

1.1 复杂系统仿真基础

1.1.1 系统仿真及其分类

系统仿真是 20 世纪 40 年代末以来伴随着计算机技术的发展而逐步形成的一门新兴学科。仿真（Simulation）就是通过建立实际系统模型并利用所见模型对实际系统进行实验研究的过程。最初，仿真技术主要用于航空、航天、原子反应堆等价格昂贵、周期长、危险性大、实际系统试验难以实现的少数领域，后来逐步发展到电力、石油、化工、冶金、机械等一些主要工业部门，并进一步扩大到社会系统、经济系统、交通系统、生态系统等一些非工程系统领域。可以说，现代系统仿真技术和综合性仿真系统已经成为复杂系统特别是高技术产业不可缺少的分析、研究、设计、评价、决策和训练的重要手段，其应用范围在不断扩大，应用效益也日益显著。

系统仿真是建立在控制理论、相似理论、信息处理技术和计算机理论基础之上的，以计算机和其他专用物理效应设备为工具，利用系统模型对真实或假设的系统进行试验，并借助于专家的经验知识、统计数据和信息资料对实验结果进行分析研究，进而做出决策的一门综合的实验性学科。从广义而言，系统仿真的方法适用于任何领域，无论是工程系统（机械、化工、电力、电子等）还是非工程系统（交通、管理、经济、政治等）。

系统仿真根据模型不同，可以分为物理仿真、数学仿真和物理—数学仿真（半实物仿真）；根据所用计算机的类型，可以分为模拟仿真、数字仿真和混合仿真；根据系统的特性，可以分为连续系统仿真、离散时间系统（采样系统）仿真和离散事件系统仿真；根据仿真钟与实际时间的关系，可以分为实时仿真、欠实时仿真和超实时仿真等。

1.1.2 系统仿真的一般步骤

对于每一个成功的仿真研究项目，其应用都包含特定的步骤。不论仿真项目的类型和研究目的有何不同，仿真的基本过程是保持不变的，一般要进行如下9步。

(一) 问题的定义

一个模型不可能呈现被模拟的现实系统的所有方面，有时是因为费用太高。一个表现真实系统所有细节的模型，常常是非常差的模型，因为它会过于复杂和难以理解。因此，明智的做法是：先定义问题，再制定目标，再构建一个能够完全解决问题的模型。在问题定义阶段，对于假设要小心谨慎，不要做出错误的假设。例如，假设叉车等待时间较长，比假设没有足够的接收码头要好。仿真的原则是，定义问题的陈述越通用越好，要详细考虑引起问题的所有可能原因。

(二) 制定目标和定义系统性能测度

没有目标的仿真研究是毫无用途的。目标是仿真项目所有步骤的导向。系统的定义也是基于系统目标的。目标决定了应该做出怎样的假设、应该收集哪些信息和数据；模型的建立和确认要考虑到能否达到研究的目标。目标需要清楚、明确和切实可行。目标经常被描述成诸如这样的问题：“通过添置机器或延长工时，能够获得更多的利润吗？”在定义目标时，详细说明那些用来决定目标是否实现的性能测度是非常必要的。例如，产出率、工人利用率、平均排队时间、最大队列长度，都是最常见的系统性能测度。

最后，列出仿真结果的先决条件。如：必须通过利用现有设备来实现目标，或最高投资额要在一定限度内，或产品订货提前期不能延长等。

(三) 描述系统和列出假设

简单地说，仿真模型可以减少完成工作的时间。不论模型是一个物流系统、制造工厂还是服务机构，清楚明了地定义如下建模要素都是非常必要的：资源、流动项目（产品、顾客或信息）、路径、项目运输、流程控制、加工时间、资源故障时间、运输时间、排队时间，等等。

以一个制造系统仿真为例，它可以将现实系统资源分成四类：处理器、队列、运输和共享资源。流动项目的到达和预载的必要条件必须定义，如：到达时间、到达模式和该项目的类型等属性。在定义流动路径时，合并和转移需要详细描述。项目的转变包括属性变化、装配操作（项目合并）、拆卸操作（项目分离）。在系统中，通常需要控制项目的流动。如：一个项目只有在某种条件或某一时刻到来时才能移动。所有的处理时间都要被定义，并且要清楚表明哪些操作是机器自动完成，哪些操作是人工独立完成，哪些操作需要人机协同完成。资源可能有计划故障时间和意外故障时间。计划故障时间通常指午餐时间、中场休息和预防性维护等。意外故障时间是随机发生的故障所需的时间，包括失效平均间隔时间和维修平均间隔时间。

在这些工作完成之后，需要将现实系统进行模型描述，这远比模型描述向计算机模

型转化困难。现实向模型的转化意味着你已经对现实有了非常彻底的理解，并且能将其完美地描述出来。这一阶段，将此转换过程中所做的所有假设进行详细说明非常有必要。事实上，在整个仿真研究过程中，最好使所有假设列表保持在可获得状态，因为这个假设列表随着仿真的递进还要逐步增长。如果描述系统这一步做得非常好，建立计算机模型这一阶段将更加容易。

注意



获得足够的、能够体现特定仿真目的的系统一手材料是必要的，但是不需要获得与真实系统——对应的模型描述。

（四）列举可能的替代方案

在仿真研究中，确定模型早期运行的可替代方案是很重要的，它将影响着模型的建立。在初期阶段考虑到替代方案，模型可能被设计成易于转换为替换方案的系统。

（五）收集数据和信息

收集数据和信息，除了为模型参数输入数据外，还可以在验证模型阶段提供实际数据与模型的性能测度数据进行比较。数据可以通过历史记录、经验和计算得到。这些粗糙的数据将为模型输入参数提供基础，同时将有助于一些较精确参数数据的收集。

有些数据可能没有现成的记录，而通过测量来收集数据费时、费钱。除了在模型分析中，模型参数需要极为精确的输入数据的情况以外，同对系统的每个参数数据进行调查、测量的收集方式相比，采用估计方法来产生输入数据更为高效。估计值可以通过少数快速测量或者通过咨询熟悉系统的系统专家来得到。即使是使用较为粗糙的数据，根据最小值、最大值和最可能取值定义一个三角分布，要比仅仅采用平均值仿真效果好得多。有时采用估计值也能够满足仿真研究的目的。例如，仿真可能被简单地用来指导相关人员了解系统中特定的因果关系。在这种情况下，估计值就可以满足要求。

当需要可靠数据时，花费较多时间收集和统计大量数据，以定义出能够准确反映现实的概率分布函数，就是非常必要的，所需数据量的大小取决于变量的不确定程度。假如要获得随机停机时间的输入参数，就必须要在一个较长时间段内捕获足够多的数据。

（六）建立计算机模型

建立计算机模型，首先要构建小的测试模型来证明复杂部件的建模是否合适。一般建模过程是呈阶段性的，在进行下一阶段建模之前，验证本阶段的模型是否工作正常，并在建模过程中运行和调试每一阶段的模型。小模型有助于定义系统的重要部分，并可以引导为后续模型的详细化而进行的数据收集活动。我们有时会对同一现实系统构建多个计算机模型，每个模型的抽象程度都不相同。

（七）验证和确认模型

验证是确认模型的功能是否同设想的系统功能相符合，模型是否同我们想构建的模

型相吻合, 产品的处理时间、流向是否正确等。还包括更广泛的确认范围: 确认模型是否能够正确反映现实系统, 评估模型仿真结果的可信度有多大等。

(1) 验证

有很多技术可以用来验证模型。最重要的是在仿真低速运行时, 观看动画和仿真钟是否同步运行, 它可以发现物料流程及其处理时间方面的差异。

另一种验证技术是在模型运行过程中, 通过交互命令窗口、显示动态图表来询问资源和流动项目的属性和状态。

通过“步进”方式运行模型和动态查看轨迹文件可以帮助人们调试模型。运行仿真时, 通过输入多组仿真输入参数值, 来验证仿真结果是否合理, 也是一种很好的方法。在某些情况下, 对系统性能的一些简单测量可以通过手工或使用对比来获得。对模型中特定区域要素的使用率和产出率通常是非常容易计算出来的。

(2) 确认

模型确认可以建立模型的可信度。但是, 现在还没有哪一种确认技术可以对模型的结果进行 100% 的确定。我们永远不可能证明模型的行为就是现实的真实行为。如果我们能够做到这一步, 可能就不需要进行仿真研究的第一步(问题的定义)了。我们尽力去做的, 最多只能是保证模型的行为同现实不会相互抵触罢了。

