东北大学软件学院

本科生毕业设计（论文）撰写说明

**目 录**

[一　引言 1](#_Toc28038)

[二　毕业论文的选题 1](#_Toc20032)

[（一）论文选题注意事项 2](#_Toc30912)

[（二）课题来源 3](#_Toc6564)

[（三）毕业设计（论文）题目 4](#_Toc639)

[三　毕业论文撰写 4](#_Toc8011)

[（一）摘要与关键词 5](#_Toc21756)

[1 摘要的含义 5](#_Toc6429)

[2 摘要的具体写法 5](#_Toc16588)

[3 关键词的选取 6](#_Toc17021)

[4 摘要撰写注意事项 6](#_Toc5214)

[5 摘要的翻译 6](#_Toc5710)

[（二）软件开发类正文撰写 6](#_Toc29814)

[1 绪论 7](#_Toc3592)

[2 相关技术（关键技术） 9](#_Toc17762)

[3 需求分析（系统分析） 9](#_Toc29328)

[4 系统设计 10](#_Toc4866)

[5 系统实现 11](#_Toc5294)

[6 系统测试 11](#_Toc5587)

[7 总结及展望 11](#_Toc12887)

[（三）算法设计类正文撰写 12](#_Toc9634)

[1 引言（绪论） 12](#_Toc14119)

[2相关工作与理论基础 15](#_Toc11928)

[3 算法的分析与设计 17](#_Toc14776)

[4 算法的实现与评测 21](#_Toc9133)

[5 结论 23](#_Toc25795)

[（四）参考文献 23](#_Toc465)

[（五）致谢 23](#_Toc12840)

[四　附录 24](#_Toc20813)

[（一）用例图 24](#_Toc20304)

[（二）业务流程图 25](#_Toc16163)

[（三）活动图 27](#_Toc32135)

[（四）数据流图 30](#_Toc30328)

[（五）功能模块图 31](#_Toc5694)

[（六）程序流程图 33](#_Toc25167)

# 一　引言

毕业论文是学术论文的一种形式，为了进一步探讨和掌握毕业论文的写作规律和特点，需要对毕业论文进行分类。由于毕业论文本身的内容和性质不同，研究领域、对象、方法、表现方式不同，不同的院系，不同的专业，毕业论文通常有不同的类型。就软件学院本科而言，毕业论文通常有以下两种类型：

1. 算法设计类论文

这一类型毕业设计的特点是带有探索性，经过文献调研后，对软件工程某一领域的先进技术或成熟产品进行分析、比较，进而提出自己的评价和有针对性的创见，对某一理论问题有一定见解，产生出一个题目（课题），利用自己所学的专业知识和数学工具，得出一个（些）有用（或者有潜在的价值）的结论，并能将该创新性技术用于自己研制的系统中。这类毕业设计工作要注意把分析和实验相结合，不要只停留在消化上。消化是前提，吸收和转化才是工作的重点。

这一类毕业设计一般先根据具体条件建立一个数学模型，推导出相应的表达式，利用计算机计算出结果，然后对结果加以分析，再提出结论性意见。最好将研究成果应用到实际进行验证。

1. 软件开发类论文

这一类型的毕业设计主要依据所学的知识，完成一个相对完整的实际项目或在某一个较大的项目中设计并完成一个子系统，或者对已有的软件产品进行改进。最后以软件工程的思路，结合项目开发文档，进行归纳总结，综合扩展形成论文。

本文会分别介绍这两类论文的撰写原则和内容指导，重点介绍如何进行毕业设计选题，论文应该包括哪些内容，每部分内容应占论文总体篇幅多大比重，以及应该如何用语言、图表来进行说明。

# 二　毕业论文的选题

毕业论文的选题要与学院的培养目标相联系，重在培养学生综合运用所学专业理论知识去解决实际问题的能力，使其受到科学研究的基本训练，所以选题一般不应超出专业内容的范围。毕业论文的题目应符合所学专业范围和知识结构的基本要求，毕业设计的内容要紧密结合当前国家经济发展和最新科学技术进展，做到既要有理论与方法的研究，又要有应用前景。

毕业论文选题时需要有具体的应用背景，围绕着某应用系统中的某个“复杂工程问题”设计解决方案，并进行实现测试（软件开发类）或实验验证（算法设计类）。其中，“复杂工程问题”必须具备下述特征（1），同时具备下述特征（2）-（7）的部分或全部：

（1）必须运用深入的工程原理，经过分析才可能得到解决；

（2）涉及多方面的技术、工程和其它因素，并可能相互有一定冲突；

（3）需要通过建立合适的抽象模型才能解决，在建模过程中需要体现出创造性；

（4）不是仅靠常用方法就可以完全解决的；

（5）问题中涉及的因素可能没有完全包含在专业工程实践的标准和规范中；

（6）问题相关各方利益不完全一致；

（7）具有较高的综合性，包含多个相互关联的子问题。

## （一）论文选题注意事项

毕业论文选题时，常见的问题有下列几种：

(1) 选题过大；

(2) 选题过难；

(3) 选题陈旧。

因此，在选题过程中，应注意以下几个问题：

(1) 选题时应对自己有正确的客观估计

客观评价自己掌握材料的深度和广度，驾驭材料的能力，对课题的理解程度等。根据自己的长处和兴趣爱好，扬长避短充分发挥主观优势。同时还要充分了解学术界的研究现状。如，本课题研究已有的成果，还存在哪些问题，尚待研究的问题，尚待解决的问题及迫切程度，社会需要和科学发展的趋势，另外，只有把主客两方面的条件结合起来，才能选出最适合自己的课题来。

(2) 课题难易要适度

选择的课题难易要适度。难度大的课题当然更有科学价值，但对刚刚涉足科学领域的大学生来说，往往力不胜任，难以完成。而难度小的课题，学生就会失去一次科学研究规范训练的机会，达不到写作毕业论文的目的。因此，课题既要有一定的难度，有一定的工作量，又要结合学生的知识水平和实际能力。

(3) 课题大小要得当

毕业论文主要是反映学生能否运用所学基础和专业知识来分析和解决本学科内某一基本问题的学术水平和运用能力。所以，毕业论题不可能囊括学习期间的全部知识，也不可能解决本学科的全部问题。一篇毕业论文只需论述某一基本问题的某一重要侧面，或是对某些基本的理论、原理有比较系统的整理等。因此，在选题时，要根据学生的专业基础和时间及其它相关因素，如资料条件、经费许可，指导力量等，综合考查以选择大小适当的课题。否则，课题过大，问题均难以研究深入，可能导致虎头蛇尾，草草收摊；题目过小，不能充分挖掘学生的潜力，发挥才能，论文达不到应有的水平和深度，也反映不出学生的实际功底和能力。

## （二）课题来源

软件学院鼓励学生毕业设计到企业去，承担企业真实项目。因此大部分同学应通过各种渠道（例如，导师推荐，就业实习，自己联系等方式）进入企业，融入企业真实项目中，根据前一节的选题注意事项，在导师的指导下，选定合适的论文题目。

但是由于我院毕业设计学生情况不同，论文题目不能全部来自企业。因此不同情况需不同处理。我院毕业设计学生分成以下几种情况：

(1) 本科毕业即就业

此类学生有相当比例可到就业单位实习，如若就业岗位为软件开发岗，可与就业单位沟通，根据我院选题注意事项，选定毕业论文题目。如果就业岗位非软件开发岗（如产品推广、测试或非软件行业），可根据企业实际生产或运营需求，选定软件开发项目。如若不能到就业单位实习，或实习单位不能提供符合软件学院要求的论文题目，请参考第3条。

(2) 保送学生

保送学生可参与到自己导师的科研项目中，导师通常愿意从项目中选取本科生能完成的模块，交给学生作，然后写成论文。教师熟悉项目，项目有实用背景，一般而言，多数学生经过努力都能完成。

(3) 无实习单位学生

由于学生考研、出国等原因没有找到合适的企业实习，此类学生需尽快与校内导师沟通。

① 可由校内导师推荐到企业实习，或者自己联系短期实习单位，此时课题来源于企业。

② 有些学生对一些问题有自己独到的看法和理解，知识面较宽，理论基础较深厚，学生可根据自己的科研兴趣及爱好拟定论文题目，与校内教师沟通，由校内导师认定。此时课题属学生自选题目。

③ 校内教师根据学生能力情况，及自己科研方向拟定题目。

总之，毕业论文的题目大部分应来源于企业，其次是教师科研项目的子课题，最后才是学生自选题目。选题要尽量早些，以便有充分的时间积累材料，也有足够的时间和精力深入探讨。如果选题太晚，就会显得很仓促，无暇把问题考虑成熟并加以实现。

