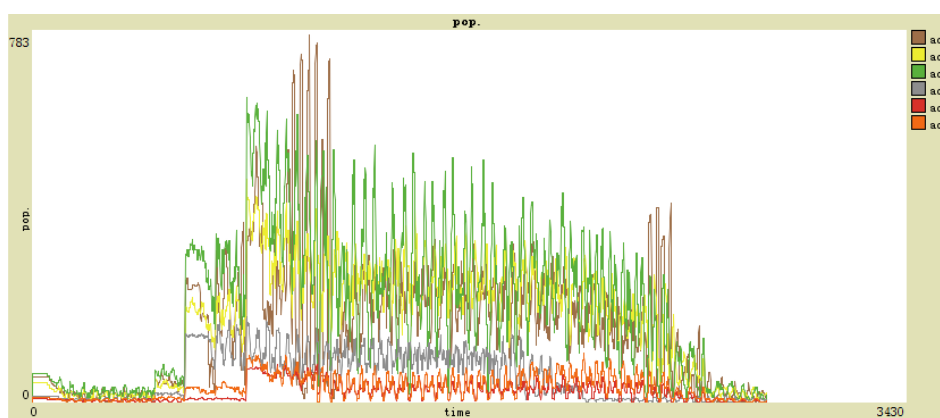


图 4.11 1500 台空调仿真的功率曲线变化图

我们对 5000 台空调进行仿真，其中负荷节点 5 上有商业空调 200 台、居民空调 800 台、工业空调 500 台；负荷节点 7 上有商业空调 500 台、居民空调 600 台、工业空调 400 台；负荷节点 9 上有商业空调 500 台、居民空调 900 台、工业空调 600 台，其功率变化曲线图如图 4.12 所示。由于空调数量的改变，各个空调聚合体以及负荷节点上的功率需求变化情况明显不同。

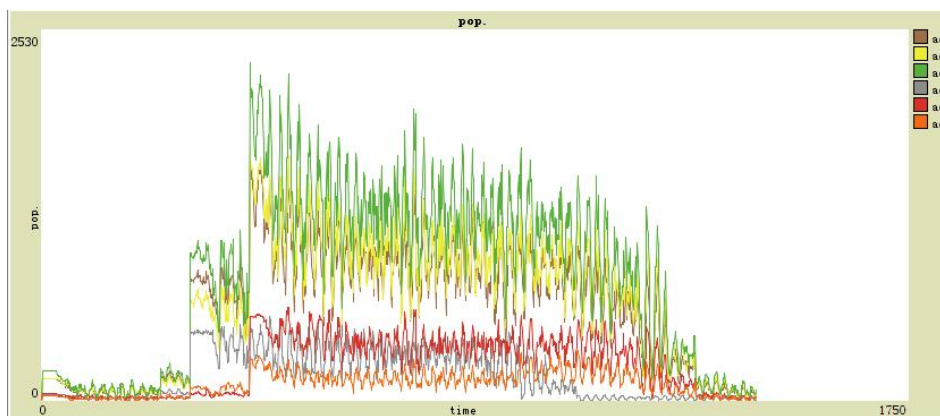


图 4.12 5000 台空调仿真的功率曲线变

## 第五章 总结与展望

### 5.1 全文总结

(1) 概述了源网荷互动电网中智能电网元件与多智能体理论体系之间的关系，特别是电网元件智能特性与多智能体系统的关联性，介绍了在多智能体理论体系中电网智能元件的准确表达，包括逆变器同步及多智能体功率平衡模型，并网型电动汽车电网频率偏差多智能体控制模型。

(2) 提出了若干个电网多智能体协调控制策略，包括考虑电力元件互动的电网优化控制策略，考虑柔性负荷互动的电网功率平衡控制策略，以及空调参与电网功率平衡调节的优化策略。

(3) 设计了基于 NETLOGO 与 MATLAB 联合的电网多智能体建模、仿真与控制方案：1) 采用 NETLOGO 对电网中电力元件受到激励信号自主改变用电模式的过程进行仿真。2) 采用 MATLAB 对电网进行电力方面的各项计算，保证电力系统能够安全可靠的运行。3) 对基于 NETLOGO 与 MATLAB 联合的电网多智能控制进行模拟仿真，验证了该建模、仿真与控制方案的可行性。

### 5.2 进一步工作的展望

(1) 论文提出的考虑电动汽车的电网功率平衡控制策略，尚没有在 NETLOGO 和 MATLAB 仿真平台上实现。

(2) 论文在 NETLOGO 和 MATLAB 仿真平台上开发了电网多智能体建模、仿真与控制的技术和方案，但在 NETLOGO 模型库中目前搭建的负荷模型较少，还需要进一步完善。