通过确认, 试着判断模型的有效程度。假如一个模型在得到我们提供的相关正确数据之后, 其输出满足我们的目标, 那么它就是好的。模型只要在必要范围内有效就可以了, 而不需要尽可能地有效。在模型结果的正确性同获得这些结果所需要的费用之间总存在着权衡。

判断模型的有效性需要从如下几方面着手:

- 模型性能测度是否同真实系统性能测度匹配?
- 如果没有现实系统来对比, 可以将仿真结果同相近现实系统的仿真模型的相关运行结果作对比。
- 利用系统专家的经验来分析复杂系统特定部分模型的运行状况。
- 对每一主要任务, 在确认模型的输入和假设都是正确的、模型的性能测度都是可以测量的之前, 需要对模型各部分进行随机测试。
- 模型的行为是否同理论相一致? 确定结果的理论最大值和最小值, 然后验证模型结果是否落入两值之间。
- 为了了解模型在改变输入值后, 其输出性能测度的变化方向, 可以通过逐渐增大或减小其输入参数, 来验证模型的一致性。
- 是否有其他仿真模拟器实现了这个模型? 要是有的话那就再好不过了, 可以将已有模型的运行结果同现在设计的模型的运行结果进行对比。

(八) 运行

当系统具有随机性时, 就需要对实验做多次运行。因为, 随机输入导致随机输出。如果可能, 在第二步中应当计算出已经定义的每一性能测度的置信区间。

有些仿真软件特别提供了“优化(Optimizer)”模块来执行优化操作, 通过选择目标函数的最大化或最小化, 定义需要实验的许多决策变量、需要达到的条件变量、需要满足

的约束等，然后让优化模块负责搜索变量的可替换数字，最终得出决策变量集的最优化解决方案，和最大化或最小化的模型目标函数。“优化（Optimizer）”模块会设置一套优化方法，包括遗传算法、禁忌搜索、分散搜索和其他的混合方法，来得出模型的最优化配置方案。

在选择仿真运行长度时，考虑启动时间、资源失效可能间隔时间、处理时间或到达时间的或季节性差异，或其他需要系统运行足够长时间才能出现效果的系统特征变量，是非常重要的。

（九）输出分析

报表、图形和表格常常被用于进行输出结果分析。同时需要利用统计技术来分析不同方案的模拟结果。一旦通过分析结果并得出结论，要能够根据仿真的目标来解释这些结果，并提出实施或优化方案。使用结果和方案的矩阵图进行比较分析也是非常有帮助的。

需要注意的是，仿真研究不能简单机械地照搬以上9个环节，有些项目在获得系统的内在细节之后，可能要返回到先前的步骤中去做大量补充工作。同时，验证和确认需要贯穿于仿真工作的每一个环节当中。

1.1.3 复杂系统仿真及其作用意义

随着计算机性能的不断提高，建模与仿真技术愈发成熟，开始能够解决现实生活与生产中的复杂问题，复杂系统建模仿真逐渐成为一个热点领域。

自20世纪80年代中期，复杂性科学的概念首次被提出以来，得到了飞速发展，究其本质，复杂性科学可以说是非线性科学和系统科学的进一步演进与深化。如果说，系统科学是建立在系统的整体性、组织性、目的性研究基础上的，非线性科学是建立在系统的非线性、不确定性、随机性研究基础上的，那么复杂性科学则是聚焦对系统复杂性、智能性和适应性的研究。但是，对于“复杂系统”一词，始终没有形成统一的定义。一般认为，复杂系统区别于一般简单系统的本质特征在于它的复杂性，它是由多个因素构成的、要素之间具有复杂非线性关系的系统，可以理解为由大量单元或子系统非线性耦合在一起的空间组织或时空过程。自然界和人类社会广泛存在着由无数个体组合而成的无限多样性和复杂性的复杂系统，与复杂系统相关的理论包括非线性系统动力学、耗散结构论、协同理论、混沌理论、分形理论、复杂适应系统（Complexity Adaptive Systems, CAS）理论等。

虽然“复杂系统”至今没有明确的严格定义，从系统建模仿真的角度可以认为，复杂系统是由相当多具有智能性、自适应主体构成的大系统，系统中没有中央控制，内部存在着许多复杂性，并具有巨大变化性，从而决定了系统主体间及与环境间的复杂相互作用，使得复杂系统涌现出所有单独主体或部分主体不具有的整体行为特性。

有些专家将“涌现”视为复杂系统的最主要特征，系统中简单的因素之间的相互作用和相互影响会产生更高水平或更复杂的组成成分，体现了整体大于部分之和的思想，其结果是系统作为具有更大的生产力、更强的稳定性或适应性的整体被人所瞩目。涌现性描述了一种系统从低层次到高层次、从局部到整体、从微观到宏观的变化，它强调个体之间的相互作用，正是这种相互作用才导致具有一定功能特征和目的性行为的整体特

性的出现,使系统的宏观特性不同于系统组成因素或子系统本身的性质,即系统具有不同于各个子系统部分特征的整体宏观行为。

由于复杂系统的整体行为特征不等价于系统组成因素或子系统个体行为特征线性迭加之和,所以在系统仿真时无法用传统的数量方程或回归统计进行线性累加。随着计算机计算和存储能力的不断提高,人们发现可以把复杂系统中各个因素之间的非线性关系转化为计算机仿真模型,以模型仿真程序自动运行的方式推演系统,从而能在较短的时间内对那些现实世界中需要很长时间演化的系统进行动态仿真。计算机建模仿真方法逐渐成为研究动态复杂系统的有效手段,应用领域越来越广泛。复杂系统仿真的一般框架如图 1-1-1 所示。

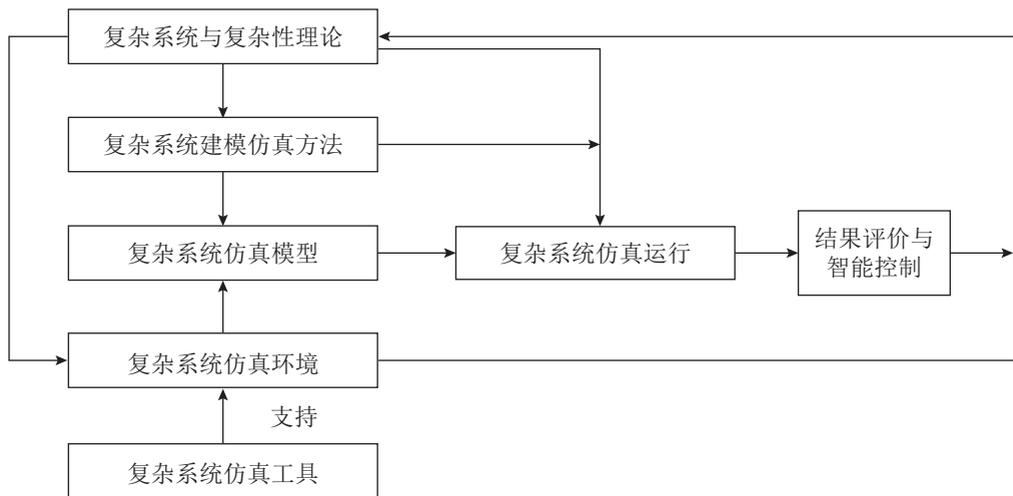


图 1-1-1 复杂系统仿真的一般框架

复杂系统仿真技术用途非常广泛,已经渗透到经济社会的各个领域,正不断促进各行各业的发展,为各行各业注入一股新的活力。因为人对复杂问题的理解能力是有限的,项目越复杂,就越有可能失败或做错事情。复杂系统仿真可以提高人的智力,帮助人们更好地对系统方案和计划进行可视化,使人们在一个较高的抽象层次上工作。这有助于正确地实施工作,并使工作进展得更快。有一个自然趋势:随着时间的推移,人们关注的实际系统都变得越来越复杂。虽然你今天可能认为不需要系统仿真,但随着系统的演化,当你决定开始使用复杂系统仿真技术时,可能为时已晚。