## （三）毕业设计（论文）题目

对毕业设计论文题目的要求是：

(1) 简短精炼

**论文题目必须正确无误，且不得超过24个汉字。**

(2) 准确得体

论文题目应是以最恰当、最简明的词语反映论文中最重要的特定内容的逻辑组合，应避免使用不常见的缩略词、首字母缩写字、字符、代号和公式等。

(3) 外延和内涵要恰如其分

毕业论文的题目应仔细推敲，尽可能从各个角度充分考虑，选择最合适的。原则上，题目要简单明了，能反应毕业论文的主要内容，使读者能一眼看出论文的中心内容要讲什么，切忌笼统、空泛。

(4) 醒目

毕业论文的标题不能像小说、散文那样经过艺术加工而引起读者的好奇心。语言要朴实，同时也能引起读者的注意。

# 三　毕业论文撰写

东北大学软件学院本科毕业设计（论文）的撰写有以下几点要求：

(1) 内容要正确充实；

(2) 格式要符合标准要求；

(3) 篇幅要满足要求

**毕业论文的撰写不应占用全部毕业设计时间，也不应该在毕业设计实习结束，回到学校之后大概一周的时间一蹴而就。**而是应该采用循序渐进，日积月累的方法。在企业实习期间，在做工程项目或科研课题的间歇，分阶段、分篇章的搜集整理论文素材，待到实习结束，回到学校之后，总结整理这些论文素材，形成一篇毕业设计论文。

毕业论文的写作步骤应该首先确定论文提要材料，论文页数和字数的大致分配，拟定论文的写作提纲，进而对论文提纲进行细化和扩展，形成论文初稿。通过与指导教师沟通讨论，对初稿进行修订，包括观点的订正、深化，材料的增、删、改、换，结构及语言的修订，最后形成一篇合格的本科毕业设计论文。

**毕业设计论文一般包含摘要、正文、总结与展望、致谢等几个章节，论文正文篇幅应在40页以上（论文正文指从绪论至总结与展望）**。下面就逐一介绍各部分应该论述的内容、撰写原则，及占总篇幅的比例。

## （一）摘要与关键词

### 1 摘要的含义

摘要是毕业论文的高度概括和总结，是论文内容不加注释和评论的简短陈述。摘要应具有独立性和自明性，拥有同正文同等量的主要信息，其叙述语言应简洁，准确，让读者尽快了解论文的主要内容，以补充题名的不足，即不阅读全文就能获得必要的信息。另外，摘要（特别是关键词）也可为科技情报文献检索数据库的建设和维护提供方便。

### 2 摘要的具体写法

摘要的撰写一般分三段式来阐述，字数在400~700之间，以英文译文一页之内为宜。不要出现第一人称我或我们的字样，要从客观的角度来阐述。

① 第一段描述论文研究或开发课题的背景、当前的问题，从而引出为什么要研究或开发该项目（意义）。一般3-5行足以。

② 第二段是摘要的主体部分，对全文进行总概，描述毕业论文所阐述的研究内容、研究方法和研究成果。如：

**软件开发类论文的摘要主体：**

本文首先调研了××客户或××系统，分析了××××方面的国内外研究现状，对×××业务所存在的主要问题进行了阐述，提出开发××××系统。接下来，对×××进行了业务需求分析和功能需求分析，在此基础上采用××××框架，结合××××技术，使用×××工具，设计并实现了×××，包含×××主要模块和×××关键算法。最后，采用××××测试方案，设计×××测试用例，对开发的系统进行了完整的测试，测试结果验证了系统的有效性，达到预期的效果。

**算法设计类论文的摘要主体：**

本文首先分析了××××方面的国内外研究现状，对×××所存在的主要问题进行了阐述，重点对×××问题和×××问题进行了研究。针对××××，提出了一种基于××××的算法，利用××××，结合××××，设计了×××，用以解决×××××（或者：在××××算法的基础上，利用××××，结合××××，对算法进行了改进）。针对××××的问题，从×××的角度出发，提出了××××的算法，用以提高××××的×××性（稳定性、安全性、鲁棒性等等，看实际情况选）。最后，将开发的算法以×××的形式嵌入到×××系统中，满足了×××系统在×××方面的需求。另外，还使用×××对该算法进行了通用性测试，测试结果表明了算法的（正确性）和（合理性），但在算法的可扩展方面还需进一步的研究（请根据实际的结果进行书写）。

③ 第三段可从论文工作的特点和贡献上进行一下总结，描述论文所阐述的研究（开发）成果，对现实社会造成的影响、意义。

### 3 关键词的选取

关键词是能描述所写论文成果、创新性、所在领域等的名词，一般在3-5个。是从论文的题名、摘要和正文中选取出来的，是对表述论文的中心内容有实质意义的词汇。关键词选用是否合适，关系到该论文被检索的概率和该成果的利用率。

### 4 摘要撰写注意事项

摘要的篇幅不易过长，最好撰写摘要时，应注意以下几点：

① 使用第三人称（尽量不要出现我、我们等字样）；

② 一般不要对论文内容作诠释和评论（尤其是自我评论）；

③ 不要与“绪论”和“总结”中大段内容重复。

④ 不得一再简单重复文章题目；

⑤ 慎用长句，采用规范化术语，新术语可使用原文或在译名后加括号注明原文；

⑥ 缩略词、略称、代号，除了相邻专业的读者也能清楚理解的之外，在首次出现时必须加以说明。

⑦ 不得出现正文中的图号、表号、公式、章节号以及参考文献等；

⑧ 除了实在无法变通以外，一般不用数学公式，不出现插图、表格。

### 5 摘要的翻译

一般而言，英文摘要应是中文摘要的转译，所以只要简洁、准确地逐段将文章译出即可，时态常用一般现在时间、一般过去时，少用或不用现在完成时、过去完成时、进行时态和其他复合时态。尽量使用短句，但也要避免单调和重复。

## （二）软件开发类正文撰写

正文是论文的绝对主体部分，我院软件开发类的论文正文一般包括“绪论”、“相关技术（或关键技术）”、“需求分析（或系统分析）”、“系统设计”、“系统实现”、“系统测试”等几个章节，论文正文篇幅应在40页以上。下面就逐一介绍各章节应该论述的内容、撰写原则，及占总篇幅的比例。

### 1 绪论

绪论又称引言、前言或导论。绪论是论文的开场白，目的是向读者说明本研究的来龙去脉，吸引读者对本篇论文产生兴趣，对正文起到提纲掣领和引导阅读兴趣的作用。

在写绪论之前首先应明确几个基本问题：你想通过本论文说明什么问题？你研究开发的系统是否有实际应用价值？甚至有理论研究价值？一般读者读了绪论以后，可清楚地知道作者为什么选择该题目进行研究并开发。为此，在写绪论以前，要尽可能多地了解相关的内容，收集前人和别人已有工作的主要资料，说明本研究开发的合理性。

**(1) 绪论的篇幅不应超过总论文的10%，即3~5页即可。**不要与摘要雷同。一般教科书中有的知识，在绪论中不必出现。

(2) 言简意赅，突出重点。不应过多叙述同行熟知的及教科书中的常识性内容，确有必要提及他人的研究成果和基本原理时，只需以参考引文的形式标出即可。在绪论中提示本文的工作和观点时，意思应明确，语言应简练。

(3) 内容不应与摘要雷同，注意不用客套话，如“才疏学浅”、“水平有限”、“恳请指正”、“抛砖引玉”之类的语言。

(4) 引言中不要插图、列表，不进行公式的推导与证明。

#### 1.1 绪论的结构

绪论作为论文的开头，首先要阐明选题的实际背景和解决该问题的现实意义和重要作用等，使读者感到此选题确实具有实用价值和学术价值，有研发和开发的必要性。其次，应简述本课题在国内外的研究和发展状况，分析前人研究或开发的系统的一些问题，比如不适应XX场景或背景，有哪些不足，从而引出本论文课题的研究内容；接着，说明本课题的具体范围及预达到的技术要求。若属子课题，在引言中还应对主课题的全貌加以介绍，说明本人的工作内容以及在整个课题中所起的作用和关系；也可以阐述本课题研究的指导思想、欲解决的主要问题以及解决此课题所需要的条件；也可适当简要地介绍一些与本课题有关的预备知识，但不必展开讨论。因此一篇学位论文的引言，大致包含如下几个部分：

