

基于 GA-ANN-GA 的
模拟星际航行控制系统
概要设计说明书

目录

1 引言.....	3
1.1 编写目的.....	3
1.2 背景.....	3
2 总体设计.....	3
2.1 需求规定.....	3
2.1.1 系统功能要求.....	3
2.1.2 系统性能要求.....	4
2.2 运行环境.....	4
2.3 基本设计概念和处理流程.....	4
2.3.1 总体设计.....	4
2.3.2 启发式遗传算法简要设计.....	5
2.4 软件结构.....	7
2.5 功能需求与程序的关系.....	9
2.6 人工处理过程.....	9
3 接口设计.....	9
4 运行设计.....	9

1 引言

1.1 编写目的

编写此设计说明书，首先是为了让用户能更好的了解本系统的整体结构。其次，为了指导开发，让每一名开发者保持一致、明确的思路并保存开发经验。

更详细的说明，包括问题描述，主要的实验及其结论，不足、改进与扩展等内容请参阅《基于 GA-ANN-GA 系统的启发式遗传算法用于模拟星际航行控制》的研究论文，在此概要说明书不一一赘说。

1.2 背景

从星际航行的控制问题出发，通过对遗传算法（GA）与传统算法的分析、比较，提出运用遗传算法解决这一复杂的动作控制问题，并实践证明其可行性。在实现遗传算法的过程中，创造性地提出了新的编码方案，新的变异算子，摒弃了累赘的交叉算子。并在生物科学中“获得性遗传”学说的启发下提出并且实现一种新的启发式遗传算法模式，进一步提炼成具有普遍意义的“GA-ANN-GA”混合智能系统模型，并通过比较实验验证了该模型的先进性和可靠性。同时在软件开发过程中融入了 MVC（Model-View-Controller）设计模式和测试驱动开发（TDD）等先进的软件开发理念，从而保证了代码的健壮性和可移植性，力求使得具有普遍意义的研究成果便于移植与扩展。

2 总体设计

2.1 需求规定

2.1.1 系统功能要求

本系统需要实现如下所述功能：

- 允许用户手工绘制太空地图。
- 提供足够的参数设置来配置物理引擎，足够描述太空航行大体需要考虑的物理环境情况。
- 可以允许用户手动控制飞船航行，感受太空航行的危险性。
- 提供遗传算法，神经网络和启发式遗传算法三个模型的训练和结果显示。
- 提供遗传算法和神经网络的各种参数的设置界面。