1.2 复杂系统建模仿真常用方法

1.2.1 离散事件系统建模仿真

在系统仿真中,将状态的瞬间变化称为事件。如果事件发生时间是一些非均匀离散的点,这样的事件称为离散事件,相应的系统称为离散事件系统,也称离散系统。在实

际工作中有许多这样的系统，例如：物流配送系统、排队系统、库存系统、通信系统、交通系统等。离散事件系统中，事件的发生可以看作在一个时间点上瞬间完成，这些时间点是离散的、不确定的，使得系统状态的变化具有随机性。

（一）离散事件系统基本要素

离散事件系统由相互关联或相互作用的要素组成，包括以下基本要素。

（1）实体（Entity）

实体是系统中有意义的个体，可分为：①永久实体，在系统仿真期间始终停留在系统中的实体，代表系统中的资源；②临时实体，在系统仿真期间流经系统，在仿真结束时已经离开系统的实体，代表系统中的加工或者服务对象。

（2）属性（Attribute）

实体所具有的特性称为实体的属性。属性是实体特征的描述，也称为描述变量，一般是实体所拥有的全部特征的一个子集，用特征参数或变量表示。

（3）状态（State）

状态指任意时刻系统中所有实体的属性的集合，描述系统在任何时间所必需的所有信息。

（4）事件（Event）

事件是使系统状态发生变化的、实体的瞬间行为，是系统状态变化的驱动力。系统的动态过程是由事件来驱动的，事件有时还会触发新的事件。事件一般分为必然事件和条件事件。

（5）活动（Activity）

活动是指实体在两个相邻发生事件之间的持续过程，它标志着系统状态的转移。活动持续一定时间，其开始和结束是由事件引起的。

（6）进程（Process）

进程由与某类实体相关的若干有序事件及活动组成，它描述了相关事件及活动之间的逻辑和时序关系。

（7）规则（Rule）

规则就是用于描述实体之间的逻辑关系和系统运行策略的逻辑语句和约定。

（8）仿真钟（Simulation Clock）

仿真钟是用于表示仿真模型内时间变化的时间标识，是仿真模型运行时序的控制机构。

（二）离散事件系统建模方法

离散事件系统建模的方法有实体流程图、活动循环图、Petri 网、Euler 网等，前两种最常用。

（1）实体流程图（Entity Flow Chart, EFC）

实体流程图方法采用与计算机程序流程图相类似的图示符号和原理，建立表示临时实体产生、在系统中流动、接受永久实体“服务”以及消失等过程的流程图。借助实体流程图，可以表示事件、状态变化及实体间相互作用的逻辑关系。由于计算机程序框图的思想 and 编制方法已广为人们所接受，加上实体流程图编制方法虽然简单，但对离散事

件系统的描述却比较全面等特点，使得实体流程图法的应用比较普遍。

图 1-2-1 给出了一个小型理发店服务系统的实体流程图。理发店有一个理发师，采取先到先服务的原则，在理发师忙时其余顾客需等待。

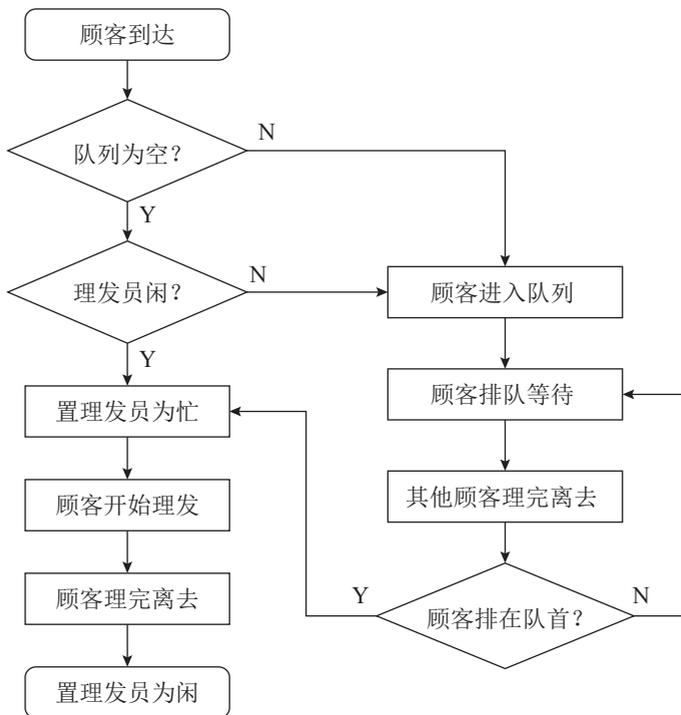


图 1-2-1 理发店服务系统实体流程图

(2) 活动循环图 (Activity Cycle Diagram, ACD)

活动循环图，也称活动周期图，它以直观的方式显示实体的状态变化历程和各实体之间的交互作用关系，是用于表示系统内各实体间逻辑关系的一种方法。活动循环图可以充分反映各类实体的行为模式，并将系统的状态变化以“个体”状态变化的集合方式表示出来，因此可以更好地表达众多实体的并发活动和实体之间的协同。它与实体类中的实体数量无关，只要系统的行为模式相同，即使它们的实体类型和活动周期不同，也可以用同一个活动循环图来描述。

实体活动循环图的绘制要以实际过程为依据，队列作为排队等待状态来处理，实体流程图中作为事件看待的某些操作或行为要拓展为活动来处理。图 1-2-2 给出了一个自动机床加工系统的活动循环图。自动机床由一名工人负责操作：如果机床的刀具完好，则为机床安装工件、然后按下运行按钮；如果机床的刀具损坏，则先要安装刀具；工人等待机床完成一次自动加工过程并停止运行后才能再次操作。实线部分是工人的活动循环图，虚线部分是自动机床的活动循环图。

(三) 离散事件系统仿真策略

离散事件系统仿真通过对真实系统中关键要素的抽象和组合，模拟一段时间内真实系统的运行过程，在仿真运行过程中离散事件系统的状态会因事件的发生而改变，而这

些事件发生在离散的时间点上。离散事件系统仿真的核心就是对事件和时间的合理处理，即用何种策略推进仿真钟，并建立起各类实体、事件、活动之间的逻辑关系。

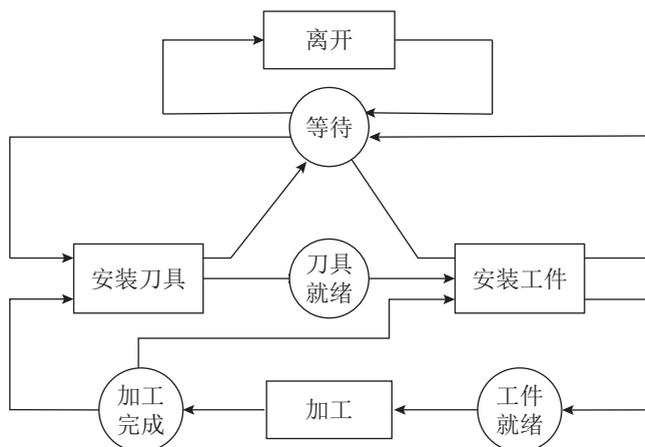


图 1-2-2 自动机床加工系统活动循环图

(1) 时间推进机制

在离散事件系统仿真过程中，仿真钟从 0 逐步增大到仿真结束时间，仿真钟的时间推进机制有两种：固定时间间隔推进机制，也称等步长推进；下次事件时间推进机制，也称变步长推进。两种机制的推进方式如图 1-2-3 所示。