（1）课题研究背景及意义

（2）国内外研究现状

（3）论文研究内容

（4）论文组织结构

接下来对这几部分的写法进行简述。

#### 1.2 课题研究背景及意义的写法

软件开发类论文要具有较强的实际应用价值。所以，论文工作需要有具体的应用背景，明确阐述论文开发系统在应用中能解决的实际问题。其次，要经过文献调研，针对此领域欲解决的问题和目前问题解决方案的不足，提出一定的见解，产生出新的一套解决方案作为课题，利用自己所学的专业知识和技术工具，可以实现此方案。研究背景中需要阐述清楚2个大问题：

(1) 研究的课题项目“是什么”，解决什么应用系统下的什么问题。

(2) 为什么选择这个项目来开发，即阐述该研究课题的现实意义，比如说明该研究课题能够解决给定应用系统中的什么现实问题，对企业生产制造有什么贡献，该研究课题具有一定的创新性等。

#### 1.3 国内外研究现状的写法

对本研究课题范围内的文献进行详尽的综合述评，在“阐述”的同时一定要有“评价”，指出现有研究成果的不足或不适应性，提出自己的开发思路。

应简述本课题在国内外的研究和发展状况；针对课题的实际背景和要解决的问题，对比国内外学者的相关工作，阐述清楚国内外学者对同类问题所采用的研究和解决方法，同时对比这些研究和解决方法的优缺点。当然也可适当简要地介绍一些与本课题有关的预备知识。

注：国内外研究现状与第2章的相关工作是有区别的。国内外研究现状描述课题研究的大背景大方向，侧重讲述课题研究的先进性和重要意义，不用涉及具体技术。

#### 1.4 论文研究内容的写法

通过对国内外研究现状的分析，针对该课题现有不足的或急需解决的问题，阐述清楚自己使用的科学研究方法，包括需要解决什么系统中的什么问题，解决该问题采用的理论依据、研究方法、工具框架和数据来源等，预期的成果内容及其地位、作用和意义。

#### 1.5 论文组织结构的写法

**例如：本论文结构安排如下：**

第1章，绪论。介绍了课题的研究背景、意义、国内外研究现状、发展特点和趋势，论文的组织结构。

第2章，相关技术。简要介绍了高压发生器的系统构成、CPU单元结构及uCLinux嵌入式操作系统简介及驱动程序开发概述。

第3章，需求分析。通过用例的方式对高压发生器的控制软件进行需求分析，包括功能性需求分析和非功能性需求分析，进而得出高压发生器的用例模型。

第4章，系统设计。进行架构设计及软件功能设计，对软件进行分层和模块划分。将软件分为硬件接口层、驱动程序层和应用程序层；将软件划分为硬件接口模块、控制模块、算法模块和数据模块。

第5章，系统实现。实现了高压控制软件，给出硬件接口层模块、驱动程序层各驱动程序、应用层各模块的具体实现。

第6章，系统测试。对高压基本功能编写测试用例，进行测试，得到相关波形。

第7章，总结与展望。对工作做了简要的总结，并对后续工作提出了设想。

### 2 相关技术（关键技术）

相关技术或关键技术章节不是论文组织所必需的章节，如果论文中所涉及的相关理论或关键技术不是很常见，对后续论文的理解需要该知识，则有必要在相关理论或关键技术这一章节进行简要介绍。**其篇幅不应超过总论文的15%，即5~8页即可**。

相关技术或关键技术主要包括两部分内容：

(1) 与课题内容相关的理论知识。

(2) 课题项目开发用到的关键技术介绍，包括开发方法、开发工具和环境等方面的内容。

**最后写一段本章小结**，最好不要少于3行。

### 3 需求分析（系统分析）

需求分析（系统分析）是论文组织所必需的章节。**其篇幅应占总论文的15%左右。**

如果论文所研究的课题属于一个相对较独立的完整课题，则应该进行较完整的需求分析。如果论文所研究的课题属于某个较大项目的子课题，则应该首先进行系统分析，对主课题的全貌加以介绍分析，说明本人的工作内容以及在整个课题中所起的作用和关系。然后再重点对自己研究的子课题进行较全面的需求分析。

软件需求包括三个不同的层次：业务需求、用户需求和功能需求。其中，业务需求和用户需求中包括功能性和非功能性需求。

需求分析建模的方法有多种，包括结构化方法、面向对象模型等。建模工具主要有四种，用例图、活动图、业务流程图、数据流图。它们之间的关系如表2.1所示。

表 2.1 需求分析建模方法

|  |  |
| --- | --- |
| **结构化方法** | **面向对象方法** |
| 业务流程图 | 用例图 |
| 数据流图 | 活动图 |

(1) 功能分析可从静态和动态两种角度来进行分析，静态分析过程中可使用“**用例图**”来进行描述，从业务的角度进行动态分析可使用“**业务流程图**”或“**活动图**”来描述，从数据的角度进行动态分析可使用“**数据流图**”来描述。附录中逐一介绍了这几种分析工具。

(2) 非功能性需求分析可包括可行性分析（技术可行性、经济可行性等）、安全性分析、性能效率分析等等。

**最后写一段本章小结**，最好不要少于3行。

### 4 系统设计

系统设计是论文组织所必需的章节。**其篇幅应占总论文的30%以上。**

系统设计也可采用结构化设计方法和面向对象的设计方法。

(1) 传统的结构化设计方法主要有3种：

① 功能模块划分设计；② 面向数据流的设计；③ 输入输出设计。

(2) 面向对象的设计方法主要有对象设计和动态模型设计。

① 对象设计：对象模型描述系统的静态结构，由**类图**和**对象图**表示，包括构成系统的类和对象、它们的属性和操作，以及它们之间的联系。

② 动态模型设计：动态模型确定对象的可能事件的顺序。动态模型由**状态图、时序图、协作图**等表示。

系统设计可分为概要（总体）设计和详细设计。

(1) 概要（总体）设计可包括体系结构设计和模块设计。体系结构设计一般需论述采用何种架构（B/S或C/S架构），何种设计模式和框架（MVC等），何种开发环境及工具等等。

模块设计一般应画出总体功能模块图。（参考附录功能模块图画法）

(2) 详细设计可分模块的详细介绍设计过程，包括算法设计、功能设计，数据库设计，接口设计、协议设计、界面设计等。可通过“**时序图**”、“**类图**”、“**E-R图**”及**数据库表**等设计工具进行更直观的阐述。

**最后写一段本章小结**，最好不要少于3行。

**需要注意的是**，论文中的图表只是对文字论述的一个补充说明，不能是论文的主体，论文还应是以论述为主，图表为辅，所以不能连续页面都是图表。

### 5 系统实现

系统实现是论文组织所必需的章节。**其篇幅应占总论文的20%以上。**实现这一章节的安排应与设计一章相呼应，设计一章出现了哪几个模块的设计，在实现一章应有相应的模块实现。每一模块的实现可通过“**程序流程图**”、“**代码**”和“**界面**”更加直观的论述。

程序流程图要规范，有“开始”，有“结束”，分支要使用“菱形框”，要有分支条件，跳转箭头要指向流程线，而不能指向“执行框”。具体见附录图3.15所示。

代码要选取关键代码或伪代码，篇幅不可太长，连续代码不应超过一页。

界面图不应太多，几个足以。

**最后要写一段本章小结**，最好不要少于3行。

### 6 系统测试

测试是论文组织所必需的章节。**其篇幅应占总论文的7%左右。**

测试一章的内容没有必要大段大段摘抄测试原则、测试方法等，应重点描述测试方案、测试用例和测试结果，**最后给出测试结论或评价**。

测试用例可以表格的形式给出测试用例序号、功能、输入、预期结果、实际结果等

表 2.2 测试用例表格示例1（仅供参考）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | **用例名** | **测试步骤** | **预期结果** | **测试结果** |
| 1 | 正向测试－登录功能－用户名、密码和验证码正确 | 1. 输入正确的用户名、账号和密码 2. 点击登录按钮 | 登录成功，跳转进入系统 | 与预期结果一致 |
| 2 | 界面兼容性测试－登录页面－不同浏览器 | 1. 在不同浏览器打开登录界面 2. 输入正确的用户名、密码和验证码 3. 点击登录按钮 | 登录成功，跳转进入系统 | 与预期结果一致 |
| 3 | 逆向测试－登录功能－验证码错误 | 1. 输入正确的账号和密码 2. 输入错误的验证码 3. 点击登录按钮 | 登录失败，提示验证码输入错误 | 与预期结果一致 |

