

多无人机编队场景建模实验

实验内容

实验背景：从各国军用无人机的协同侦察与作战，到民用无人机的立体测绘、组队运输与救灾，再到南大百廿校庆中闪耀璀璨的无人机秀，多架无人机(Unmanned Aerial Vehicle, **UAV**)协同编队来完成任务的例子数不胜数。无人机协同编队，涉及将多架具有自主控制能力的无人机按照指定的三维队形进行排列，实时保持或调整队形、协同完成各项任务。

实验目标：分别使用**Scenario**和**SpaceEx**这两个建模工具，对无人机协同编队的任务场景（包含：成环集结、多队形变换、速度控制等）进行形式化的系统建模，建模语言为混成自动机及其组合。

实验提供材料：

- **Scenario.ova:** Ubuntu Linux虚拟机镜像，已配置环境、可直接运行工具Scenario和SpaceEx（**强烈建议使用VirtualBox**打开该镜像；使用VMware可强行打开，但可能出现不可预料的意外）
 - **用户名:** njucs, **密码:** 123;
 - **~/Scenario:** 在该目录下执行./run.sh 运行工具Scenario, ~/Scenario/materials中包含Scenario可接受的UAV模型文件(UAV.xml)、多UAV组合的模板文件(yyyTemplate.cfg);
 - **~/SpaceEx:** 在该目录下执行./run.sh 运行工具SpaceEx, ~/SpaceEx/materials中是SpaceEx可接受的模型文件(uav_spaceex.xml), 包含UAV模型及用于控制组合的模板模型。
- **实验手册.pdf(本手册):** 除本节实验内容的说明以外，还包含
 - 知识回顾: 以UAV的混成自动机模型为例，回顾混成自动机及组合混成自动机的定义;
 - 实验场景: 本实验需要你进行建模的，多无人机协同编队的具体任务场景。
- **Scenario用户手册.pdf**
- **SpaceEx参考说明.pdf**
- **tutorial.mp4**

实验提交方式与材料：

- 提交方式: <https://box.nju.edu.cn/u/d/d819ed75f50948919633/>
- 提交内容: 学号.tar.gz, 内含:
 1. **uav_学号.tar.gz:** Scenario建模文件 (以"uav_学号"为名创建project, 完成建模后打包项目 tar -zcvf uav_学号.tar.gz uav_学号/);
 2. **uav_学号.xml:** SpaceEx建模文件(在uav_spaceex.xml的基础上修改完成后重命名即可);
 3. **uav_学号.log:** 关闭SpaceEx前, 手动点击Log、Ctrl+A全选、Ctrl+C复制后, 粘贴进你自己创建的uav_学号.log.

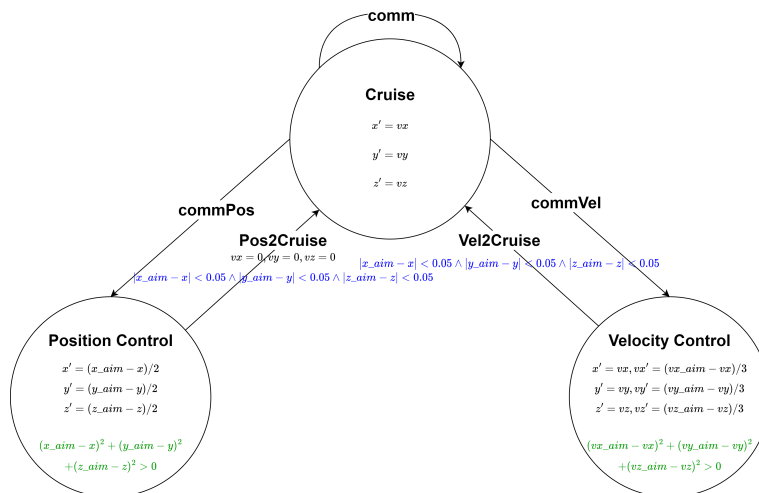
实验评分依据与说明 (!!!实验前必看):

- 实验评分依据各场景建模的完整性与正确性 (log仅用于证明你是使用工具可视化界面逐步建模的, 而非复制粘贴他人文件。故, 使用工具Scenario构建project过程中, 会在你的project文件夹内自动生成加密的log文件, **切勿编辑删改; 每次关闭SpaceEx前务必复制记录log; 最终务必确保提交了log**);
- 如对需建模场景的理解或对工具的使用有任何问题, 都欢迎在群里咨询或咨询QQ:594421107;
- 两个工具(SpaceEx v0.9.5和Scenario v0.1)都是以教学/科研为导向的免费实验版本, 尤其是在工具Scenario 使用过程遇到的任何bug都欢迎报错, 任何本实验相关的idea都**十分欢迎与我们沟通**, [bug/idea提交链接](#).