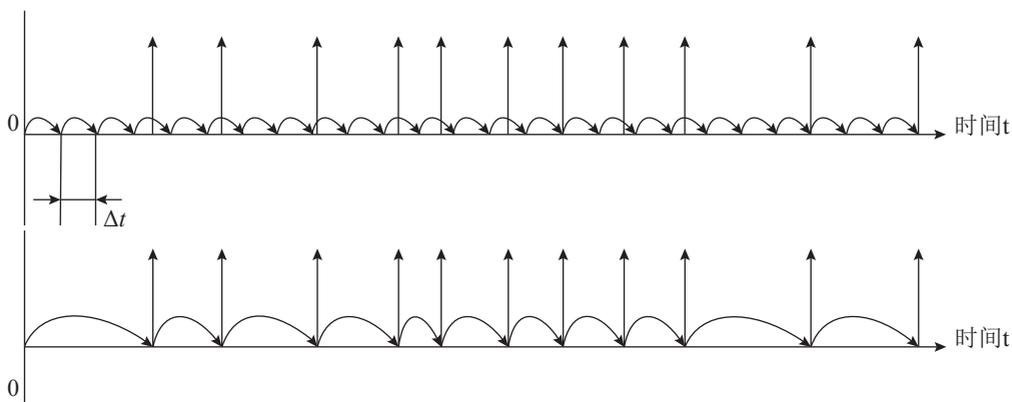


图 1-2-3 等步长和变步长时间推进机制对比

固定时间间隔推进机制首先要确定某一时间间隔作为仿真钟推进的固定时间增量，仿真钟按固定时间间隔等距推进。每次推进都需要扫描所有的活动，检查该时间区间内是否有事件发生。若无，则仿真钟继续等距推进。若有事件同时发生，除了记录该事件的时间参数外，还需要确定处理事件的先后顺序并依次执行这些事件。固定时间间隔的仿真钟推进机制存在一个难点，即如何合理确定时间增量，如果过大，则有可能丢失事件，产生误差；如果过小，则由于每步都要检查是否有事件发生，会大大增加仿真运行时间。

下次事件时间推进机制则是按下一个最早发生事件的发生时间来推进仿真钟。采用这种方法，当有一个事件发生，则将仿真钟推进到该事件发生时刻，在处理完事件所引起的系统变化之后，从将要发生的各事件中挑选最早发生的下一个事件，将仿真钟推进

到此事件发生时刻，然后继续重复以上步骤。这种机制下，仿真钟以不等距的时间间隔跳跃式推进，直到仿真运行满足终止条件为止。

(2) 事件调度机制

离散事件系统仿真大多数采用变步长时钟推进机制，即下次事件时间推进机制。在这种时间推进机制下，事件调度是核心，离散事件系统仿真程序通过定义事件及每个事件发生对系统状态的变化，按时间顺序确定并执行每个事件发生时相关的逻辑关系，同时策划新的事件来驱动模型的运行，其仿真程序流程图如图 1-2-4 所示。

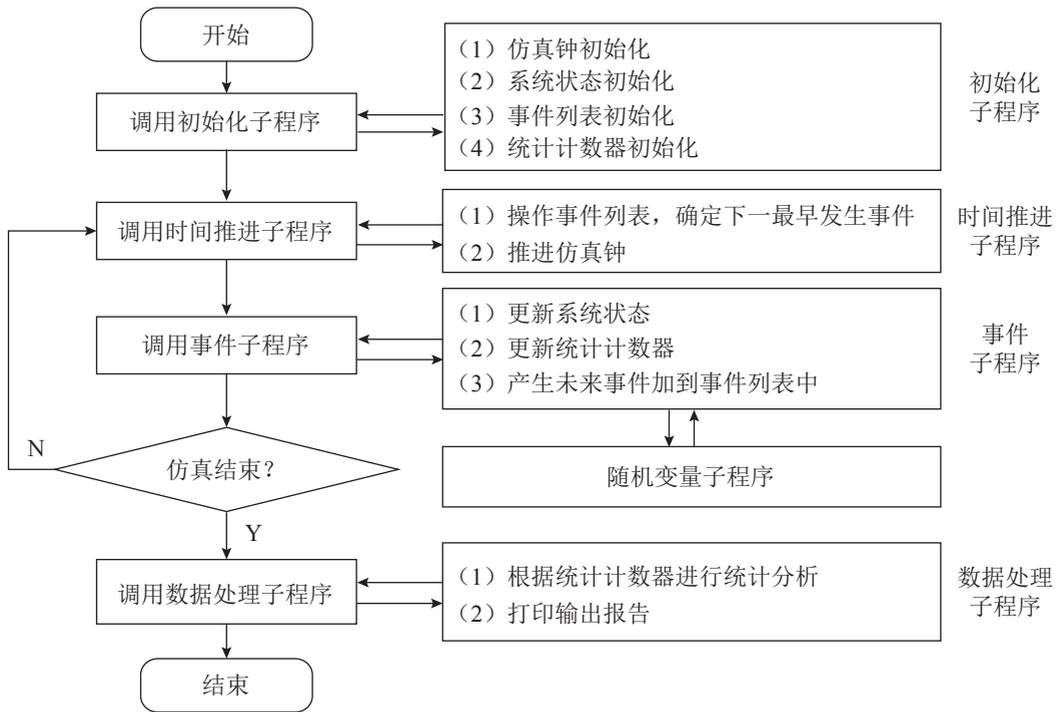


图 1-2-4 下次事件时间推进机制下的离散事件系统仿真程序流程图

(四) 离散事件系统仿真实用软件

目前市场上大量的专业仿真软件都是以离散事件系统建模仿真为主要应用领域，图 1-2-5 至图 1-2-14 列出了最常见的十种典型离散事件系统仿真软件，相关详细信息可以在相应软件产品网站查看。