表 2.3 测试用例表格示例2（仅供参考）

|  |  |
| --- | --- |
| 测试用例 | 查看交易回测策略测试 |
| 测试目的 | 验证查看策略功能是否正常 |
| 测试条件 | 用户成功登录进入查看策略界面 |
| 测试过程 | 1. 用户点击查看策略 2. 用户填写策略名称 3. 用户点击搜索 |
| 期望结果 | 若用户已经登录则可以查看、搜索系统支持的策略信息 |
| 测试结果 | 通过 |
| 说明 | 无 |
| 测试日期 | 2023年5月14日 |

### 7 总结及展望

总结是对整个研究工作进行归纳和综合而得出的总结，对所得结果与已有结果的比较和课题尚存在的问题，以及进一步开展研究的见解与建议。总结要写得概括、正确、完整、明确、精炼。

总结不是个人总结，不是自己在毕业设计期间的流水帐，在总结中要以整个研究工作为主体，进行阐述相关的问题。

总结与摘要不同：总结主要是对课题研究内容进行总结，摘要是对论文本身进行概括。

## （三）算法设计类正文撰写

正文是论文的绝对主体部分，一般需要包括“引言”、“相关工作与理论基础”、“算法的分析与设计”、“算法的实现与评测”、“结论”以及参考文献部分。在篇幅上，算法类论文的正文部分需要在40页以上。当涉及到多个算法时，可以扩展对应部分内容，也可以按算法划分章节、组织内容，同时篇幅也可相应增加。本文仅对论文写作的结构进行说明，不涉及到论文的排版格式。有关排版格式，请参考其他文献。

### 1 引言（绪论）

引言又称绪论，前言或导论。引言是开篇之作，[论文](http://www.5ykj.com/Article/" \t "_blank)的开场白，目的是向读者说明本研究的来龙去脉，吸引读者对本篇论文产生兴趣，对正文起到提纲掣领和引导阅读兴趣的作用。

在写引言之前首先应明确几个基本问题：你想通过本文说明什么问题？有哪些新的发现，是否有学术价值？一般读者读了引言以后，可清楚地知道作者为什么选择该题目进行研究。为此，在写前言以前，要尽可能多地了解相关的内容，收集前人和别人已有工作的主要资料，说明本研究设想的合理性。

（1）引言的篇幅不应超过总论文的10%。应言简意赅，不要与摘要雷同。一般教科书中有的知识，在引言中不必出现。

（2）言简意赅，突出重点。不应过多叙述同行熟知的及教科书中的常识性内容，确有必要提及他人的研究成果和基本原理时，只需以参考引文的形式标出即可。在引言中提示本文的工作和观点时，意思应明确，语言应简练。

（3）内容不应与摘要雷同，注意不用客套话，如“才疏学浅”、“水平有限”、“恳请指正”、“抛砖引玉”之类的语言。

（4）引言中不要插图、列表，不进行公式的推导与证明。

#### 1.1 引言（绪论）的结构

引言作为论文的开头，要结合文中算法所面向的具体应用系统，以简短的篇幅介绍论文的写作背景和目的，缘起和提出研究要求的现实情况，以及相关领域内前人所做的工作和研究的概况，说明本研究与相关工作的关系，目前的研究热点、存在的问题及作者的工作意义，引出本文的主题给读者以引导。引言也可点明本文的理论依据、实验基础和研究方法，简单阐述其研究内容；简要概括预示本研究的结果、意义和前景，但不必展开讨论。因此一篇学位论文的引言，大致包含如下几个部分：

(1) 研究背景；

(2) 国内外研究现状；

(3) 研究内容；

(4) 论文组织结构。

接下来将对这几部分的写法进行简述。

#### 1.2 研究背景的写法

算法类论文要具有应用性和探索性。首先，论文工作需要有具体的应用背景，属于给定应用系统中的一个带解决的实际问题。其次，要经过文献调研，针对此领域欲解决的问题和存在的问题有一定的见解，产生出一套解决方案作为课题，利用自己所学的专业知识和数学工具，得出一个（些）有用（或者有潜在的价值）的结论和有价值的数据结果。研究背景中需要阐述清楚2个大问题：

（1）研究的问题“是什么”，属于什么应用系统下的什么问题。

（2）为什么选择这个题目来研究，即阐述该研究的现实意义，比如说明该研究能够解决给定应用系统中的什么现实问题，对学科发展有什么贡献，该研究具有一定的前沿性等。

研究内容一是“立题”的背景，说明论文选题在本学科领域的地位、作用以及目前研究的现状，特别是研究中存在的或没有解决的问题。二是针对现有研究的状况，确立本文拟要解决的问题，从而引出下文。

#### 1.3 国内外研究现状的写法

对本研究主题范围内的文献进行详尽的综合述评，“述”的同时一定要有“评”，指出现有研究成果的不足，讲出自己的改进思路。

应简述本课题在国内外的研究和发展状况；针对课题的实际背景和要解决的问题，对比国内外学者的相关工作，阐述清楚国内外学者对同类问题所采用的研究和解决方法，同时对比这些研究和解决方法的优缺点。当然也可适当简要地介绍一些与本课题有关的预备知识。

**注：**国内研究现状与第二章的相关工作是有区别的。国内外研究现状描述课题研究的大背景大方向，侧重讲述课题研究的先进性和重要意义，不用涉及具体技术。

#### 1.4 研究内容的写法

通过对国内外研究现状的分析，针对该课题现有不足的或急需解决的问题，阐述清楚自己使用的科学研究方法，包括需要解决什么系统中的什么问题，解决该问题采用的理论依据、研究方法和实验基础，预期的结果及其地位、作用和意义。

在研究内容的最后需写清楚本研究的创新点或理论与(或)实践意义。如果研究的项目是别人从未开展过的，这时创新性是显而易见的，要说明研究的创新点。但大部分情况下，研究的项目是前人开展过的，这时一定要说明此研究与被研究的不同之处和本质上的区别，而不是单纯的重复前人的工作。如果要引出新的概念或术语，则应加以定义或阐明。

#### 1.5 论文组织结构的写法

论文的组织结构是对整篇论文的概述，阐述清楚论文的章节，每一章的研究内容或者介绍章和章间的关系。比方说《无线传感器网络上的数据聚集调度算法》论文在对论文组织结构进行书写时按如下方式描述：

本论文共分5章，每章的组织结构安排和内容如下：

第1章是引言。本章简要介绍了无线传感器网络的基本知识、数据聚集问题以及数据聚集调度问题的意义、国内外对于传感器网络研究现状以及传感器网络上的数据聚集及调度问题的研究现状、本文的主要贡献等。

第2章是相关工作。本章介绍了无线传感器网络中数据聚集的相关知识以及数据聚集调度的相关方法并提出了其中的问题。

第3章是传感器网络中数据聚集的分布式调度算法分析与设计。本章在对以前的研究工作进行全面分析的基础上，提出了一种在无线传感器网络中数据聚集的分布式调度算法。

第4章是传感器网络中数据聚集的分布式调度算法实现。我们对该算法的正确性给出了证明，并且对算法的性能进行了理论上的分析，包括时间延迟（优化目标），通信复杂性和时间复杂性。同时，我们利用模拟实验验证了该算法具有较低的时间延迟和较少的通信开销。

第5章结论，给出了本文的结论以及未来工作。

### 2相关工作与理论基础

论文的相关工作指的是与论文所研究内容直接相关的同行研究进展，在行文过程中需要有准确的参考文献支撑。一般来说，相关工作与第一章绪论篇幅加起来不能超过全文的1/3。

#### 2.1 相关工作

针对研究的问题，目前他人已经做了哪些方面的研究，提出了哪些算法，简单总结其解决的问题与相关算法，并指出现有算法的不足之处。

注意这里的相关工作不仅仅包含现有工作的简单描述，还要在现有的研究工作中进行对比，指出其中的不同之处，并且说明现有算法的不足之处，本论文要在哪些方面做改进等。

**下面是一个典型的相关工作示例：**

目前典型的道路网中移动对象连续k近邻查询处理算法有IMA/GMA算法[7]和ER2CkNN算法[8]。

IMA/GMA算法将移动对象数据、多用户并发查询、道路网数据全部组织在内存中。对于每一个查询来说使用网络扩张的方法获得其初始结果集，即从查询所在的位置开始，遍历周围的边及其上的移动对象，根据到移动对象的网络距离不断地更新查询结果集。此外，IMA 算法将网络扩张中遍历过的结点组织成一个称为查询扩张树的数据结构，基于这种数据结构，IMA算法通过判断边权重、移动对象位置、查询位置的变化，对扩张树进行修剪，从而重用扩张树中的查询结果。GMA算法则将路径（起点和终点的度数不等于2）上的查询组成一组，同一路径上查询的结果集是路径两个端点的k 近邻查询结果和该路径上移动对象并集的子集。基于这一性质GMA算法利用IMA算法监控每条路径两个端点的k近邻查询结果，并判断数据的更新会影响哪些查询的结果，而后重新计算这些受影响的查询。