知识回顾：混成自动机及其组合

混成自动机 (Hybrid Automata)

混成自动机在状态机的基础上进行了实时连续变量扩展，常被用于对同时包含离散和连续行为的复杂系统进行建模。下图是一个对无人机行为（实验简化版本）进行建模描述的混成自动机。我们将以该UAV的模型为例，回顾、解释混成自动机的定义。



混成自动机可被定义为多元组 $H = (X, V, \Sigma, E, \alpha, \beta)$ ，其中：

- X 是系统**变量 (Variable)** 的集合，上图中系统变量包括： x, y, z (UAV当前位置)， vx, vy, vz (当前速度)， $x_{aim}, y_{aim}, z_{aim}$ (目标位置)， $vx_{aim}, vy_{aim}, vz_{aim}$ (目标速度)；
- V 是系统**位置节点 (Location)** 的集合，上图每一个**圆圈**表示UAV的一个位置节点，包括：Cruise巡航模式节点、Position Control位置控制模式节点、Velocity Control速度控制模式节点；系统处于不同节点时行为不同：
 - 标注函数 α 为每个节点标注**流条件 (Flow)**，系统在该节点上运行时，变量必须按照流条件中的微分方程随时间变化（若变量的变化率为0，可省略）。如：在Velocity Control节点时， x 随时间的变化率为 vx ， vx 随时间的变化率为 $(vx_{aim} - vx)/3$ ；因此，在该节点时UAV的速度 vx 会不断逼近它的目标速度 vx_{aim} 。
 - 标注函数 β 为每个节点标注**不变式 (Invariant)**，系统在该节点上运行时须满足该约束，图上的不变式用绿色写在圆圈内。如：在Velocity Control节点时必须满足 $(vx_{aim} - vx)^2 + (vy_{aim} - vy)^2 + (vz_{aim} - vz)^2 > 0$ ，即：一旦UAV的当前速度已经达到了目标速度，则系统必须离开当前节点；
- E 是**离散跳转 (Transition)** 的集合，上图每一条**有向边**都是一个离散跳转，每个跳转 $e \in E$ 形如 $(v, \sigma, \gamma, \phi, v')$ ，其中：
 - $\sigma \in \Sigma$ 是跳转的**事件名 (Label)**， Σ 是跳转事件名的集合，上图中 $\Sigma = \{commPos, commVel, comm, Pos2Cruise, Vel2Cruise\}$ ；
 - $v, v' \in V$ ，分别代表跳转的出发、到达节点。如：触发Pos2Cruise跳转时，UAV从Position Control节点跳转至Cruise节点；
 - γ 是**跳转卫式 (Guard)**，即系统跳转时须满足的约束，图上蓝色写出。如：Pos2Cruise跳转发生时须满足 $|x_{aim} - x| < 0.05 \wedge |y_{aim} - y| < 0.05 \wedge |z_{aim} - z| < 0.05$ ，即：UAV的当前位置与目标位置在xyz方向上的距离均小于0.05；
 - ϕ 是**重置动作 (Reset/Assignment)**，跳转后变量值据此被瞬间重置。如：Pos2Cruise跳转发生时 vx, vy, vz 都被重置为零。

上图的混成自动机描述的UAV的行为：系统可以处于巡航模式匀速前进；也可以处于位置控制模式向目标位置移动，在到达（或逼近）目标位置时，将速度重置为零、跳回巡航模式悬停；也可以处于速度控制模式向目标速度更新，在到达（或逼近）目标速度后，跳回巡航模式匀速前进。该模型将贯穿本实验，值得注意的是，我们并未给出commPos、commVel等离散跳转的详细信息（如重置动作等），在实验过程中你可能需要根据具体场景需求对它们进行设置。