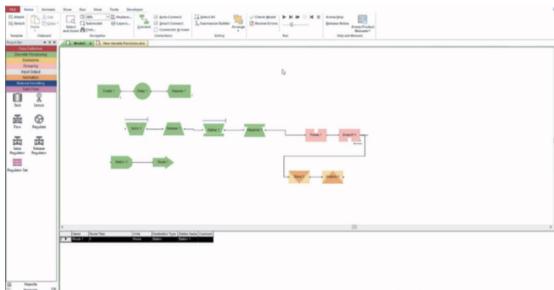


图 1-2-5 Arena 仿真软件界面

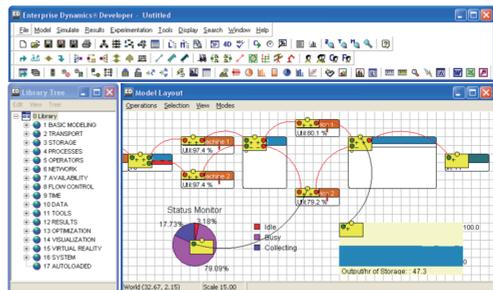


图 1-2-6 Enterprise Dynamics 仿真软件界面

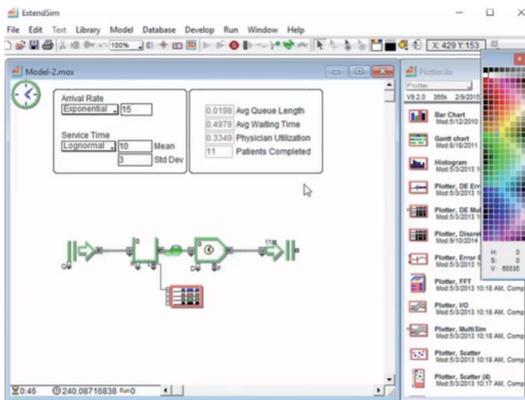


图 1-2-7 ExtendSim 仿真软件界面

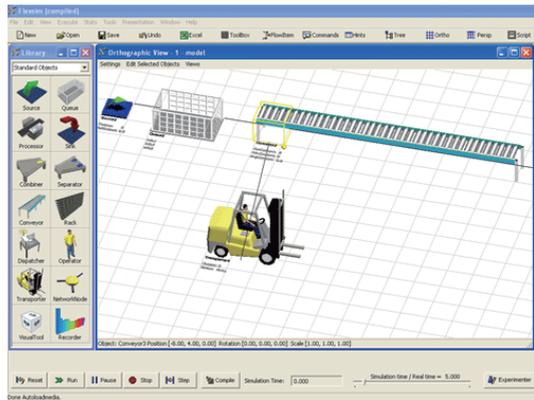


图 1-2-8 Flexsim 仿真软件界面

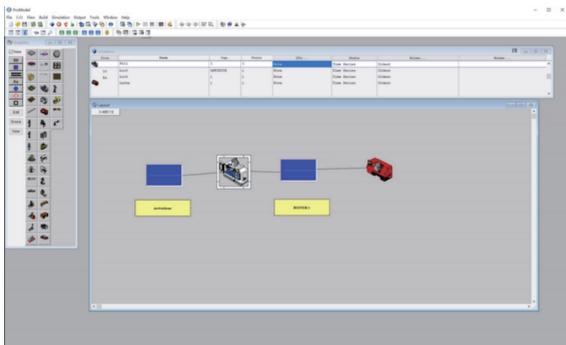


图 1-2-9 ProModel 仿真软件界面

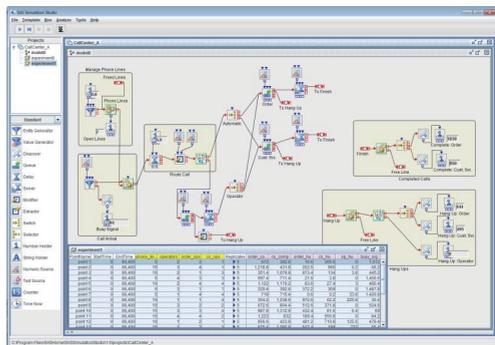


图 1-2-10 SAS Simulation Studio 仿真软件界面

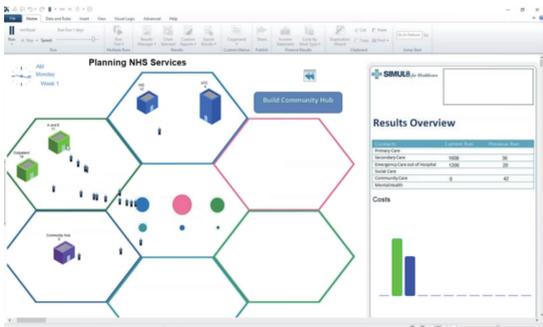


图 1-2-11 Simu8 仿真软件界面

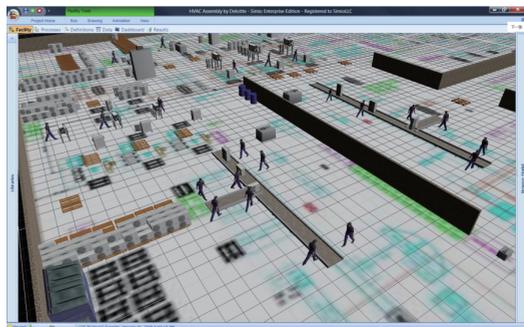


图 1-2-12 Simio 仿真软件界面

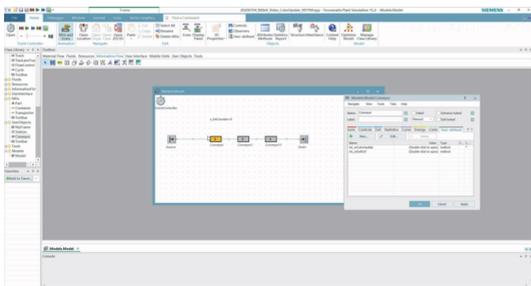


图 1-2-13 Plant Simulation 仿真软件界面

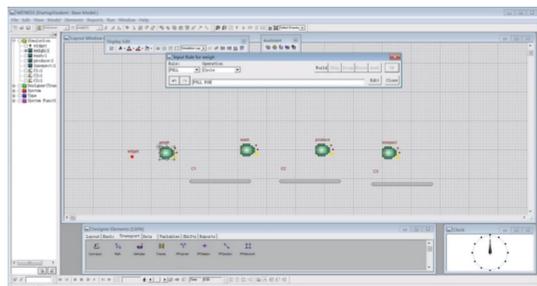


图 1-2-14 Witness 仿真软件界面

1.2.2 系统动力学建模仿真

系统动力学(System Dynamics, SD)是20世纪50年代美国麻省理工学院(MIT)的Forrester教授提出的系统仿真方法。系统动力学基于系统论,吸取反馈理论与信息论的精髓,综合控制论、信息论、决策论的成果,借助计算机仿真技术,对社会经济系统进行定性和定量相结合的系统分析。

(一) 系统动力学基本要素

按照系统动力学的观点,系统界限是封闭的,应把那些与建模目的关系密切、重要的变量都划入边界内。一个最简单的系统动力学模型如图1-2-15所示。

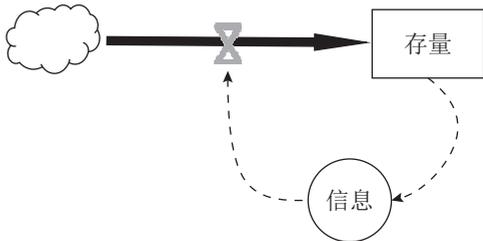


图 1-2-15 系统动力学存量流量图

系统动力学存量流量图的基本组成要素如下。

(1) 存量(Level), 代表图例 , 也称为水平、存量、积累量、流位等, 它是系统的状态, 也就是系统的某个指标值。

(2) 流量(Rate), 代表图例 , 也称为决策函数、速度、速率、流率等, 它控制着存量的变化。

(3) 实物流, 代表图例 , 用来连接存量和流量, 它表示在系统中流动着的物质, 具有守恒的性质。有的系统动力学软件用双线箭头表示物质流。

(4) 信息流, 代表图例 , 它指向流量, 表示根据什么信息控制流量, 模拟信息传递的过程。有的系统动力学软件用实线箭头表示信息流。

(5) 辅助变量, 代表图例 , 是存量和流量之间信息传递或转换过程的中间变量。

(6) 常量, 代表图例 , 它是系统中相对不变的量。

(7) 源或汇, 代表图例 , 源指来源, 汇指去向。

(二) 系统动力学反馈回路和方程

(1) 因果回路图

因果回路图(Causal Loop Diagram, CLD)是指由两个或两个以上的因果关系连接而成的闭合回路图示方法, 它定性描述了系统中因果变量之间的因果关系, 是表示系统反馈结构的重要工具。

一张因果回路图包含多个变量, 变量之间由标出因果关系的箭头所连接, 称为因果链。变量之间的因果关系往往具有不同的性质, 其影响作用可正可负, 通常用因果链的正、负极性加以定义: 因果链取正号, 表示A的变化(因)使B(果)在同一方向上发生变化; 因果链取负号, 表示A的变化(因)使B(果)在相反方向上发生变化。

反馈回路就是由一系列的因果与相互作用链组成的闭合回路, 其极性取决于回路中因果链符号的乘积。为了确定回路的极性, 可沿着反馈回路绕行一周, 看一看回路中全部因果链的累积效应: 若反馈回路包含偶数个负的因果链, 则其极性为正; 若反馈回路

包含奇数个负的因果链，则其极性为负。正反馈回路的作用是使回路中变量的偏离增强，而负反馈回路则力图控制回路的变量趋于稳定。如图 1-2-16 举例所示，正反馈回路也称做增强回路，用“+”或“R”标识；负反馈回路也称做平衡回路，用“-”或“B”所标识。

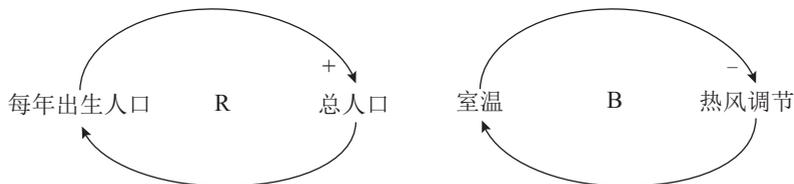


图 1-2-16 正反馈回路和负反馈回路

(2) 系统动力学方程

如图 1-2-17 所示，以一个商店进销存的系统动力学存量流量图为例。顾客订货后，由店员处理这些订单，并按照订货单到商店的仓库取货，再将货发送给顾客。为了保持商店的仓库总有货，商店就必须按一定规则订货来补充仓库中的库存。从中可以发现，存量流量图（含有因果回路图）只能定性描述出系统各要素间的逻辑关系与系统构造。

要定量分析系统动态行为，必须使用系统动力学方程。系统动力学方程确定的过程，就是把存量流量图转换成描述变量间函数关系的数学方程的公式化过程。把非正规的、概念的构思转换成正式的、定量的数学方程式，其目的在于利用计算机程序一步一步算出变量随时间的变化，以研究模型假设中隐含的动力学特性，并确定解决问题的方法与对策。

