

A NOTE *of* MASTER

菜鸟硕士手册

V1.0

@LonelyBag

2019 年 3 月 12 日

如果说我比别人看得更远，那是因为我站在  
巨人的肩膀上

# 目录

<b>第一章 试着不虚此行</b>	<b>1</b>
1.1 假如重新来过 . . . . .	1
1.2 COMSOL . . . . .	2
<b>第二章 快速抓住研究热点</b>	<b>3</b>
2.1 传统方式 . . . . .	3
2.2 辅助工具 . . . . .	4
2.2.1 安装 . . . . .	4
2.2.2 导出文献 . . . . .	4
2.2.3 文献分析 . . . . .	5
<b>第三章 罗马是如何建成的</b>	<b>9</b>
3.1 是时候学点管理学了 . . . . .	9
3.1.1 什么是工作流 . . . . .	9
3.1.2 工作流与科研 . . . . .	9
3.2 理论分析 . . . . .	11
3.2.1 为什么要建数学模型 . . . . .	11
3.2.2 真实与仿真 . . . . .	11
3.3 实验验证 . . . . .	12
3.4 文献写作 . . . . .	12
3.4.1 写作工具 . . . . .	12
3.4.2 写作流程 . . . . .	12
3.4.3 参考文献 . . . . .	12
跋	13



# 第一章 试着不虚此行

## 1.1 假如重新来过

假如我回到了 3 年前再次读研。我觉得我会这样度过：

- 研一上，认真学好数值分析、矩阵论、智能控制理论、电力电子、DSP 以及文献检索（比较有代表性的，不代表其他不重要）。原因：数值分析和矩阵论是一切仿真软件内核的数学基础；智能控制理论涉及到了人工神经网络的基本原理，而它是未来的必然趋势，此外书中提到的算法都可以解决优化问题，如：电路拓扑的构建、电路元件参数的调整、线圈形状的优化等等。。。；电力电子电路和 DSP 的结合是除了仿真外验证思路的最优实现方案；文献检索，可能是硕士期间最重要、也最值得花时间研究的课。
- 研一下，只做两件事。第一：系统地学习 MATLAB+Simulink、COMSOL 或 Ansys、DSP、Altium Designer、ADS 或 Multism、Latex+git+vscode、Rhino 或 3DSMAX、PS 或 Adobe Illustrator。第二：整理近两年的论文。
- 研二上（开题 10 月底），精进 MATLAB+Simulink、COMSOL 或 Ansys、DSP、Altium Designer、ADS 或 Multism、Latex+git+vscode，并着手仿真实验 + 写论文引言。
- 研二下（中期 5 月底），完成实物实验，完成论文并在 11 月份前投出。
- 研二上，着手第二篇论文，在 9 月份前投出。
- 研三上（毕业答辩 1 月底），完成毕业论文。

这将是异常丰富的经历，如果坚持下来，你的人生可能会因此改变。

## 1.2 COMSOL

## 第二章 快速抓住研究热点

研究热度往往体现在论文的发表数量中，但是一个领域中的主要研究方向却往往很难仅通过阅读论文而获得全面认识。换句话说，统计论文中的研究方向远比统计发表数量重要。为了解决这一问题，研究人员也提出了若干解决方案。

### 2.1 传统方式

词典，是语言学家针对特定语言而将常见字词按照一定的逻辑编纂起来的文本工具。科学家也有自己的词典，他们针对某一特定研究领域对热点论文进行总结归纳并提出相应见解，最终以论文的形式发表，这就是文献综述。

通过阅读文献综述，可以快速地对所关注的研究领域形成大致了解，而这也是快速抓住研究热点的传统方式。通过限定检索词可以较为有效地检索该类型的文章，对于英文文献，其检索词可以选择如下几种：

- Review
- Challenge
- Survey
- Statement
- ... ..

这样，再加上一些限定词（如：WPT）就可以有效地检索出特定研究领域的文献综述。但是，这样检索出的文献往往比较松散，并且会使我们忽略一些更有价值的 paper。因此，有时（甚至往往）会采用辅助工具进行这项工作。

## 2.2 辅助工具

为了快速地了解一个领域的研究热点，仅仅通过**关键词检索 - 论文下载 - 阅读 - 二次检索**这类流程是不能达到“快速”的要求的，真正有效的方法是借助论文分析工具，这类工具一般有如下特点：

- 可以分析大量文献间的交叉检索关系
- 结果可视化
- 多功能

常用的辅助工具有：**HistCite Pro 2.1** 以及 **VOSviewer**。这两种工具的最大区别就是 HistCite Pro 2.1 仅支持英文文献，但 VOSviewer 不仅支持英文还支持中国知网导出的中文文献。笔者只用过 HistCite Pro 2.1，如果读者希望对中文文献也进行处理，可以参考[这个](#)）。下面我通过若干 gif 对软件的使用进行介绍。