通过分析和实验，发现IMA/GMA 算法的不足可以概括为：(1) IMA/ GMA 会进行大量计算以判断是否需要对查询重计算，当数据频繁更新时，绝大多数查询都需要重计算，因此性能急剧下降；(2) GMA 算法不能处理k 值不同情况下的多用户并发查询；(3)当道路网规模较大时，其基本的网络扩张算法性能下降。

ER2CkNN算法的提出者注意到了IMA/ GMA算法由于网络扩张造成的性能不佳，因而提出了预计算一种称为Edge Bitmap Encoding 的数据结构，通过这种结构能够快速计算给定两点的最短路径。在初始结果计算时，除了利用这一结构外，还采用了文献[9]中的欧氏距离限制的思想，即快速找到候选结果集，而后利用欧氏范围查询不断对结果集精炼得到最终结果。在维护数据更新时，与IMA/GMA算法类似需要进行大量判断，实现增量式的结果更新机制。

ER2CkNN 算法的不足可以概括为：(1) 当移动对象数据频繁更新时，性能急剧下降；(2) Edge Bit2map Encoding 数据结构在道路网中边权重更新时需要重新计算。

此外，文献[10]中针对自由运动移动对象的连续k近邻查询处理问题，提出了基于流水线式多线程的查询处理框架和k近邻查询处理算法，证明了多核平台上多线程技术能够提高连续查询处理的性能。但是，道路网连续查询算法中的数据依赖使得流水线式多线程处理框架难以发挥流水线的优势，而正如引言中所述，基于欧氏距离的算法不适用于道路网上的距离计算。

正是注意到了已有算法在面对数据频繁更新时的性能低下问题，且多线程技术能够带来算法性能的提高，开展了本文的研究。

#### 2.2 理论基础

理论基础指的是解决该问题使用的一些基本理论，但不是必需内容。如果论文中所涉及的相关理论，对后续论文的理解需要该知识，则有必要在这一章节进行简要介绍。否则，没有必要进行介绍。

这里的基本理论指的是与所研究问题直接相关的内容，而不应是该领域内所有知识的全部概括。

**以下是理论基础的示例：**

信息论之父 C. E. Shannon 在1948年发表的论文“通信的数学理论（A Mathematical Theory of Communication)”中，Shannon 指出，任何信息都存在冗余，冗余大小与信息中每个符号（数字、字母或单词）的出现概率或者说不确定性有关。Shannon 借鉴了热力学的概念，把信息中排除了冗余后的平均信息量称为“信息熵”，并给出了计算信息熵的数学表达式。在现代信息论中，熵的定义为：

定义2.1：熵*H(X)*。

设X为离散型随机变量，有*n*个可能取值，它的一切可能取值为分别记为*x1，x2，……，xn*，，各取值所对应的概率为*p(x1)，p(x2) ，……，p(xn)*。这些概率满足以下公理条件

 （2.1）

则作为随机变量不确定性的度量—熵,可以定义为

 （2.2）

#### 2.3 本章小结

对该章的内容简单的总结。

### 3 算法的分析与设计

算法的分析与设计这一章是论文的核心内容，主要包括算法系统背景描述、算法定义、算法的概要设计、算法的详细设计、以及算法复杂度分析等内容。

这里仅给出了算法分析与设计的写作模板，如果论文工作涉及多个算法，可选择以下两种方式之一进行书写。其一是分为“算法的分析与设计”和“算法的实现与评测”两章，分别在这两章中同时展开多个算法的内容。其二可以按算法来划分章节，在每章中展开算法的分析、设计、实现和评测等内容。

#### 3.1 算法系统背景描述

要求论文中的算法必需有实际的应用背景，属于某应用系统中需解决的某个特定问题。该节内容主要包含：

（1）算法所面向的应用系统的大致介绍：该应用系统是一个什么样的系统，大致的功能架构是什么等；

（2）算法在系统中作用与地位：描述该算法解决系统中的什么问题，对应系统中的什么功能，对系统起到什么样的作用等；

（3）系统对算法的具体要求：针对系统具体的应用需求，描述算法需要在哪些方面满足什么样的要求。

#### 3.2 算法的定义

算法的定义是对算法所解决问题的详细、完整描述与定义。通常应该对所研究的问题用语言进行描述，然后对该问题采用类似于数学语言对其进行定义，最后列举实例对定义进行说明。通常包括以下三个部分，但根据具体问题可以适当的增减。

**（1）研究问题采用的模型或体系结构**

例如：研究分布式环境的查询问题，采用何种分布式结构（网状的、树状的等），在这里需要对其体系结构进行说明。

或者是采用的数据模型的说明，如研究随机变量问题。随机变量采用的是离散随机变量或者是连续随机变量等。

**（2）问题的定义**

应该采用科学严谨的语言对其进行定义（如数学符号），而不应是大段的文字说明。

**一个轮廓查询的定义如下所示：**

在数据库中，数据集的轮廓（skyline）由该数据集中所有不比其它任何元组差的元组组成。要形式化定义轮廓，就要首先定义两个元组之间的“好坏”关系，称之为支配关系，如定义3.1所示。

定义3.1 在关系数据库中，维度空间D上的数据集T中的元组*ti*支配元组*tj*当且仅当满足以下两个条件：

(1) 在维度空间D中的任何一维*k*，元组*ti*都不比*tj*差；

(2) 在维度空间D中存在一维*l*，元组*ti*都比*ti*好。

这里的元组维度数值之间的“好坏”关系，可以是用户指定的任意标准，既可以是“大于”、“小于”，也可以是“不同于”。在此以“小于”为例进行说明，

元组间的支配关系可以形式化表示为公式3.1 的形式。

 （3.1）

根据定义3.1 中的支配关系定义，可以得到轮廓的定义如定义3.2 所示。

定义3.2 在关系数据库中，数据集T的轮廓S是数据集 的子集(S⊆T)，S包含T中所有不被其它任何元组所支配的元组。

****

**（3）对问题的解释**

用具体示例，对问题进行描述。例如上例的skyline查询，可以用一个具体示例解释。

例如，对定义3.2采用图3.1对其进行解释说明。



图 3.1 酒店的轮廓

轮廓查询有着非常广阔的实际应用价值，例如，当你要去某个海滨度假胜地旅游的时候，你一定希望能找到一个既价格便宜，又距离海滩较近的酒店入住，以便使得本次消费的性价比达到最高。然而，事实却是离海滩近的酒店往往价格较高，而价格较低的酒店离海滩也较远，这种情况下，你将陷入两难的境地，变得难以取舍。轮廓查询刚好可以解决这个问题，在距离和价格两维上的轮廓中的酒店就是那些性价比最高的酒店，其余的酒店都会在轮廓中找到既比它价格低，也比它距离海滩近的酒店，

如图3.1所示，实心的圆点为轮廓点，而空心的圆点为非轮廓点。有了轮廓查询的帮助，你只需要在轮廓中的酒店之中进行选择就能找到最令你满意的酒店了。

#### 3.3算法的概要设计

根据上一节对算法的具体定义，概要描述算法的大致解决思路。该节的主要内容包含：

（1）用语言概要描述算法的大致解决思路；

（2）本文算法采用的技术手段。并阐述对原算法在哪些方面做了改进。

（3）设计算法运行所依赖的数据结构；

（4）描述算法的大致执行过程，以及各步中算法的大致计算方法。

#### 3.4算法的详细设计

和上一节相比，该节为算法的详细描述，表现形式为算法的伪代码或者是程序流程图。并用文字对算法进行详细、具体的描述。最后一般需要对所设计的算法进行复杂度分析。

**伪代码算法描述举例：**

BEGIN:

i = 0; //进化种群代数

Initialize P(i); //初始化种群

Fitness P(i); //计算适宜值

While(not Terminate-Condition) //不满足终止条件时，循环

{

i ++; //循环

GA-Operation P(i); //交叉、变异操作

Fitness P(i); //计算适宜值

}

END //结束算法

**步骤形式的算法描述举例：**

初始化：种群规模P，遗传代数N，交叉概率pc，变异概率pm，计数变量i=0；

(1) 生成初始种群。

(2) 计算初始种群中每个染色体的适宜值。

(3) i=i+1；如果i<=N，则生成赌轮，转步骤4；否则，转步骤9。

(4) 根据赌轮选择策略选择染色体生成基因池，即基因池中的染色体个数即为种群规模P。

(5) 根据交叉概率pc进行交叉操作，基因池中未进行交叉操作的直接复制到下一代。

(6) 根据变异概率pm，进行变异操作。

(7) 计算子代种群中每个染色体的适宜值。

(8) 子代种群取代父代种群，转步骤3。

(9) 计算种群中每个染色体对应路由数的网络延迟、出错率以及网络费用。按适宜值排序，适宜值最小的染色体即为求得的解。

流程图式的算法描述举例：

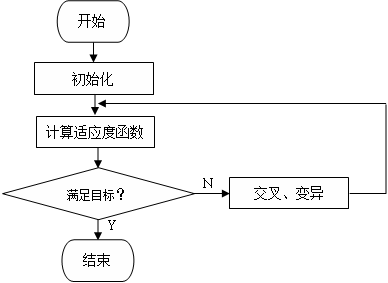


图 3.2 遗传算法流程图

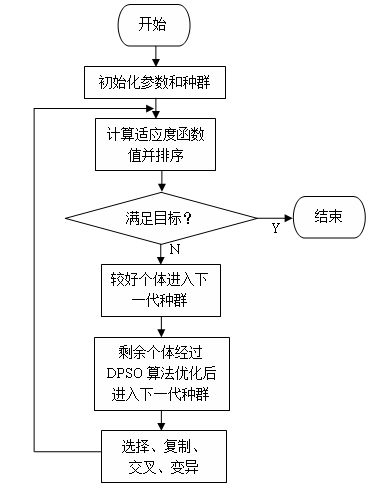
****

图 3.3 改进的粒子群遗传算法流程图

普通遗传算法流程图如图3.2所示，改进的粒子群遗传算法如图3.3所示。改进的粒子群遗传算法(MPSO/ GA)，该算法是以基本遗传算法为基础，同时将改进的粒子群算法作为遗传算法的一个重要算子，具体算法步骤如下：

(1) 设定参数，并随机产生初始种群。

(2) 计算每个个体的适应度函数值，并且按照适应度函数值进行排序。

(3) 判断是否满足目标条件（包括程序收敛以及达到指定的进化代数）, 如果满足，结束进程，输出结果；否则转步骤4。

(4) 更新个体种群。 根据适应度函数值的大小确定一部分个体直接进入下一代种群, 剩余个体通过MPSO 算法优化过后进入下一代种群。

(5) 对新一代种群执行遗传算法的复制、交叉和变异等操作, 转步骤2。

### 4 算法的实现与评测

#### 4.1 算法的实现

该部分主要是根据上一节所设计的算法，给出具体的编码实现，内容可按以下小节的方式展开：

**（1）×××算法的实现与部署**

如果论文算法分为几个模块或步骤完成，可以将各模块（步骤）组织成独立小节进行描述。

该部分内容可以包含算法实现的具体代码、算法的详细流程图、算法实现的详细类图等，以及不可缺少的关于算法实现的文字说明。

**（2）算法实现的结果展示**

该部分主要展示论文工作的最终结果。应包含证明算法实现的界面：可以是算法自带的GUI界面、应用系统使用该算法产生的结果界面、或者算法测试运行时产生的中间结果界面等。

#### 4.2 算法的评测

该部分主要是采用应用系统中的数据、也可以增添公共测试数据，对所实现的算法进行功能和性能等方面的测试。内容可按以下小节的方式展开：

**（1）实验环境**

该部分是对算法评测环境中涉及到的硬件、软件、及开发工具的说明。

例如：

针对以上约束条件以及目标函数设计编码序列模型, 在Matlab 7.0 环境下, 使用MPSO/ GA 算法进行仿真，运行环境是Pentium Dual E2104，116GHz，512MB，Microsoft XP。

**（2）实验数据**

对算法测试采用数据或产生的数据集规模等的说明。必须有来自于应用系统的实际数据，也可以增添公共测试数据或人工合成数据，均需要对实验评测时所使用的数据源进行详细的说明。

例如：

遗传算法参数设置如下：

为了评价本算法所产生的DNA 序列的性能，本文根据文献[8]的约束条件产生初始化种群。

①基本遗传算法，最大进化代数为300，种群规模为20，DNA 序列编码长度为20，交叉率为0.85, 变异率为0.005。

②MPSO算法，最大进化代数为200，学习因子分别为c1 = 2, c2 = 1.8，惯性权重因子w 从2 降低到0.18, 扰动因子u= 10, 最大速度为4。

**（3）评测结果分析**

① 给出具体实验结果，并对结果进行定量或定性的分析。写作要点是：以绘图和（或）列表（必要时）等手段整理实验结果。

② 得出实验结论：所开发的算法是否满足了其所面向的应用系统的需求，和其它通用算法相比，在哪些方法性能有优劣等。

算法的各个性能指标（例如时间、空间、通信量等）在不同参数变化的情况下进行分析，但是根据不同问题，分析的方面可以不同。

例如：

表 4.1 MPSO/ GA 算法产生的DNA 序列

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| DNA序列 | Hd | Id | GC成分 | Continuity | Hairpin | Fitness value |
| CGTGT GC AGT ACT GAGT ATG | 15 | 14 | 50 | -1 | -1 | 40 |
| AGTAGT TC TCAGACGCTGCT | 12 | 13 | 50 | 0 | 0 | 39 |
| GCATGAT CGATCT CGTCAGA | 14 | 13 | 50 | -2 | 0 | 39 |
| AT CGGT AGTCGT AGACGT CT | 12 | 15 | 50 | -3 | -2 | 38 |

表 4.2 GA 算法产生的DNA 序列

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| DNA序列 | Hd | Id | GC成分 | Continuity | Hairpin | Fitness value |
| GAGTT AGATGTC ACGT CACG | 15 | 14 | 50 | -1 | -4 | 37 |
| AGGC GAGT AGGGGTAT ATC T | 14 | 12 | 50 | -4 | -1 | 34 |
| T TAT GATT CCACT GGCGCT C | 13 | 13 | 50 | -2 | 0 | 33 |
| CCT GTCAACAT TGACGCT CA | 12 | 14 | 50 | -3 | -2 | 31 |

如表4.1与表4.2所示，MPSO/ GA 算法比文献[8]提到的普通GA算法的适应度函数值(Fitness value)有所提高。

### 5 结论

结论不是研究结果的简单重复，而是对研究结果更深入一步的认识，是从正文部分的全部内容出发，并涉及引言的部分内容，经过判断、归纳、推理等过程，将研究结果升华成新的总观点。其内容要点如下：

（1）本研究结果说明了什么问题，得出了什么规律性的东西，解决了什么×××系统中的×××实际问题，并在理论方面有哪些提升；

（2）对前人有关本问题的看法作了哪些检验，哪些与本研究结果一致，哪些不一致，作者做了哪些修正、补充、发展或否定；

（3）本研究的不足之处或遗留问题。

对于某一篇论文的“结论”，上述要点（1）和（3）是必需的，而（2）视论文的具体内容可以有，也可以没有。

结论里应包括必要的数据，但主要是用文字表达，一般不再用插图和表格。

## （四）参考文献

采用顺序编码制时，在引文处，按它们出现的先后用阿拉伯数字连续编码，并将序码置于方括号内，视具体情况把序码作为上角标，或者作为语句的组成部分。

## （五）致谢

致谢是论文写作者对在论文写作过程中帮助过他的所有老师、同学、家长及朋友的一种感谢，致谢写得是否用心，也可以直接看出作者是不是一个会感恩的人。大致可以对以下几类相关人员进行感谢：

（1）对论文指导老师进行感谢，大致内容为感谢该导师对于你的教导和对于该论文写作方法、写作技巧、写作内容等的影响和指导。

（2）对在论文写作过程中对你提供过帮助的所有老师进行感谢，大致内容为感谢老师们对本论文写作提供的帮助及指导，对论文的完善提出的自己的建议或意见。

（3）对在论文写作过程中帮助过你的师长、同学进行感谢，感谢他们帮助你查资料、提供相关资料等。

（4）感谢你的家人及朋友，简单表达他们对于你在精神、生活以及学习方面的鼓励和支持。

# 四　附录

## （一）用例图

用例图主要用来描述客户的需求，即用户希望系统具备的完成一定功能的动作，通俗地理解用例就是软件的功能模块，所以是系统分析阶段的起点，设计人员根据客户的需求来创建和解释用例图，用来描述软件应具备哪些功能模块以及这些模块之间的调用关系，用例图包含了用例（Use Case）和参与者（Actor），用例之间用关联来连接以求把系统的整个结构和功能反映给非技术人员（通常是软件的用户），对应的是软件的结构和功能分解。