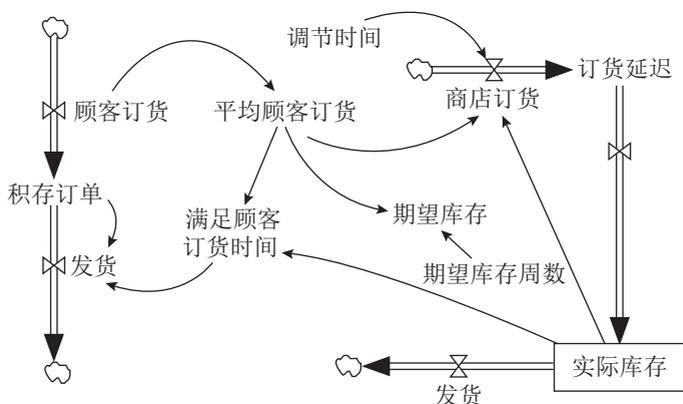


图 1-2-17 某商店进销存的系统动力学存量流量图

(三) 传统系统动力学仿真软件

20 世纪 50 年代，系统动力学发展初期，仿真软件系统采用的是 SIMPLE (Simulation of Industrial Management Problems with Lots of Equation)。SIMPLE 后来发展成为 DYNAMO，DYNAMO 名称来自“动态 (Dynamics)”和“模型 (Model)”的混合缩写。DYNAMO 只是提供了建模语言和编译环境，缺乏图形化界面操作支持，第三方开发的图形 DYNAMO 建模仿真扩展工具也一直没有得到大规模广泛应用。

进入 20 世纪 90 年代后，陆续出现了一大批系统动力学仿真软件，如：STELLA (Structural Thinking, Experiential Learning Laboratory with Animation)、iThink、Vensim、

Powersim、NDTRAN、DYSMAP (Dynamic Simulation Modeling Analysis and Program) 等。

现在, 在我国应用较为广泛的系统动力学专用仿真软件是 Vensim。Vensim 最早由 Ventana 公司在 1988 年开发, 是一个可视化的建模工具, 可以实现系统动力学模型的建模、仿真、分析以及优化, 软件界面如图 1-2-18 所示。

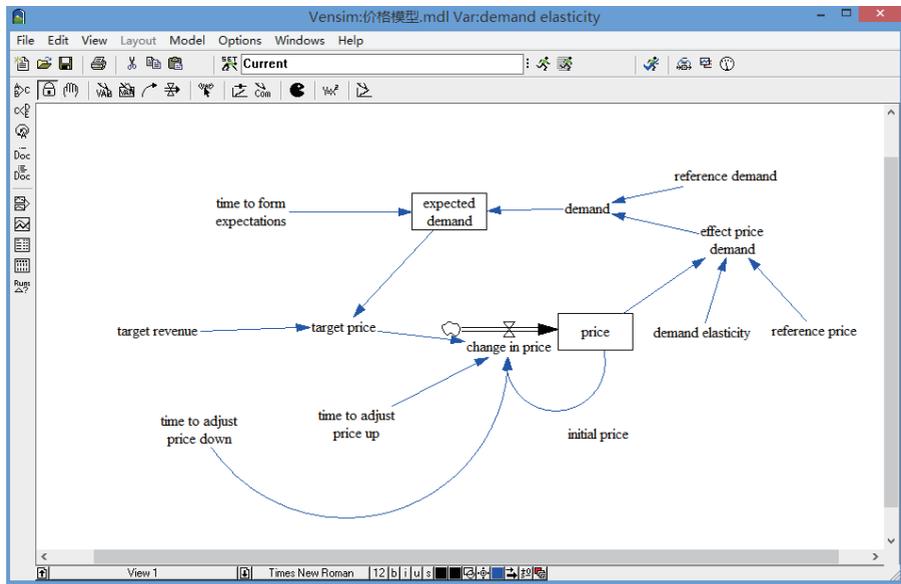


图 1-2-18 Vensim 仿真软件界面

1.2.3 多智能体建模仿真

多智能体 (Multi-Agent) 仿真是 20 世纪 80 年代出现的一种建模仿真方法, 在许多领域得到广泛应用, 尤其适用于异构的、分散的复杂系统。基于多智能体的仿真注重的是分散而不是集成, 每一个智能体是对复杂系统中某个单独个体的简单仿真。智能体具有更高的主动性、自治性和智能性, 它们之间存在动态、自主的交互行为。通过交互、合作与协调机制, 多个功能单一的智能体间的交互作用可以产生复杂的行为, 涌现一系列复杂现象。多智能体建模方法能够充分利用计算机系统的并行计算和分布式计算的能力, 实现传统方法难以完成的复杂系统仿真。

(一) 智能体和多智能体

(1) 智能体 (Agent)

Agent (智能体, 也译为主体、代理等) 最初来源于分布式人工智能领域的研究。人工智能专家 Minsky 在他 1986 年出版的著作《思维的社会》(The Society of Mind) 中提出了 Agent, 认为社会中的某些个体经过协商之后求得问题的解, 这些个体就是 Agent。一般认为, 智能体的特性包括: 自治性 (智能性)、社会 (交互) 性、反应性、合作性、移动性、理性、诚实性、友好性等。

事实上, 智能体的特性常常因为应用的领域不同而有所不同, 这也就形成对智能体

理解的不同。至今还没有一个普遍接受的关于智能体 (Agent) 的定义, 但人们普遍形成共识: 智能体是所研究系统的某种抽象 (可以是系统物理实体的抽象, 也可以是系统功能逻辑的抽象), 它能够在一定的环境中为了满足其设定目标, 而采取一定的自主行为; 智能体总是能够感知其所处的环境, 适应环境的变化, 并且具有可以影响环境的行为能力。

(2) 多智能体系统 (Multi-Agent System, MAS)

传统人工智能的目的是构造出具有一定智能的单个个体对问题进行求解, 单一智能体范式取得了一定的成功, 但是随着应用的深入, 人们发现受有限理性的限制, 单一智能体很难对存在于动态开放环境之中的大规模复杂问题进行求解。

Durfee 和 Lesser 在 1989 年提出了多智能体系统 (Multi-Agent System, MAS) 的概念。多智能体系统是由多个可以相互交互的智能体 (Agent) 所组成的系统, 系统中每个智能体都是自主的行为实体, 封装了状态和行为, 因而相对独立; 同时, 不同的智能体通过通信进行交互, 智能体之间可能存在复杂的关系。可以看出, 多智能体系统实际上是对社会智能的一个抽象, 许多现实世界中的群体都具有这些特征。

多智能体系统的特点主要有: 有限视角, 即每个 Agent 都面临不完全信息或只具备有限能力; 没有系统全局控制; 数据分散; 异步计算。

(二) 多智能体仿真策略

从某种意义上讲, 多智能体是最简单的建模仿真方法, 它为用户提供了一种独特观察视角, 从下到上来构建模型。用户使用多智能体仿真, 可以不用知道系统整体行为, 可以不识别系统关键变量及其动态变化, 可以不用确定整体业务逻辑, 也不需要像用系统动力学仿真那样练习抽象能力, 或者像用离散事件系统仿真那样总是强迫自己从流程方面思考问题。

多智能体建模仿真时, 用户只需了解系统中的个体是如何单独运行的, 通过创建智能体并定义其行为, 然后连接创建的各个智能体使其互动, 或将其放置在具有动态特性的特定环境中, 使系统的全局行为从大量 (数十个、数百个、数千个、数万个、数十万个、数百万个, 乃至更多) 智能体并发的独立行为中涌现出来。

多智能体仿真需要重点关注以下关键内容:

- ① 识别在实际系统中的那些至关重要的对象, 确定为模型中的智能体。
- ② 识别智能体间的具有持续性 (或部分持续性) 的关系, 建立相应连接。
- ③ 识别智能体在实际空间的变化规律, 包括智能体的速度、行进路线等。
- ④ 识别智能体生命周期中的重要事件, 这些事件可能由外部驱动, 也有可能是由智能体自身原因导致的内部事件。
- ⑤ 识别定义智能体的行为, 包括智能体对外界事件的反应、智能体的状态及变化、智能体内部事件处理、智能体的内部流程等。
- ⑥ 识别确定智能体之间的通信模式及其规则。
- ⑦ 识别智能体需要记录的各类信息, 确定存储方式和信息内容。
- ⑧ 识别存在于全部智能体外并被全部智能体分享的信息或动态, 定义其为模型全局变量。

(三) 传统多智能体仿真软件

传统多智能体仿真软件有：美国西北大学网络学习和计算机建模中心的 NetLogo，芝加哥大学社会科学计算实验室的 Repast，美国爱荷华州立大学 McFadzean、Stewart 和 Tesfatsion 开发的 TNG Lab，意大利都灵大学 PietroTerna 开发的 JES，美国布鲁金斯研究所 Parker 开发的 Ascape，美国桑塔费研究所的 Swarm 等。其中，Swarm 在我国应用较多。