### 2.2.1 安装

安装就不赘述了，参考[这里](#)就够了。

### 2.2.2 导出文献

这一步是初学者最容易出现差错的地方，这是因为 HistCite Pro 2.1 对源数据的格式要求非常严谨，而且它仅支持由 Web of Science 导出的文献格式。下面我通过 Step-by-step 的方式展示如何从 WOS 导出符合要求的格式。

**第一步：**选择检索**数据库**并给定检索词，一般首次使用需要我们对整个领域有一个宏观认识，因此检索词可以非常概括，比如我使用的 wireless power，如图2.1。

**第二步：**选择正确格式，下载，如图2.2所示。值得注意的是，WOS 仅支持一次性导出 500 篇文献，所以更多的文献需要多次下载，这步完成后会获得一个包含所有文献检索信息的 txt 文件。






图 2.1: WOS 检索关键词: wireless power




图 2.2: WOS 下载特定格式检索结果

### 2.2.3 文献分析

将下载获得的所有 txt 移动至 TXT 文件夹, 然后打开 main.exe, 此时输入 1 并回车, 软件将自动打开程序界面, 接着点击 graph maker, 我们就会获得文献之间的引用网络, 文献由节点表示, 所有的节点自上而下按照时间顺序排列, 之间的引用和被引关系由连接的线条表示。修改 count 将对分析的文献数量进行控制, 一般来说, 取得大一点可以获得更多的信息, 如图2.3所示。

注意到, select by 一栏有两个选项: LCS 和 GCS, LCS 是 Local Citation Score 的缩写, 代表某篇文献在本地数据集 (也就是我们所下载的 txt 集合) 的总被引次数, 而 GCS 是 Global Citation Score 的缩写, 代表在 WOS 数据库中的总被引次数。一般来说, 由于本地数据集是根据我们的关键词获得的, 因此 LCS 排名更能够反映某一学科的发展。此外, 还有两个参数会出现在

图 2.3: HistCite Pro 2.1 操作

浏览器界面, CR 和 LCR, 分别是 Cited references 和 local cited references 的缩写, 代表对 WOS 数据库和本地数据库文献引用的数量。这里, CR 越高(一般以 50 篇为标准)则表示该文献越可能是综述性文献, 这样就可以方便地追根溯源, 清晰地看到一个学科乃至一个领域的发展脉络, 一个典型的发展脉络可以参考图2.4(完整版请[移步](#))。

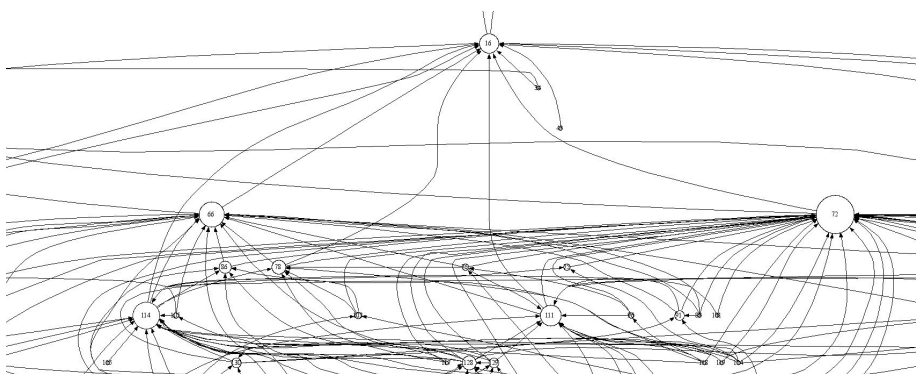


图 2.4: 一个示例

可以看出, 文献 16 显然是这个领域的开创性论文, 随着时间的推移又分裂为两个分支, 接着又发展为多个分支。通过阅读关键节点论文不仅可以快速了解这个领域的发展脉络, 甚至还可以看出:

- 研究之初, 两个小方向是经过试探却又半途而废的
- [完整版图片](#)右侧出现了两个领域的交叉, 这里往往是研究思路的“不

老泉”

- 完整版图片最左侧有一个研究历程非常持久的小分支，这可能是某个专家的“独门秘籍”
- 完整版图片右侧还有一个完全没有关联的领域，虽然在 2013 年后就没有产出了，但是这并不一定意味着研究的停滞，也有可能是研究成果实现了产业化，被大佬们拿去挣钱了。。。
- .....