- 支持用户编写自动控制策略，提供编写接口。
- 形象具体演示训练过程和训练后的结果。

2.1.2 系统性能要求

系统有如下性能要求：

- 算法完成训练时间低于 30 分钟。
- 算法结果演示必须准确反映算法训练的真实情况。

2.2 运行环境

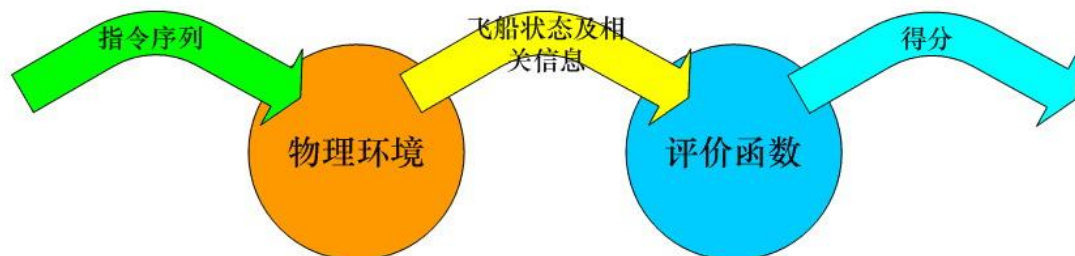
操作系统：Microsoft Windows NT\2000\XP

系统配置需要：.net framework 2.0 和 directX 9.0

硬件基本配置：PIII500 CPU 256M 内存和 3D 图形显示卡或更高配置

2.3 基本设计概念和处理流程

2.3.1 总体设计



流程图

本系统是通过每一个离散的時刻生成指令序列（各个方向的喷气加速的操作），然后物理引擎会综合飞行器当前速度矢量，所受合力矢量，与及当前位置，按照经典力学法则计

算出飞行器经过一个单位时间（离散时间的单位）段飞行器的新状态，包括飞行器新速度矢量，和新位置，并进行碰撞检测。最后采用一个评价函数（在遗传算法里面是适应度函数）对当前状态评价，作为该指令规则的评分，通过各种算法可以修正指令产生的模式。不断重复这一过程，达到训练飞行的目的。

创新点

由于本次研究涉及较前缘的技术与设想，没有现成的研究成果以供参考，所以从算法到编码均在一些基本原则的指导下原创出来的。比较显著的创新点有下面几点：

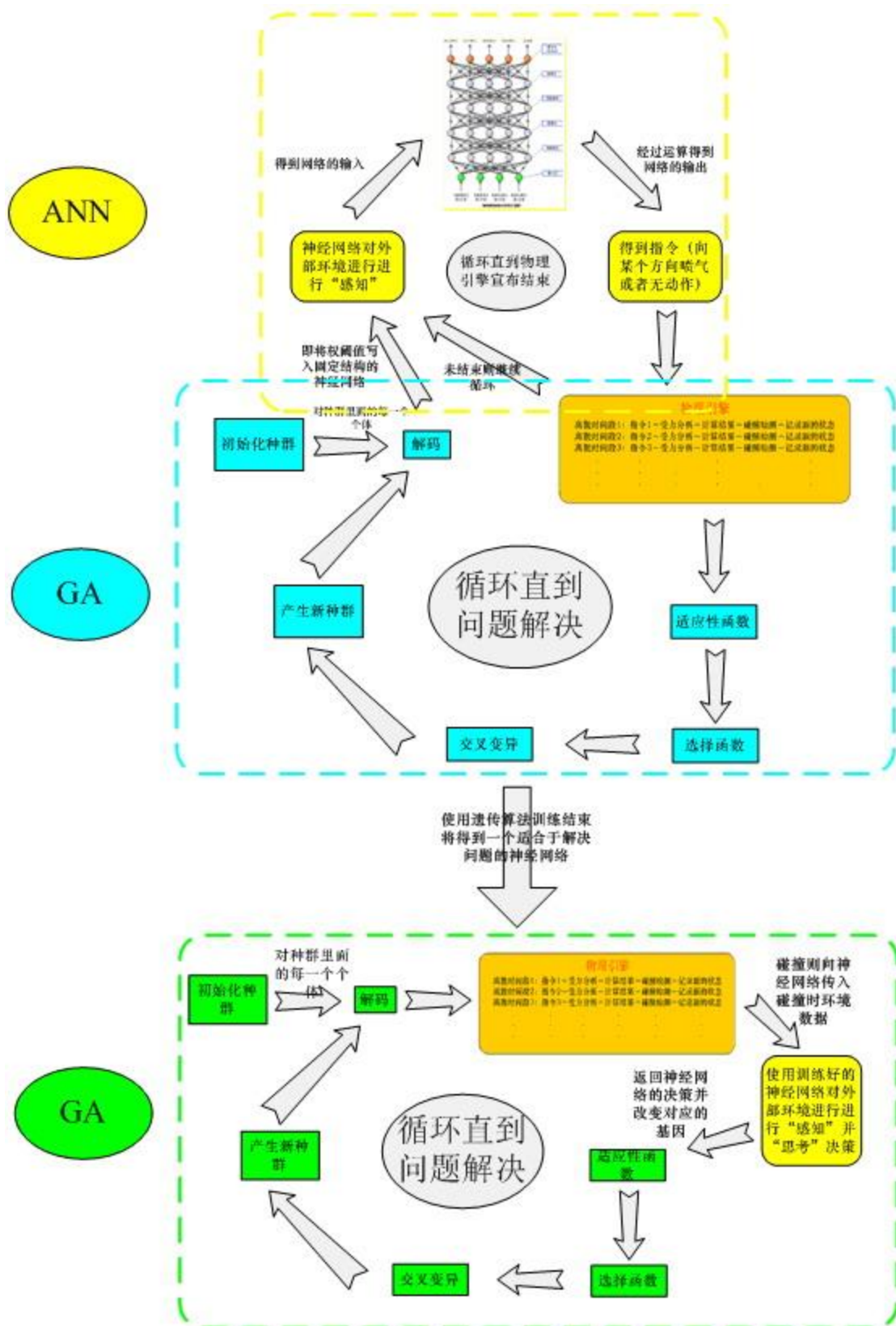
- 1) 提出一种新型的混合编码方式：**命令—时间**编码方式，并验证其可行性。
- 2) 提出一种新型的变异算子：**插入式变异**，并通过实验验证其先进性。
- 3) 从生物和仿真的角度论证交叉算子对于解决这类问题是**累赘**的。
- 4) 受到生物学中“**获得性遗传**”学说的启发，设计并实现一种基因编码能在“一生中”发生基因突变的新型启发式遗传算法。并由此提炼出具有普遍意义的**混合智能系统模型—GA—ANN—GA 模型**。