用例是从系统外部可见的行为，是系统为某一个或几个参与者（Actor）提供的一段完整的服务。从原则上来讲，用例之间都是独立、并列的，它们之间并不存在着包含从属关系。但是为了体现一些用例之间的业务关系，提高可维护性和一致性，用例之间可以抽象出包含(include)、扩展(extend)和泛化(generalization)几种关系。

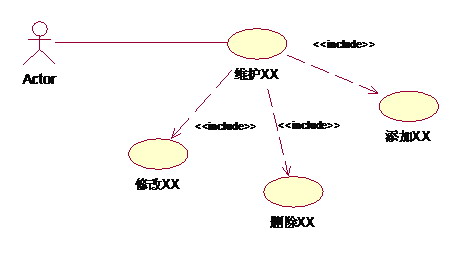
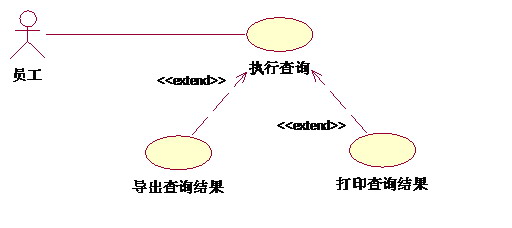
(1) 包含(include)关系：虽然每个用例的实例都是独立的，但是一个用例可以用其它的更简单的用例来描述。这有点像通过继承父类并增加附加描述来定义一个类。一个用例可以简单地包含其它用例具有的行为，并把它所包含的用例行为作为自身行为的一部分，这被称作包含关系。如图1所示。

图 1 用例包含关系示例图

(2) 扩展(extend)关系：一个用例也可以被定义为基础用例的增量扩展，这被称作扩展关系，扩展关系是把新的行为插入到已有的用例中的方法。同一个基础用例的几个扩展用例可以在一起应用。基础用例的扩展增加了原有的语义，此时基础用例而不是扩展用例被作为例子使用。在UML中，扩展关系表示为虚线箭头加<<extend>>字样，箭头指向被扩展展的用例。如图2所示。

图 2 用例扩展关系示例图

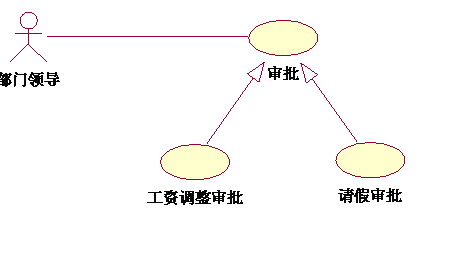
(3) 泛化(generalization)关系：子用例和父用例相似，但表现出更特别的行为；子用例将继承父用例的所有结构、行为和关系。子用例可以使用父用例的一段行为，也可以重载它。父用例通常是抽象的。在实际应用中很少使用泛化关系，子用例中的特殊行为都可以作为父用例中的备选流存在。例如，业务中可能存在许多需要部门领导审批的事情，但是领导审批的流程是很相似的，这时可以做成泛化关系表示。如图3所示。

图3 用例泛化关系示例图

## （二）业务流程图

业务流程图是一种描述系统内各单位、人员之间业务关系、作业顺序和管理信息流向的图表，利用它可以帮助分析人员找出业务流程中的不合理流向，它是物理模型。业务流程图主要是描述业务走向，例如病人，首先要去挂号，然后在到医生那里看病开药，然后再到药房领药，然后回家。业务流程图描述的是完整的业务流程，以业务处理过程为中心，一般没有数据的概念。

业务流程图的绘制是按照业务的实际处理步骤和过程进行的。业务流程图是一种系统分析人员都懂的共同语言，用来描述系统组织结构、业务流程。

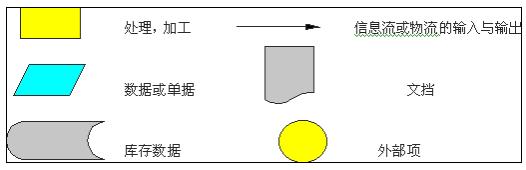
[](http://wiki.mbalib.com/wiki/Image:ä¸å¡æµç¨å¾çåºæ¬ç¬¦å·åå«ä¹è¯´æ.jpg)(1) 业务流程图的基本符号及含义如图4所示。

图 4 业务流程图的基本符号和含义说明

(2) 画业务流程图的步骤与例子

① 现行系统业务流程总结

在画业务流程图之前，要对现行系统进行详细调查，并写出现行系统业务流程总结。例如，开发人员在系统调查阶段了解到某[企业](http://wiki.mbalib.com/wiki/%E4%BC%81%E4%B8%9A" \o "企业)的[会计核算形式](http://wiki.mbalib.com/wiki/%E4%BC%9A%E8%AE%A1%E6%A0%B8%E7%AE%97%E5%BD%A2%E5%BC%8F" \o "会计核算形式)是[科目汇总表](http://wiki.mbalib.com/wiki/%E7%A7%91%E7%9B%AE%E6%B1%87%E6%80%BB%E8%A1%A8" \o "科目汇总表)的核算形式，其帐务处理业务流程如下：

* 根据审核无误的[原始凭证汇总表](http://wiki.mbalib.com/wiki/%E5%8E%9F%E5%A7%8B%E5%87%AD%E8%AF%81%E6%B1%87%E6%80%BB%E8%A1%A8" \o "原始凭证汇总表)编制[记帐凭证](http://wiki.mbalib.com/wiki/%E8%AE%B0%E5%B8%90%E5%87%AD%E8%AF%81" \o "记帐凭证)，包括[现金](http://wiki.mbalib.com/wiki/%E7%8E%B0%E9%87%91" \o "现金)收付、[银行](http://wiki.mbalib.com/wiki/%E9%93%B6%E8%A1%8C" \o "银行)收付、转帐凭证。
* 根据现金收付款凭证登记现金[日记帐](http://wiki.mbalib.com/wiki/%E6%97%A5%E8%AE%B0%E5%B8%90" \o "日记帐)。
* 根据银行收付款凭证登记[银行存款](http://wiki.mbalib.com/wiki/%E9%93%B6%E8%A1%8C%E5%AD%98%E6%AC%BE" \o "银行存款)日记帐。
* 根据银行送来的对帐单对银行存款日记帐核对。
* 根据记帐凭证及所付[原始凭证](http://wiki.mbalib.com/wiki/%E5%8E%9F%E5%A7%8B%E5%87%AD%E8%AF%81" \o "原始凭证)登记有关[明细帐](http://wiki.mbalib.com/wiki/%E6%98%8E%E7%BB%86%E5%B8%90" \o "明细帐)。
* 根据记帐凭证，按相同的借贷方汇总出[科目汇总表](http://wiki.mbalib.com/wiki/%E7%A7%91%E7%9B%AE%E6%B1%87%E6%80%BB%E8%A1%A8" \o "科目汇总表)。
* 根据科目汇总表登记汇[总分类帐](http://wiki.mbalib.com/wiki/%E6%80%BB%E5%88%86%E7%B1%BB%E5%B8%90" \o "总分类帐)。
* 将明细帐科目余额与财产物资实用数核对。
* 把总分类帐余额与有关明细帐余额核对。
* 根据总帐、明细帐余额编制各种[会计报表](http://wiki.mbalib.com/wiki/%E4%BC%9A%E8%AE%A1%E6%8A%A5%E8%A1%A8" \o "会计报表)。

② 业务流程图的绘制

根据上述业务流程可以绘制出该企业帐务处理业务流程图，如图3.5所示。

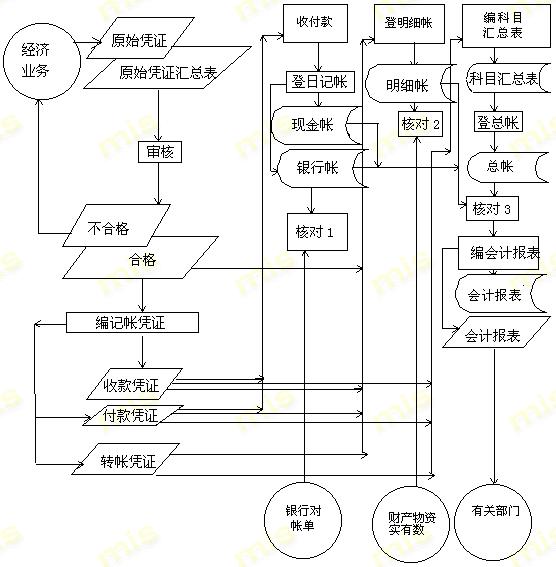
(3) 业务流程图的特点

① 图的形式是按业务[部门划分](http://wiki.mbalib.com/wiki/%E9%83%A8%E9%97%A8%E5%88%92%E5%88%86" \o "部门划分)的横式图。