所以，这个方法不仅仅可以帮助我们找文献，更重要的是带给了我们关于这项研究的历史发展历程，带给了我们冰冷研究的人文情怀，我们甚至可以想象出，那位开山鼻祖级别的大佬在当时仅仅是一位年轻博士生的时候的艰苦、悲痛、遇到机遇时狂喜、受挫后的再一次站立、以及功成名就之后的平凡和悠闲。。。

因此，每一张图都可以说是一位科学家的成名史、一个领域的发展史、以及整个人类文明的历史。

所以，为什么不试试它或它呢？



## 第三章 罗马是如何建成的

### 3.1 是时候学点管理学了

#### 3.1.1 什么是工作流

我平常接触 github 比较多（一个著名的代码托管网站，这是[我的主页](#)）。我发现，在进行大多数的代码项目前，大家都会提到一个叫做 workflow（可以翻译为工作流）的东西。经过了解后，我发现这个东西非常强大，其本质就是所谓的“思路”或者“技术路线”，并且除了给出方法论以外，它还告诉了我们如何进行团队协作。

workflow 源于 Github，因此先对 Github 进行介绍。Github 由 Linus Torvalds 编写的版本控制工具 Git 发展而来，这两样工具的发明最初仅是为了服务于[Linux](#)项目的开发。Linux 作为一个操作系统内核，一个人不可能进行维护，而 Github 的出现，使得世界范围的代码爱好者可以共同维护一个项目，而这在以前是绝无可能的。

如果说 Github 提供了一个巨型的合作平台，那么 workflow 则提供了开发者们共同协作的机制。一个典型的 github workflow 如图[3.1](#)所示。