2.3.2 启发式遗传算法流程将要设计说明

启发式遗传算法训练的流程如下：

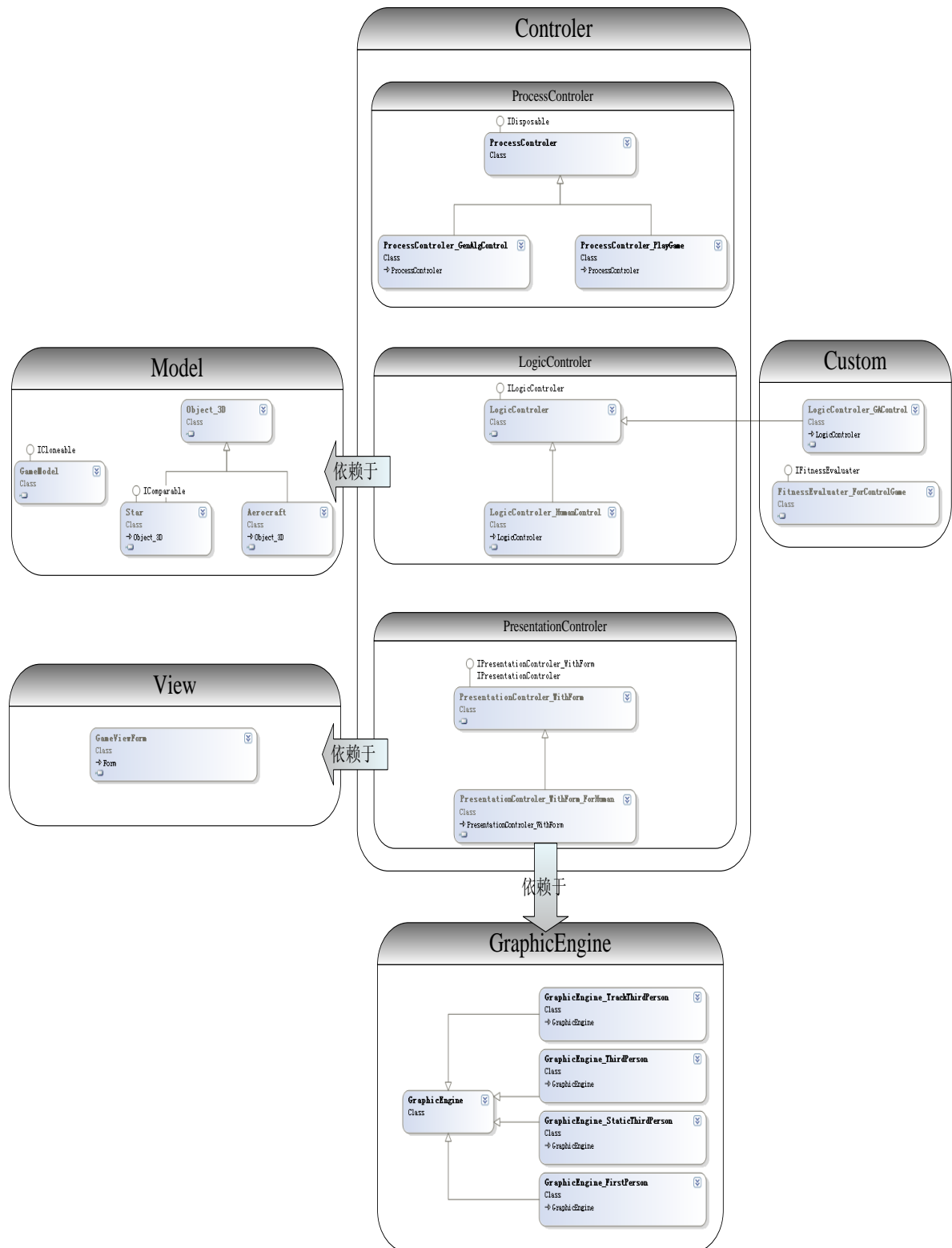
1. 建立一个随机权值的 6 层包含伪隐层的神经网络。（对伪隐层的定义，参考 **GA—ANN—GA 模型** 论文）
2. 把外部的环境相关变量输入神经网络，并把神经网络的输出向量（即指令序列）放到物理引擎模拟。
3. 模拟后的结果即碰撞的位置输入遗传算法的适应度函数并得到适应度评价。
4. 对种群（染色体为神经网络权值的编码）进行交叉变异。
5. 新一代的种群对应的神经网络再次接受环境变量并输出指令序列，重复上述过程对神经网络的权值进行训练。