## 参考文献

- [1] 殷树刚, 张宇, 拜克明. 基于实时电价智能用电系统[J]. 电网技术, 2009, 33(19): 11-16.
- [2] 阮文骏, 王蓓蓓, 李扬, 等. 峰谷分时电价下的用户响应行为研究[J]. 电网技术, 2012, 36(7): 86-93.
- [3] 李广凯, 李庚银. 电力系统仿真软件综述[J]. 电气电子教学学报, 2005, 27(3): 61-65.
- [4] 陈皓勇, 王锡凡, 王秀丽, 等. 基于Java的电力市场竞价实验平台设计、实现及应用[J]. 电力系统自动化, 2004, 28 (17): 22-26.
- [5] 邵丽琴, 管晓宏, 高峰. 基于.net平台的电力市场仿真系统的实现[J]. 电网技术, 2004, 28(13):60-64.
- [6] 王玉荣, 魏萍. 电力市场仿真和培训系统的设计及实现[J]. 电力系统自动化, 2007, 3(12):96-99.
- [7] Xin H, Lu Z, Qu Z. et al. A cooperative control strategy for multiple photovoltaic generators in distribution networks[J]. IET Control Theory and Applications, 2011, 5(14): 1617–1629
- [8] Yang Hongming, Chung C Y, Zhao Junhua. Application of plug-in electric vehicles to frequency regulation based on distributed signal acquisition via limited communication[J]. IEEE Transactions on Power Systems, 2013, 28(2): 1017-1026.
- [9] 彭建飞, 任岷, 王树锦. MATLAB 在电力系统仿真研究中的应用[J]. 计算机仿真, 2005, 22(6): 193-196.
- [10] Tisue S, Wilensky U. Netlogo:A simple environment for modeling complexity[C]. International Conference on Complex Systems. 2004: 16-21.
- [11] 吴正平, 关治洪, 吴先用. 基于一致性理论的多机器系统队形控制[J]. 控制与决策, 2007, 22(11): 1241-1244.
- [12] 袁利平, 陈宗基, 周锐, 等. 多无人机同时到达的分散化控制方法[J]. 航空学报, 2010, 31(4): 797-805.
- [13] 王长城, 戚国庆, 李银舫, 等. 传感器网络一致性分布式滤波算法[J]. 控制理论与应用, 2012, 29(12): 1645-1650.
- [14] Olfati-Saber R, Fax J A, and Murray R M. Consensus and cooperation in networked multi-agent systems[J]. IEEE Proceedings, 2007, 95(1): 215–233.
- [15] Xin H, Qu Z, Seuss J, et al. A self-organizing strategy for power flow control of photovoltaic generators in a distribution network[J]. IEEE Transactions on Power Systems, 2011, 26(3): 1462–1473.
- [16] Zhang Z, Chow M Y. Convergence analysis of the incremental cost consensus algorithm under different communication network topologies in a smart grid[J]. IEEE Transactions on Power Systems, 2012, 27(4): 1761–1768.
- [17] Xin Huanhai, Zhang Meidan, Seuss J, et al. A real-time power allocation algorithm and its communication optimization for geographically dispersed energy storage systems[J]. IEEE Transactions on Power Systems, 2013, 28(4): 4732-4740.
- [18] Wood A J, Wollenberg B F. Power generation, operation, and control[M]. New York: Knovel, 1996.
- [19] 刘萌, 褚晓东, 张文, 等. 负荷分布式控制的云计算平台构架设计[J]. 电网技术, 2012, 36(8): 140-144.
- [20] Xu Yinliang, Zhang Wei, Gabriela Hug. Cooperative control of distributed energy storage systems in a microgrid[J]. IEEE Transactions on Smart Grid, 2015, 6(1):238-248.
- [21] Bidram A, Davoudi A, Lewis F L, et al. Distributed cooperative secondary control of microgrids using feedback linearization[J]. IEEE Transactions on Power Systems, 2013, 28(3):3462-3470.
- [22] Karlsson D, Hill D J. Modelling and identification of nonlinear dynamic loads in power systems[J]. IEEE Transactions on Power Systems, 1994, 9(1): 157-166.
- [23] Praca I, Ramos C, Uale Z, et al. MASCEM:A Multi-agent system that simulates competitive electricity markets[J]. IEEE Intelligent Systems, 2003, 8(6):54-60.