Swarm 提供了基于脚本语言的非图形界面的多智能体仿真环境，是一种多时间线程运行机制，使得其中构建的大量智能体能并行运行和互相作用。Windows 系统中，Swarm 仿真运行环境可以在 Java 开发包（JDK）基础上构建。Swarm 仿真模型源文件编译后运行界面如图 1-2-19 所示。

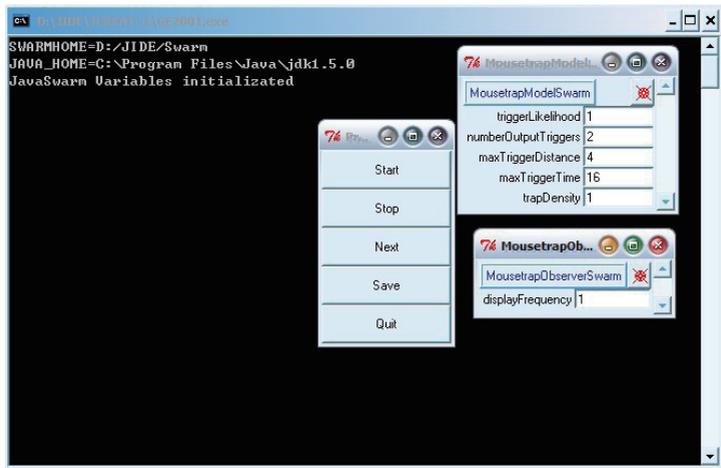


图 1-2-19 Swarm 仿真运行界面

1.2.4 多方法集成建模仿真

前述三种建模仿真方法事实上可视为三种不同的映射真实世界系统的观点：离散事件系统建模把真实系统的动态视为一系列基于实体的事件，是过程导向的建模仿真方法；系统动力学建模以连续变化视角，从真实系统抽象出存量、流量、反馈回路和方程；多智能体建模则将系统从个体的角度加以描绘，进而实现智能体之间以及智能体和环境之间的互动。

但是在工作实践中，往往用户研究的对象系统并不能用单一方法来建模仿真，或者说用单一的建模仿真方法并不能很好地对其进行模拟。因为，只使用一个方法工具，用户不可避免地要使用变通措施（非自然的、烦琐的语言结构），或者将一部分问题隔离在建模范围之外（将其视为外生），这种做法会带来一些问题。这种情况下，多方法集成建模仿真变得尤为重要。通过无缝集成和组合不同的仿真建模方法，构建起高效且可管理的仿真模型，可以克服单一建模仿真方法的局限性，并在每个方法中获得最大收益。

为实现多方法集成仿真，人们往往会运用高级程序语言来编写和运行仿真程序，如 C++ 语言或者 Java 语言，或采用专门的计算机仿真语言来进行，如 SLAM（Simulation

Language for Alternation Modeling)、SIMSCRIPT、GASP 等。近年来,一些最新的专业仿真软件也开始全面支持多方法集成仿真,如完全基于标准 Java 开发的仿真软件 AnyLogic。

AnyLogic 支持在一个仿真模型中使用前述三种建模仿真方法,并在仿真运行过程中保证各部分同步和交互。一个 AnyLogic 的传染病和诊所的多方法集成仿真运行结果,如图 1-2-20 所示。



图 1-2-20 AnyLogic 多方法集成仿真运行界面



扫码看彩图

1.3 常见系统仿真软件对比

计算机系统仿真被应用于大多数业务领域,在物理世界和数字世界之间架起一座桥梁,包括制造和材料处理、运输、仓库操作、资产管理和业务流程等。在模拟复杂业务系统时,需要一个多功能的计算机仿真软件工具。它有助于将各种对象系统和功能逻辑合并到一个模型中,更好地表示它们之间的相互联系,并以透明的方式优化它们。

下面对 1.2 中提到的 11 种常用仿真软件的主要参数进行对比,如图 1-3-1 至图 1-3-7 所示(来源:www.anylogic.com)。

		 软件厂商	 典型应用	 面向市场	 厂商其他软件
 ANYLOGIC	AnyLogic	多方法通用仿真方法应用, 包括离散事件、多智能体、系统动力学等	<ul style="list-style-type: none"> • 供应链 • 运输 • 仓库运营 • 铁路物流 • 采矿 • 石油和天然气 • 道路交通 • 客流量 • 制造和材料处理 • 卫生保健 • 业务流程 • 资产管理 • 营销 • 国防 • 政府机构 	AnyLogistix——供应链仿真和优化软件 AnyLogic Cloud——允许AnyLogic用户在任何设备上从Web浏览器运行和访问模型, 比较结果, 创建自定义仪表板以及执行各种实验的Web服务	
 ARENA	Rockwell Automation	用于对现有和推荐的系统进行仿真和分析, 以及操作分析	<ul style="list-style-type: none"> • 制造业 • 供应链 • 政府机构 • 卫生保健 • 物流 • 食物与饮料 • 打包 • 采矿 • 呼叫中心 	无	
 ENTERPRISE DYNAMICS	INCONTROL Simulation Solutions	制造业、物流, 以及材料处理仿真	<ul style="list-style-type: none"> • 仓储 • 配送中心 • 机场和港口 • 医疗保健和药品 • 快速消费品 	Pedestrian Dynamics——人群仿真软件应用程序, 专为在复杂基础设施中创建和执行大型行人仿真模型而设计	
 EXTENDSIM PRO	Imagine That Inc	专业级工具, 用于建模和分析复杂的离散速率、连续、基于代理和混合系统	<ul style="list-style-type: none"> • 消费产品 • 卫生保健 • 能源 • 石化 • 纸浆/纸张 • 运输 • 制药 • 半导体 • 军事和政府 • 采矿 	ExtendSim DE——入门级通用, 离散事件和连续仿真工具	
 FLEXSIM	FlexSim Software Products, Inc.	任何过程的仿真和建模, 目的是分析、理解和优化该过程	<ul style="list-style-type: none"> • 制造业 • 打包 • 仓储 • 物料搬运 • 供应链 • 物流 • 卫生保健 • 工厂 • 航空航天 • 采矿 	FlexSim Healthcare——用于分析、优化和更好地了解医疗保健系统的仿真和建模工具	

图 1-3-1 仿真软件供应商和市场对比 (一)

		 软件厂商	 典型应用	 面向市场	 厂商其他软件
软件厂商和市场	 PROMODEL OPTIMIZATION SUITE	ProModel Corporation	流程优化和改进, 资源利用、系统容量、吞吐量等约束分析	<ul style="list-style-type: none"> • 国防 • 政府机构 • 制造业 • 制药 • 物流 • 仓库和配送中心 	Enterprise Portfolio Simulator——基于Web的多个同步项目计划的仿真分析 FutureFlow Rx——ADT决策, 患者流程和床位管理 MedModel——临床环境的计算机仿真 Process Simulator——过程仿真
	 SAS SIMULATION STUDIO	SAS	离散事件仿真: 供应链、资源管理、容量规划、工作流分析和成本分析	<ul style="list-style-type: none"> • 制造业 • 银行 • 制药和医疗保健 • 能源 • 政府机构 • 零售 • 教育 • 运输 	无
	 SIMUL8 PROFESSIONAL	SIMUL8 Corporation	装配线、生产线平衡战略规划、运营、医疗保健系统、精益、共享服务、容量计划	<ul style="list-style-type: none"> • 制造业 • 卫生保健 • 教育 • 工程 • 供应链 • 物流 • 政府机构 • 精益 • 汽车 • 呼叫中心 	无
	 SIMIO ENTERPRISE EDITION	Simio LLC	适合专业建模师和研究人员, 强大的面向对象建模和集成的3D动画, 用于快速模型	<ul style="list-style-type: none"> • 航空航天和国防 • 机场 • 卫生保健 • 制造业 • 采矿 • 石油和天然气 • 供应链 • 运输 	无
	 PLANT SIMULATION	Siemens Product Lifecycle Management Software Inc.	离散事件仿真、可视化、生产吞吐量、物流的分析和优化	<ul style="list-style-type: none"> • 汽车制造和物流 • 航空航天和国防 • 消费产品 • 物流 • 电子产品 • 机械 • 卫生保健 • 咨询 	无
	 WITNESS	Lanner	专业建模和应用开发的快速、生产性预测仿真软件	<ul style="list-style-type: none"> • 商业计划 • 流程优化 • 决策 	无