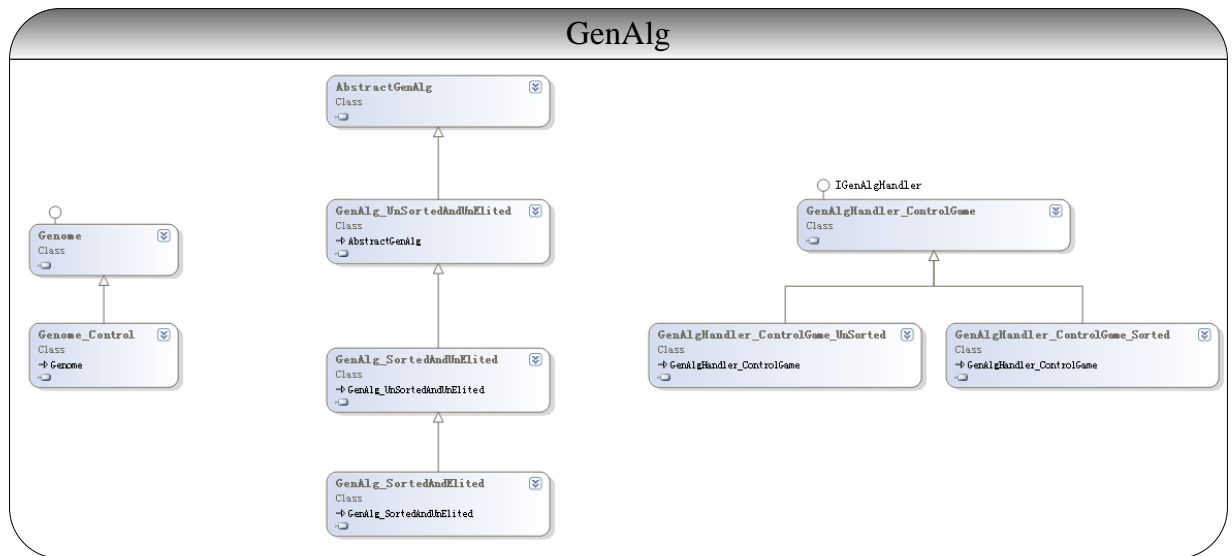
（具体遗传编码的方式和神经网络的结构等内容参阅《基于 GA—ANN—GA 系统的启发式遗传算法用于模拟星际航行控制》的研究论文）