- [24] Yu Nanpeng, Liu Chencheng, James P. Evaluation of market rules using a multi-agent system method[J]. IEEE Transactions on Power Systems, 2010, 25(1):470-479.
- [25] 王锡凡. 现代电力系统分析[M]. 科学出版社, 2003:113-162
- [26] Arrillaga J, Arnold C P. Computer analysis of power systems[M]. New York: Wiley, 1990.
- [27] Bompard E F, Han Bei. Market-based control in emerging distribution system operation[J]. IEEE Transactions on Power Delivery, 2013, 28(28):2373-2382.
- [28] 姚明涛, 胡兆光, 张宁, 等. 工业负荷提供辅助服务的多智能体响应模拟[J]. 中国电机工程学报, 2014, 34(25): 4219-4226
- [29] 黄伟, 黄婷, 周欢, 等. 基于改进微分进化算法的微电网动态经济优化调度[J]. 电力系统自动化, 2014, 38(9): 211-217.
- [30] 孙惠娟, 彭春华, 易洪京. 大规模风电接入电网多目标随机优化调度[J]. 电力自动化设备, 2012, 32(5): 123-128
- [31] 纪昌明, 周婷, 向腾飞, 等. 基于网格搜索和交叉验证的支持向量机在梯级水电系统隐随机调度中的应用[J]. 电力自动化设备, 2014, 34(3): 125-131
- [32] Andrea R D, Dullerud G E. Distributed control design for spatially interconnected systems[J]. IEEE Transactions on Automat. Control, 2003, 48(3): 1478-1495.
- [33] Huang A Q, Crow M L, Heydt G T, et al. The future renewable electric energy delivery and management (FREEDM) system: The energy internet[J]. IEEE Proceedings, 2011, 99(1): 133-148.
- [34] Bertsekas D P, Tsitsiklis J N. Parallel and distributed computation: numerical methods[M]. Englewood Cliffs, NJ: Prentice Hall, 1989.
- [35] Chow M Y, Zhang Z. Convergence analysis of the Incremental cost consensus algorithm under different communication network topologies in a smart grid[J]. IEEE Transactions on Power Systems, 2012, 27(4):1761-1768.
- [36] Kar S, Hug G. Distributed robust economic dispatch in power systems: a consensus + innovations approach[C]. Power and Energy Society General Meeting, 2012: 1 - 8.
- [37] Zhang Z, Chow M Y. The leader election criterion for decentralized economic dispatch using incremental cost consensus algorithm[C]. The 37th Annual Conference of the IEEE Industrial Electronics Society (IECON), 2011: 2730-2735.
- [38] Zhang Z, Ying X C, Chow M Y. Decentralizing the economic dispatch problem using a two-level incremental cost consensus algorithm in a smart grid environment[C]. Proceedings North American Power Symposium, 2011: 1-7
- [39] Borowsky E, Gafni E, Lynch N, et al. The BG distributed simulation algorithm[J]. Distributed Computing, 2001, 14(3):127-146.
- [40] 段炜, 胡兆光, 姚明涛, 等. 考虑需方响应资源的电力系统生产模拟[J]. 电网技术, 2014, 38(6): 1523-1528.
- [41] 刘小聪, 王蓓蓓, 李扬. 智能电网下计及用户侧互动的发电日前调度计划模型[J]. 中国电机工程学报, 2013, 33(1): 30-38.
- [42] 李杨, 王治华, 卢毅. 峰谷分时电价的实施及大工业用户的响应[J]. 电力系统自动化, 2001, 25(8): 50-52.
- [43] 钟宇峰, 黄民翔, 叶承晋. 基于电池储能系统动态调度的微电网多目标运行优化[J]. 电力自动化设备, 2014, 34(6): 114-121
- [44] 石庆均, 江全元. 包含蓄电池储能的微网实时能量优化调度[J]. 电力自动化设备, 2013, 33(5): 76-82
- [45] 王海波, 杨秀, 张美霞. 平抑光伏系统波动的混合储能控制策略[J]. 电网技术, 2013, 37(9): 2453-2458.
- [46] 李蓓, 郭剑波. 平抑风电功率的电池储能系统控制策略[J]. 电网技术, 2012, 36(8): 39-43, 2012.