多个混成自动机的组合

对于由多个agent组件统构成的复杂系统，我们往往先分别对每个agent进行建模，然后通过定义agent间的通信来完成对整个系统的建模。每个agent组件都可由一个混成自动机建模描述，混成自动机之间又可通过共享事件、共享变量进行通信组合，故整个系统则可由多个混成自动机的组合描述。简单起见，我们此处不再形式化地给出组合混成自动机的定义，而是以无人机队列为例，分别展示在Scenario和SpaceX中我们应如何设计混成自动机的组合。

Scenario中的混成自动机的组合

1. 通过Add Model添加我们提供的UAV混成自动机模型文件，根据通信能力：
 - 模型中的变量被分为output、input、local三种类型：
 - output变量是组合通信时可输出给其他自动机的变量，包括 x, y, z, vx, vy, vz ;
 - input变量是组合通信时可以接受外界输入的变量，UAV模型中提供了12个变量作为接收端口，包括 $inp11/12/13 \dots \dots inp41/42/43$;
 - local变量是系统的本地变量，既不能在通信时输出给其他外界、也不能在通信时被外界修改，包括 $x/y/z_aim, vx/vy/vz_aim$;
 - 模型中的跳转被分成了comm和local两种类型：
 - comm跳转可承担组合通信的作用，默认情况下不被激活（deactivated），无法被触发；在具体任务中，comm跳转可以作为input/output跳转被激活（activated），来承担输入/输出通信内容的作用；
 - local跳转无需激活，但不可与其他自动机的跳转一同组合通信；
2. 通过Create Agent，基于上一步添加的UAV model，来为队列中的每一个UAV agent建模；
3. 当需要描述某一场景下多个UAV agent之间的组合时，首先Create Task来定义一个用于描述该场景的原子任务；然后选定涉及的agent并定义该任务开始/结束时各agent应处于的位置节点；最后通过添加一组或多组IO-SYN来为多个agent添加用于通信同步的共享事件，一组IO-SYN需：
 - 定义共享事件：为涉及通信的每个agent选定一条可通信的comm跳转，将其作为input或output跳转被激活；我们称这组被激活的跳转是共享事件，在当前任务下只能同时被触发；
 - 定义通信关系communication（可选）：对于每个包含input跳转的agent，它的input变量可设置输入值，可以是任何包含output跳转的agent的输出变量；
 - 定义附加的重置动作reset（可选）：对于每条被激活的跳转，都可设置额外的重置动作用于处理通信信息；
4. 当需要描述多个场景任务时，根据任务的层次结构，从最顶层任务开始，自顶向下依次定义即可。

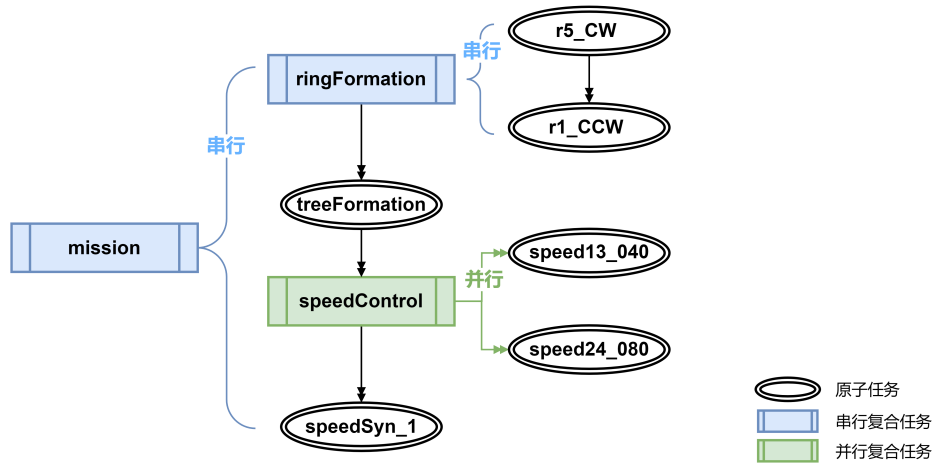
SpaceX中的混成自动机的组合

1. 通过open a model file打开我们提供的模型文件，内含3个UAV混成自动机模型：其中，UAVtemplate与上图的自动机一致，另两个UAVxxxtemplate是基于UAVtemplate修改得到的，适用于队形控制相关的任务场景。选择UAVtemplate，点击Duplicate current selection复制后，你也可根据其他场景需要，修改并保存UAV模型；
2. 当需要描述某一场景下多个UAV agent之间的组合时，点击Add a new network component来定义一个用于描述该场景的网络组件；然后右键Add Bind来基于已有的UAV模型来依次建模每一个UAV agent；最后可通过Add Bind添加一个controller用于处理通信内容；