顺序图

2.4 软件结构





系统类关系图

以上类关系图来自于软件的 3D 版本，由于 3D 版本开发晚于 2D 版本，而且选用了完全面向对象语言 C# 进行开发，所以在类的层次架构设计方面更加成熟，更加稳健。

相对于 2D 版本里面使用的经典的 MVC 设计模式，在 3D 版本里面我们把 MVC 模式改造成更适合此软件具体情况的设计架构。也更能体现软件设计的一些基本原则：单一职责原则（SRP），开放封闭原则（OCP），依赖倒置原则（DIP），接口隔离原则等（ISP）。

首先，我们把 Controller 类分成了功能更细，职责更加单一的三个类：ProcessController，LogicController，PresentationController。他们的职责分别是：

ProcessController 对整个软件的功能流程进行控制，包括对线程的控制，游戏流程的控制，利用 LogicController 对算法流程进行控制，利用 PresentationController 对界面显示进行控制。

LogicController 的作用是沟通游戏模型和算法引擎或者人的控制逻辑，根据当前的游戏模型的具体情况，对游戏过程进行逻辑上的控制（比如决定什么时候进行向上加速，什么时候进行向下加速等）。

PresentationController 是游戏表现层的控制器。沟通了游戏视窗和底层的 DirectX API。从而实现对游戏过程的显示。PresentationController 里面也包含对显示逻辑的空实现。从而能够使游戏可以在不进行复杂的 3D 场景搭建和界面运算的情况下高速运行。这主要被遗传算法进行后台仿真模拟的时候调用。

而 Model 模块和 View 模块基本没有进行过多的改动，它们的职责已经非常单一，类的内聚度适当。

在整个系统设计的过程中我们还考虑了这样一个功能：让用户自己可以按照自己的想法来实现对飞行器的控制。在软件里面提供了用人手（HumanControl），遗传算法（GA），人工神经网络（ANN）还有启发式遗传算法（HGA）对游戏进行控制。（软件的 0.5 版本还没有实现 ANN 和 HGA）但是解决控制问题的方案有无数种的，我们在软件里面不能一一列举，比如用模糊逻辑，用有穷状态机，用专家系统，甚至用 if-then 语句都可能对飞船进行合理的控制。我们希望使用者可以借助这个平台实现自己的想法。所以我们软件里面留出一个 custom（自定义）模块，使用者只要按照自己的方式来重写这个 dll 文件，就可以实现自己的控制方法。在这个 dll 里面使用者可以调用软件其它模块的一些功能函数，或者借助参

数来得到游戏 Model 的信息，借助这些信息，可以用自己的办法来作出相应的反映，从而实现该 dll 为软件另外一些提供调用的接口。

2.5 功能需求与程序的关系

	GenAlg	GraphicEngine	Controller	Model	View
绘制太空地图		✓			✓
物理引擎		✓		✓	
控制飞船航行	✓		✓	✓	
训练飞行模型	✓			✓	
模型参数设置				✓	
显示图形结果		✓			✓

主要类型与功能需求关系表

2.6 人工处理过程

物理引擎模块，遗传算法模块和神经网络模块的各个参数设置需要有一定的专业知识，不同参数设置对实验的影响的更详细的说明，包括问题描述，主要的实验及其结论，不足、改进与扩展等内容请参阅《基于 GA-ANN-GA 系统的启发式遗传算法用于模拟星际航行控制的研究论文》

3 接口设计

请参阅《基于 GA-ANN-GA 系统的启发式遗传算法用于模拟星际航行控制》的研究论文中的 5-6 节。

4 运行设计

请参阅《基于 GA-ANN-GA 系统的启发式遗传算法用于模拟星际航行控制》的研究论文中的 3-6 节。