- [47] Horn R A, Johnson C R. Matrix analysis[J]. University of Alberta.downloaded on December at from IEEE Xplore.restrictions Apply, 1985:1-17.
- [48] 周磊, 李扬, 高赐威. 聚合空调负荷的温度调节方法改进及控制策略[J]. 中国电机工程学报, 2014, 34(31): 5579-5589.
- [49] Bashash S, Fathy H K. Modeling and control of aggregate air conditioning loads for robust renewable power management[J]. IEEE Transactions on Control Systems Technology, 2013, 21(4): 1318-1327.
- [50] 汤奕, 邓克愚, 孙华东, 等. 智能家电参与低频减载协调配合方案研究[J]. 电网技术, 2013, 37(10): 2861-2867.
- [51] 汤奕, 鲁针针, 宁佳, 等. 基于电力需求响应的智能家电管理控制方案[J]. 电力系统自动化, 2014, 38(9): 93-99.
- [52] 张素香, 刘建明, 赵丙镇, 等. 基于云计算的居民用电行为分析模型研究[J]. 电网技术, 2013, 37(6): 1542-1546.

## 附录 1 攻读硕士学位期间撰写的论文

(1) 金珍、吴英俊、岳东, 基于 NETLOGO 与 MATLAB 联合的负荷分布式控制仿真方案, 南京邮电大学学报, 2015. 06;



## 附录 2 攻读硕士学位期间申请的专利

- (1) 金珍、吴英俊、岳东，一种基于 Matlab 和 Netlogo 的电力系统负荷多智能体控制方法，201410850131.9，22014.12.31；
- (2) 吴英俊、王晓亮、金珍，一种基于 Matlab 和 Netlogo 的电力系统联合仿真建模方法，201410851143.3，2014.12.31；
- (3) 陈凯旋、谢俊、岳东、吴英俊、雍太有、金珍，一种基于自适应粒子群算法的楼宇算法的楼宇一种基于自智能用电策略获取方法，201510154779.7，2015.04.02；
- (4) 雍太有、李亚平、金珍，一种基于 Netlogo 与 Matlab 的负荷分布式控制方法，201510725409.4，2015.10.29；
- (5) 吴英俊、金珍、岳东、谢俊，一种家庭用户智能家电的用电安排优化方法，201510119071.8，2015.03.18。

### 附录 3 攻读硕士学位期间参加的科研项目

(1) 国家电网公司重点科技项目“‘源-网-荷’互动环境下电网功率与频率控制关键技术的研究”。

## 致谢

本论文主要是在我的导师岳东教授和吴英俊老师的细心指导下完成的，在撰写论文的两个多月中，我遇到了很多的困难与问题，每当我遇到困难，我的导师总是牺牲自己的时间帮助我修改论文，解答我的问题。能够完成论文的撰写离不开两位老师给予我的帮助，在这里对我的导师岳东教授和吴英俊老师致上最诚挚的谢意。当然，我能成功完成论文的工作也离不开我周围的同学们，在我遇到解不开的疑惑时，我的同学愿意牺牲自己写论文的时间与我讨论问题，尽自己的努力为我解答疑惑。每一次和同学讨论完，不仅让我解开了原本的疑惑，同时也开拓了我的思路，想到了可能之前从未考虑到的问题，真的非常感谢给予我帮助的小伙伴们。是你们在我有问题时与我讨论，是你们在我遇到挫折时鼓励我，谢谢你们。除了我的老师，我的同学，一直支持我鼓励我前进的是我的爸妈，是你们一直以来的理解、支持、鼓励，才能让我无论遇到什么问题都用积极向上的心态面对，才能让我勇敢的面对一切好的或不好的事，谢谢你们，我亲爱的爸爸妈妈！

本论文工作是基于国家电网公司重点科技项目“‘源-网-荷’互动环境下电网功率与频率控制关键技术的研究”而开展的。我很荣幸能够在研二的时候参与该项目的工作，并且能够在中国电力科学研究院实习。我非常感谢给我这次实习机会的老师以及电科院的领导们，我也非常珍惜能够在电科院向各位领导前辈们学习的机会，在电科院的实习期间我真的学到了很多书本上学不到的东西，在这里我要非常感谢给予我帮助的电科院的雍太有博士、李亚平老师和毛文博老师。在我刚进入电科院接触该项目工作时什么都不懂，当时的我拼命查阅相关资料，想快点了解项目的工作内容，快点加入大家的工作，帮助老师们完成相关工作，但是因为我的基础比较薄弱导致学习较慢，压力也非常大，幸好有我的老师、同事、同学给我帮助，我才能顺利完成我的项目工作。在这里对所有参与该项目的老师同学致上诚挚的谢意，是你们的帮助和鼓励才能让我顺利完成该项目的相关工作，完成该论文的撰写，谢谢你们！

最后，也要感谢本论文涉及到的各位学者。本文引用了多位学者的研究文献，从你们的研究文献中我得到了很多的启发，在撰写论文的过程中正因为有你们的研究让我解决了很多的问题。由于本人学术水平有限，文中难免有不足之处，欢迎各位专家和老师指正批评！