② 图描述的主体是[票据](http://wiki.mbalib.com/wiki/%E7%A5%A8%E6%8D%AE" \o "票据)、帐单的业务处理。

③ 票据、帐单流动路线与实际业务处理过程一一对应。

④ 图中票据、帐单是有“ 生”、“ 死”的，即用它的一次生命周期来表示出一笔业务的处理情况。

图 5 帐务处理现行系统业务流程图

(4) 业务流程图的作用

① 制做[流程图](http://wiki.mbalib.com/wiki/%E6%B5%81%E7%A8%8B%E5%9B%BE" \o "流程图)的过程是全面了解业务处理的过程，是进行系统分析的依据。

② 它是系统分析员、管理人员、业务操作人员相互交流思想的工具。

③ 系统分析员可直接在业务流程图上拟出可以实现计算机处理的部分。

④ 用它可分析出业务流程的合理性。

## （三）活动图

活动图本质上是一种流程图，每个“活动”可以是某个具体的“事务”，比如审核单据等。用不太严谨的表述来说，活动图是一种粒度比较粗的事件流程图，多在需求阶段使用。

**(1) 基本活动图**

一个活动图可能包括以下元素：

① 活动状态：表示在工作流程中执行某个活动或步骤。

② 转移：表示各种活动状态的先后顺序。

③ 决策与警戒条件：显示业务用例的工作流程中的备选线程。

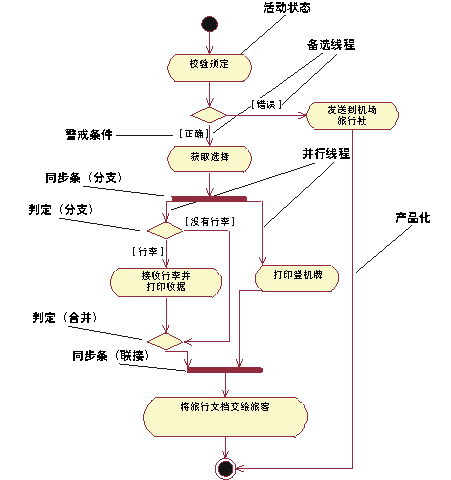
④ 同步示意条：用于显示平行分支流，能够显示业务用例的工作流程中的并行线程。如图6 所示。

图 6 “机场登记”业务用例模型中“个人登记”业务用例的活动图

**(2)** **条件线程**

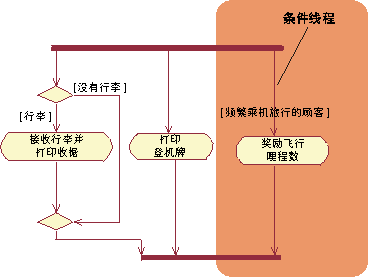
警戒条件用于说明一组并行线程中的某个线程是有条件的。例如，在上面的“个人登记”示例中，进行登记的乘客可能是频繁乘机旅行的顾客。在此情况下，您需要给他奖励一些飞行里程数。如图7所示。

图 7 “机场登记”业务用例模型中“个人登记”业务用例的活动图

(3) 嵌套活动图

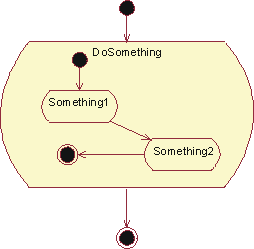
一个活动状态可能要引用另一个活动图，因为后者显示了前者的内部结构，称为嵌套活动图。可以显示活动状态中的子图或是让活动状态引用另一个图。如图8所示。

图 8 活动状态中嵌套的活动图

(4) 使用泳道

可以使用垂直实线将活动图划分为**泳道**。每条泳道代表整个工作流程的某个部分的职责，该职责由组织的某个部门来执行。泳道最终可以由组织单元或者业务对象模型中的一组类来实施。泳道之间的排序并不会影响语义。每个活动状态都指派了一条泳道，而转移则可能跨越数条泳道。如图9所示。

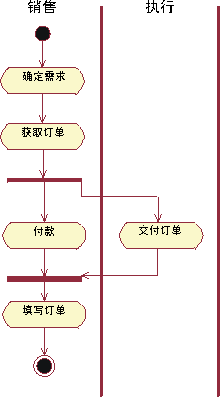


图 9 泳道活动图

## （四）数据流图

系统分析阶段必须进行全面准确的收集、整理、分析收集的数据及其流程。

数据流图(Data Flow Diagram，简称DFD)是描述系统逻辑模型的工具之一——只反映信息在系统中流动和处理情况的图。它能精确地在逻辑上描述系统的功能、输入、输出和数据存贮等，而摆脱了其物理内容。

(1) 数据收集

从客户的现行组织机构、现行系统的业务流程、现行的决策方式、各种报表、报告、图示等各种渠道进行数据收集。通过查阅档案，面谈调查，发调查表，测定，采样等多种方法进行数据收集。

(2) 数据分析

收集上来的数据是“原材料”，其中有些数据不能用作系统设计的依据，要把这些原材料加工成系统设计可用的资料，就必须做数据的分析工作。

(3) 数据流图（DFD）

将收集、整理、分析的数据用数据流图的形式进行描述。

① 数据流图的基本符号

数据流图由四种基本符号组成，如图10所示。

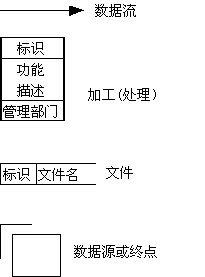


图 10 数据流图的基本符号

② 简单数据流图举例

图11是一个简单的数据流图，它表示数据X从源S流出，经P加工转换成Y，接着经P加工转换为Z，在加工过程中从F中读取数据。

|  |
| --- |
| 逻辑模型的工具－数据流图DFD |

图11 简单数据流图举例

③ DFD的画法

对于不同的问题，数据流图可以有不同的画法。一般情况下，应该遵守“由外向里”的原则。即先确定系统的边界或范围，再考虑系统的内部，先画加工的输入和输出，再画加工内部。具体实行时可按下述步骤进行：

* 识别系统的输入和输出，画出顶层图。
* 画系统内部的数据流、加工与文件，画出一级细化图。
* 加工的进一步分解，画出二级细化图。

其它注意事项：

* 数据流的命名，名字要确切，能反映整体。
* 从左至右画数据流图。各种符号布置要合理，分布均匀，尽量避免交叉线。
* 先考虑稳定态，后考虑瞬间态。如系统启动后在正常工作状态，稍后再考虑系统的启动和终止状态。

(4) 数据字典(Data Dictionary,DD)

在数据流图的基础上，还需对其中的每个数据流、文件和数据项加以定义， 我们把这些定义所组成的集合称为数据字典。数据流图是系统的大框架，而数据字典以及加工说明则是对数据流图中每个成分的精确描述。它们有着密切的联系，必须结合使用。

## （五）功能模块图

所谓功能模块图就是将系统的功能进行分解，按功能从属关系画成的图表。管理信息系统的各子系统可以看作是系统目标下层的功能，对其中每项功能还可以继续分解为第三层、第四层……甚至更多的功能。图中的每一个框都称为一个功能模块。功能模块可以根据具体情况分的大一点或小一点，分解得最小功能模块可以是一个程序中的每个处理过程，而较大的功能模块则可能是完成某一个任务的一组程序。

画出功能模块图是系统设计过程中的一个重要环节，它将显示出工程所要实现的各种功能，并分类，然后软件工程师根据功能模块图来具体实现这些功能（通过代码）

对于系统概要设计可以画出总体功能模块图如图12所示。



图 12 总体功能模块图

**注意：**模块图上层与下一层只有包含关系没有流程关系，**所以要使用无箭头的线条**。

在详细设计中对其中的某一模块再进行细化，如图13所示。



图 13 系统设置功能模块图

如果模块细化不多，也可以总体画出，如图14所示。

图 14 总体功能模块图



## （六）程序流程图

在系统实现章节，可以通过程序流程图来说明某个function的实现，如图15所示。

图 15 流程图示例图



**注意：**程序流程图要规范，有“开始”，有“结束”，分支要使用“菱形框”，要有分支条件，跳转箭头要指向流程线，而不能指向“执行框”。