- 设置共享事件：将多个UAV agent需同步触发的跳转都link to同一个label param;
- 定义通信关系、处理通信内容：
 - 方法一：将UAV agent中的输出变量以及另一UAV agent对应的需输入变量link to相同的variable param; (仅适用于无需处理通信内容的时候)
 - 方法二：将UAV agent中的输出变量以及controller对应的输入变量link to相同的variable param, 利用controller中的不变式处理通信内容, 将controller中的输出变量与另一UAV agent对应的需输入变量link to相同的variable param;
- 3. 当需要描述多个场景任务时, 先用network component定义每一个场景(原子任务); 然后根据原子任务的先后以及串并行关系, 来设计一个控制任务执行顺序的controller模型; 最后定义一个最终的network component, 通过Add Bind添加所有原子任务的network component以及任务顺序controller模型, 来描述整个任务场景。

实验场景：多无人机协同编队任务

编队场景仅涉及4个无人机(u1-u4)。需要你建模一个如下图所示, 叫做mission的无人机协同编队任务。



mission任务包含下面4个要求依次串行执行的子任务：

1. **ringFormation**: 采用**集中式**通信控制的就地成环任务, 包含下面2个要求依次串行执行的子任务：
 - **r5 CW**: 以5为半径**顺时针**(clockwise)成环, 对于u1-u4的任一架UAV, 通信获取其他所有UAV的位置并计算u1-u4的中心点, 以此中心点为圆心计算自身的目标位置(如 $u_i.x_{aim} = \sum_{k=1}^4 (u_k.x) / 4 + r * \sin(\frac{(i-1)*\pi}{2})$)。提示: u1-u4同时从巡航模式进入位置控制模式, 到达目标位置后自行回到巡航模式悬停。
 - **r1 CCW**: 以1为半径**逆时针**(counter-clockwise)成环, 通信控制方法同上;
2. **treeFormation**: 采用**长机-僚机**(leader-follower)通信控制的树形编队任务。u1向u2-4广播自己的位置后保持不动, u2-4据此计算并各自移至目标位置(XY方向的目标位置见下图、 z_{aim} 为u1当前高度)。提示: u2-4同时从巡航模式进入位置控制模式, 到达目标位置后自行回到巡航模式悬停。
3. **speedControl**: 包含下面2个要求同时并行执行的速度控制子任务：
 - **speed13 040**: 更新u1和u3的目标速度为 $vx_{aim} = 0, vy_{aim} = 4, vz_{aim} = 0$ 。提示: u1、u3同时从巡航模式进入速度控制模式调整速度, 完成后自行回到巡航模式。
 - **speed24 080**: 类似地, 控制u2和u4速度, $vx_{aim} = 0, vy_{aim} = 8, vz_{aim} = 0$ 。
4. **speedSyn 1**: 采用**长机-僚机**(leader-follower)通信控制的速度同步任务。u1向u2-4广播自己的速度(三维), u2-4据此更新目标速度、调整自身速度。提示: u2-4同时从巡航模式进入速度控制模式, 调整速度完成后自行回到巡航模式。

注：前三个原子任务的XY平面示意图如下。无论是Scenario还是SpaceEx，我们都为前三个原子任务提供了通信组合的模板。Scenario提供的是名为RingFormationTemplate和TreeFormationTemplate的用于创建原子任务的模板，他们分别定义了4个UAV成环任务和树形编队任务所需的IO-SYN，你可以在Create Task时选择从模板新建原子任务，并指定任务中agent的名字即可。SpaceEx提供的是处理组合通信的controller，包括名为ringController和treeController的用于组合的base component，他们分别定义了4个UAV成环任务和树形编队任务所需的通信内容处理。你可以从有模板的原子任务开始做起，熟悉了解流程后再构造其他任务。