图 1-3-2 仿真软件供应商和市场对比 (二)

		 支持的操作系统	 可兼容软件执行专门功能	 可被外部程序控制或运行	 是否支持多CPU
 ANYLOGIC		Windows、Mac、Linux	<ul style="list-style-type: none"> • Excel、Access等数据库 • OptQuest • Stat::Fit • 任何Java/DLL库 	AnyLogic模型可以作为独立的Java应用程序导出，这些应用程序可以从任何其他应用程序或由其他应用程序运行。它们也可以通过AnyLogic Cloud Web服务在线运行	是
 ARENA		Windows	OptQuest	Visual Studio	
 ENTERPRISE DYNAMICS		Windows	否	否	
 EXTENDSIM PRO		Windows、Mac	<ul style="list-style-type: none"> • Excel、Oracle、Access、SQL Server、MySQL • Stat::Fit • JMP • Minitab • 任何DLL库 	任何可配置为自动化控制器(如Excel或Access)的Windows应用程序都可以与ExtendSim进行控制和通信	
 FLEXSIM		Windows	<ul style="list-style-type: none"> • Excel 等数据库 • C++ 程序 	OLE以及ActiveX	
 PROMODEL OPTIMIZATION SUITE		Windows	<ul style="list-style-type: none"> • Excel和Access • Stat::Fit • MiniTab 	<ul style="list-style-type: none"> • Excel和Access • C# • VB和VBA 	
 SAS SIMULATION STUDIO		Windows、Linux	SAS和JMP软件，可以通过外部运行，也可以通过SAS程序块嵌入	任何可以启动Java应用程序的程序	
 SIMUL8 PROFESSIONAL		Windows	<ul style="list-style-type: none"> • Excel • Stat::Fit • OptQuest • SQL数据库 	Excel和任何可以启动COM的IDE	
 SIMIO ENTERPRISE EDITION		Windows	<ul style="list-style-type: none"> • Microsoft Azure • Wonderware • OptQuest • .Net应用程序 • Excel、Access、SQL Server、MySQL 	<ul style="list-style-type: none"> • Wonderware • OptQuest • .Net应用程序 	
 PLANT SIMULATION		Windows	<ul style="list-style-type: none"> • Matlab • Excel • Simatic IT • Teamcenter • Autocad • Microstation 	<ul style="list-style-type: none"> • Excel • Siemens PLCSIM高级版 • OPC、OPC UA • ODBC • ERP (SAP或Oracle) 	
 WITNESS		Windows	否	否	

图 1-3-3 仿真软件技术兼容性对比

模型建立		+			
		输入分布拟合	图形模型建立	输出分析支持	代码重用
	ANYLOGIC	31个预定义分布和自定义分布。 Stat::Fit、ExpertFit以及其他用于分布拟合的软件	可以	<ul style="list-style-type: none"> • 报告 • 模型执行日志 • 图表 • 输出到内置数据库或任何外部数据存储（数据库、电子表格、文本文件） 	可以
	ARENA	输入分析器以拟合分布		Arena Output Analyzer和Process Analyzer用于检查结果，用户也可以使用外部产品。	
	ENTERPRISE DYNAMICS	自动调整——内部功能		实验向导——内部特性	
	EXTENDSIM PRO	35个预定义的分布。用于分布拟合的Stat::Fit软件		<ul style="list-style-type: none"> • 输出到图表和报告 • 集成场景管理器，具有对话或数据库因子和响应、灵敏度分析、置信区间、甘特图、分位数和区间统计分析。 • 导出到外部分析应用程序 	
	FLEXSIM	与ExpertFit集成		仪表板中的全套图表以及广泛的Excel输出选项	
	PROMODEL OPTIMIZATION SUITE	提供16种统计分布，与Stat::Fit集成		<ul style="list-style-type: none"> • Output Viewer • Minitab • Excel 	
	SAS SIMULATION STUDIO	通过JMP和SAS软件集成。		通过SAS软件产品进行输出分析，包括稳态分析	
	SIMUL8 PROFESSIONAL	软件和Stat::Fit中的自定义选项		无	
	SIMIO ENTERPRISE EDITION	适用于ExpertFit以及Stat::Fit，支持表驱动输入采样。		用于风险分析、灵敏度分析、自定义仪表板、枢轴表中的全面数据，导出摘要或外部软件包的详细信息	
	PLANT SIMULATION	22个预定义分布		<ul style="list-style-type: none"> • Datafit • Charts • Sankey • Bottleneck analyzer • Energy Analyzer • Neural networks 	
	WITNESS	无	无		

图 1-3-4 仿真软件模型建立对比（一）

		 优化	 运行时调试	 模型封装	 封装是否免费
 ANYLOGIC		包括OptQuest, 此外用户可以使用任何自定义优化算法		模型可以作为独立的Java应用程序导出, 也可以通过AnyLogic Cloud Web服务在线共享	是
 ARENA		专用OptQuest		Arena Runtime	/
 ENTERPRISE DYNAMICS		提供对各种第三方优化器的支持		通过提供该软件的免费查看器许可证	是
 EXTENDSIM PRO		Evolutionary Optimizer包含在ExtendSim的所有版本中		试用版运行在Extensim中构建的任何模型, 分析运行时允许进一步的模型分析	提供免费和付费选择
 FLEXSIM		一个优化引擎, 由OptQuest提供, 可作为一个附加		免费试用版本的FlexSim可以运行任何使用FlexSim构建的仿真模型	是
 PROMODEL OPTIMIZATION SUITE		SimRunner	可以	无	/
 SAS SIMULATION STUDIO		通过数据传输到SAS/OR软件, 可以通过SAS程序块嵌入仿真模型中。		无	/
 SIMUL8 PROFESSIONAL		OptQuest		SIMUL8 Studio和SIMUL8 Web技术	提供免费和付费选择
 SIMIO ENTERPRISE EDITION		OptQuest(可选)充分利用了所有处理器。具有多目标和模式边界优化的特点		需要Team Edition或更高版本来封装模型	是
 PLANT SIMULATION		遗传算法、局部优化、神经网络、爬山、动态规划、分支定界		内置封装功能	是
 WITNESS		无		云部署、实验、优化	否

图 1-3-5 仿真软件模型建立对比 (二)

模型建立		 通过编程构建模型  批量运行/实验设计  成本分配/成本计算  离散/连续混合建模			
		通过编程构建模型	批量运行/实验设计	成本分配/成本计算	离散/连续混合建模
	ANYLOGIC	可以	灵活的用户界面，可创建以下实验：参数变化、比较运行、蒙特卡洛、灵敏度分析、校准和自定义	可以	可以
	ARENA		Process Analyzer可以批量运行一系列不同的模型		
	ENTERPRISE DYNAMICS		提供实验向导和场景管理器		
	EXTENDSIM PRO		用户选择将运行结果存储在内部数据库中或导出到外部应用程序；DOE包括手动和全因子两个选项，分别用于JMP定制设计和Minitab优化设计		
	FLEXSIM		内置了一个实验引擎		
	PROMODEL OPTIMIZATION SUITE	场景管理员	可以	可以	
	SAS SIMULATION STUDIO	不可以	实验设计；通过JMP或SAS软件集成，在模拟工作室界面或自动化(与交互修改)中手动操作	不可以	不可以
	SIMUL8 PROFESSIONAL	可以	多个复制和场景管理	可以	可以
	SIMIO ENTERPRISE EDITION		运行具有多个复制的手动方案，同时充分利用所有处理器进行内置排名和选择		
	PLANT SIMULATION		支持分布式仿真的实验管理器		
	WITNESS		无		

图 1-3-6 仿真软件模型建立对比（三）

		 动画	 动画输出	 实时观看	 三维动画	 CAD图纸导入
动画	 ANYLOGIC	可以	可以	可以	可以	可以
	 ARENA		不可以			
	 ENTERPRISE DYNAMICS		可以			
	 EXTENDSIM PRO		不可以			
	 FLEXSIM		可以			
	 PROMODEL OPTIMIZATION SUITE		不可以			
	 SAS SIMULATION STUDIO		不可以			
	 SIMUL8 PROFESSIONAL		可以			
	 SIMIO ENTERPRISE EDITION		可以			
	 PLANT SIMULATION		可以			
 WITNESS	可以					

图 1-3-7 仿真软件动画对比