#### 3.1.2 工作流与科研

看完这张图大家是不是就明白了，APP 中的版本标签 V1.1, V1.2 其实就是这么来的。

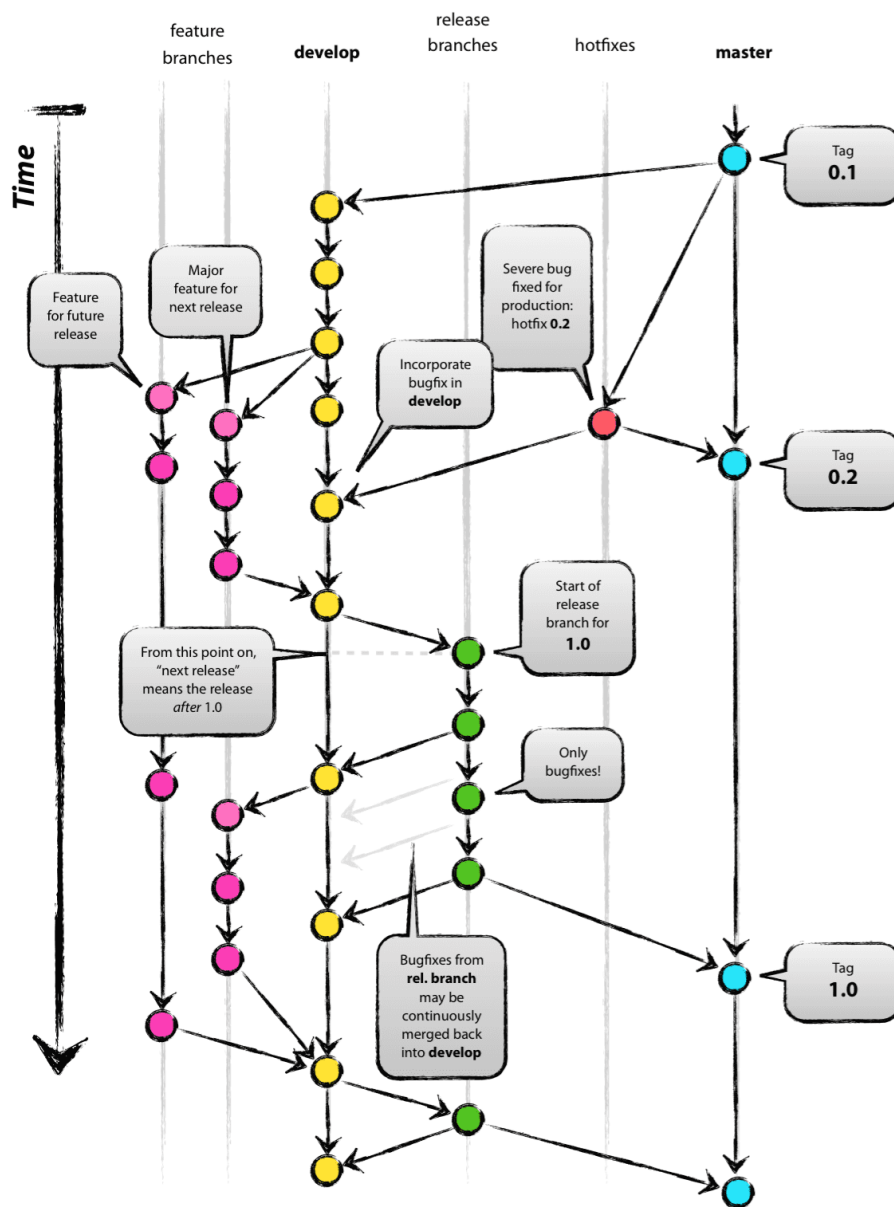


图 3.1: workflow

## 3.2 理论分析

### 3.2.1 为什么要建数学模型

大到宇宙洪荒，小到亚原子领域，与人类活动相关的一切都有数学模型。那么，为什么要研究呢？我觉得目的可以分为两种，为了控制。

### 3.2.2 真实与仿真

钱学森有一个世纪之问：“为什么中国出不了诺贝尔奖”，我自己有一个解答，不是从教育的角度，而是从国人与老外的思考方式的差异。

如果大家看过《降临》这部电影就会了解，语言决定了（或者反映了）一个人的思考方式，电影中的外星人具有非线性的高维语言能力，没有时间空间的线性概念，这也是电影中人类语言学家不能理解外星人文字的症结。我们可以仿照着看一下汉语与英语的差别，如表3.1：

表 3.1: 中英对比

对比	汉语	英语
奇怪的量词	少	多
感情色彩	委婉	直接
时态	无	有

可以看出，中国人的语言是弱化量词而强化情感的，我们作画作诗也是这样，总是注重写意而避免写实，描述人与自然的的关系的时候也总是旁击侧引，生怕赏画吟诗的人一眼就看穿作者的心思。但是外国人并不是这样，他们致力于用最简练最准确的语言描述事物，生怕哪个词读者不明白而影响了本意。

从牛顿时代开始，这个社会的生产力就是纯粹的科技力量了，而科技恰恰要求的是高精度：优化算法是为了更加精确和更加快速地求解方程、学习CAD 是为了更加精确地描述事物的几何特性、发明电子计算机是为了更加精确地制导。。。看到这里，不知道读者是否明白了我的意思。

那么，数学模型的建立，使得我们可以通过量化的方式对系统进行控制与优化，

### 3.3 实验验证

### 3.4 文献写作

#### 3.4.1 写作工具

#### 3.4.2 写作流程

#### 3.4.3 参考文献



# 跋

我讲几个事例作为结束，其中蕴含了我的一些感悟。

## 热爱

时常有人问我：“你怎么会有那么多时间去学那么多东西的？”。其实很简单，因为热爱。

## 持久的输出

知乎上有一个问题，苹果公司有哪些黑科技？其中一个回答的答案是 iPhone 一代。2007 年 1 月，苹果公司发布了被后人称为“重新定义了手机”的 iPhone 一代，自此开始了长达十年的封神之路。作为历史上市值第一个超过万亿美元的科技公司，它的成功也让很多人萌发了试图复制这一商业成功的欲望，当然他们并未成功，因为十年后的今天，苹果依然还是第一。很多人将苹果的成功归功于乔布斯，这种想法其实很愚蠢。那么到底是什么造就了它的成功呢？

如果大家读过《乔布斯传》就会了解到，2007 年发布的 iPhone 一代，其实早在 5 年前就已经开始准备了。那就是 2002 年，还是各种翻盖、滑盖、旋转盖手机迭起的风云时代。。。

我们输的不冤枉吧。对手感觉到的“黑科技”，其实对自己，不过是孕育和必然。

## 活得痛快

人为什么活着？

不知道大家看过《本杰明巴顿奇事》吗，可能剧情也不错吧，但是给我带来感悟的却是电影快结束时的旁白。每个人都生而平凡，但是工作却赋予

了我们不同的社会角色和人生意义：有的人是妈妈，有的人是商人，有的人是舞者、有的人是政客、有的人是船长。。。

他们不一样吗？

他们太一样了：妈妈的愿望是孩子活下来、商人的愿望是挣钱后让自己的孩子继承工厂、舞者的愿望是在世界的舞台上独舞、政客的愿望是包养舞者、船长的愿望是为国捐躯。。。看到了吗，他们都有一个共同的东西在支撑自己去奋斗、去生活，那就是梦想。

试着寻找自己的内心所爱，也试着让那一份热爱去做决策吧。相信我，当你找到她的时候，你将成为无限宇宙之王。