

基于 HLA 时间管理的实时时间控制和乐观时间同步算法设计*

姚新宇 黄柯棣

(国防科技大学自动控制系 长沙 410073)

摘要 首先提出 HLA 系统中仿真时间一致性的概念,并归纳出三条实现策略,同时给出充分性证明;然后从支撑系统实现的角度,分别针对基于 HLA 时间管理的实时仿真控制和乐观时间同步给出了设计算法,并进行了严格的数学证明。

关键词 HLA, RTI, 时间管理, 实时仿真, 乐观时间同步

分类号 TP391.9

Algorithms of Real-Time Control and Optimistic Time Synchronization Based on Time Management in HLA

Yao Xinyu Huang Kedi

(Department of Automatic Control, NUDT, Changsha, 410073)

Abstract The paper puts forward the conception of "time coherence" and the realization policy in HLA. Its sufficiency is also proved. In order to construct a HLA system, we solve the problems of real-time control and optimistic time synchronization based on HLA time management mechanism by providing the appropriate algorithms and the mathematical proofs.

Key words HLA, RTI, Time Management, Real-Time Simulation, Optimistic Time Synchronize

HLA(High Level Architecture)是 DIS(Distributed Interactive Simulation)的进一步发展,HLA 在统一数据接口规范的基础上提出统一的方法接口以及仿真应用开发的规范,其内容更加广泛。本文主要就 HLA 的重要组成部分 RTI(Run Time Infrastructure)中的时间管理问题展开讨论,首先介绍 HLA 的时间管理策略,然后在此基础上重点解决 HLA 系统中实时仿真的时间控制和乐观时间同步控制问题。

1 HLA 时间推进机制

HLA 系统是采用分布对等的时间管理模式,即通过相互交换消息来实现节点仿真时间的推进和协调,根本目标是获得分布仿真时间的一致性,保证分布节点间特定消息的接收和处理次序。对于 HLA 系统,可定义为:

定义 1,HLA 的时间一致性,如果 RTI 能保证 HLA 系统中所有联邦成员(federate)都按消息的时间戳顺序接收 TSO 消息,则称此 HLA 系统的时间是一致的。

文[1]就 HLA 的时间管理机制作了较为详细的介绍,但其描述较含糊,甚至有不少前后矛盾之处。为便于后文中的讨论,我们根据自己的理解和经验,将 HLA 的时间管理机制概括为三条规则(限于篇幅,其中术语的定义可以参考文[1]):

(1) 联邦成员的规则:每个联邦成员选择一个 Lookahead。这是联邦成员对未来行为的预测。假如一个成员在时间 T 时的 Lookahead 为 L ,则该成员不能产生时戳小于 $T+L$ 的消息。

(2) RTI 的约束:RTI 为每个受逻辑时间约束的联邦成员 F_i 计算 LBTS $_i$ (时间戳下限): $LBTS_i = \min(T_j + L_j)$,其中 F_j 为所有向联邦成员 F_i 发送 TSO 消息的其它具有逻辑时间校正特性的联邦成员,

* 863 计划资助和重点预研项目

1999 年 3 月 8 日收稿

第一作者:姚新宇,男,1971 年生,博士生

T_j, L_j 分别为 F_j 的当前逻辑时间和 Lookahead。当 F_i 是受逻辑时间约束的联邦成员时, RTI 不能将时间戳大于 $LBTS_i$ 的 TSO 消息发送给联邦成员 F_i , 即不允许 F_i 的联邦成员时间推进到大于 $LBTS_i$ 。

(3) 联邦成员时间推进规则: 联邦成员的时间推进是由联邦成员自身向 RTI 发出时间推进请求服务, 将本联邦成员的时间由 t_{old} 推进到 t_i ; 而由 RTI 按第二条规则和其它联邦成员协调($t_i - LBTS_i$)后, 将时间戳不大于 t_i 的 TSO 消息发送给该联邦成员, 然后调用应答服务来满足时间推进请求。

我们认为这三条规则对实现 HLA 系统的时间一致性是充分的, 有以下两个结论:

定理 1 在上述三条规则条件下, 受逻辑时间约束的联邦成员接收 TSO 消息的时间戳必然不小于联邦成员时间。

定理 2 在采用上述三条规则条件下, HLA 系统的时间一致性得到实现。

虽然 HLA 的时间管理机制保证了对分布仿真的逻辑时间的一致性维护, 这对一般的仿真应用是可接受的, 但对于某些特定的场合, 仅有这些规则就显得不充分了, 因此, 必须在此基础上设计专门的解决算法来满足需求。

2 HLA 的实时时间推进控制算法设计

分布仿真的一个主要应用领域是实时仿真, 无论是人在回路的人员培训仿真或硬件在回路的武器系统作战体系的仿真都强调仿真的实时性。这两类实时仿真都要求仿真时间 T_s 和自然时间 t 保持一致, 对于 HLA 而言, 就是要求联邦成员时间的推进和自然时间保持一致。

为实现实时仿真, 首先要将自然时间引入到 HLA 系统中, 这通常采用定时联邦成员 F_c (clock federate 或称 timing federate) 的方法实现。 F_c 具有逻辑时间校正特性, 根据自然时间(实际上是该联邦成员自身的墙上时间(wall clock))推进其联邦成员时间 T_c , 其时间步进步长为 T_{hc} , Lookahead 等于 L_c 。通过此定时联邦成员可以对具有逻辑时间约束特性的联邦成员的成员时间推进进行约束:

定理 3 在仿真中的任一时刻 t , 设定定时联邦成员的成员时间为 $T_c = \lfloor t / T_{hc} \rfloor \times T_{hc}$, 则 HLA 的时间管理机制可以保证 HLA 系统中所有逻辑时间约束的联邦成员的成员时间必然不大于 $T_c + L_c$ 。

然而采用这种方法, 只能约束联邦成员时间的上限, 对时间下限则无能为力, 导致的后果是部分联邦成员时间可能远远滞后于自然时间。这显然不能满足实时仿真的要求。因此我们在此基础上提出改进实时控制算法:

(1) 设置定时联邦成员的 Lookahead(L_c) 大于等于 T_{hc} ;

(2) 需要满足实时性的联邦成员 F 设置为逻辑时间约束特性, 其 Lookahead 不小于时间步进步长 T_{hk} ;

(3) 需要满足实时性的仿真联邦成员的仿真帧计算时间(包括消息传输的时间)要小于仿真步长, 即要求仿真计算的速度足够快。若步进时间步长为 T_{hi} , 则最大的帧计算时间要小于 T_{hi} 。

这个改进算法既不影响合理性, 又对联邦成员时间的上限和下限加以控制, 取得较理想的效果。

定理 4 设 F_i 为 HLA 联邦中任何一个具有逻辑时间约束特性的联邦成员, 其成员时间、时间步长、Lookahead 分别为 T_i, T_{hi} 和 L_i , 在忽略 RTI 内部消息的传输延迟时有: $T_c + L_c - T_{hi} \leq T_i \leq T_c + L_c$ 。

证明: “ $T_i \leq T_c + L_c$ ”证明见定理 3。

为证明 “ $T_c + L_c - T_{hi} \leq T_i$ ”, 采用反证法。假设 $T_c + L_c > T_i + T_{hi}$, 则定时联邦成员 F_c 不能阻塞 F_i 的成员时间推进。

根据实时控制算法的第三条, 仿真计算不能阻塞联邦成员时间推进, 因此必然存在一个具有时间校正特性的联邦成员 F_j 阻塞了 F_i 的成员时间推进, 即有 $T_j + L_j < T_i + T_{hi}$ 。

根据实时控制算法第二条, 有 $L_j \geq T_{hj}$, 故有 $T_j + T_{hj} \leq T_j + L_j < T_i + T_{hi} < T_c + L_c$ 。

显然定时联邦成员 F_c 也不能阻塞 F_j 的成员时间推进, 因此必然存在一个闭环, 即还存在具有时间校正特性的联邦成员 F_k , 有: $T_j + L_j < T_k + T_{hk}; T_k + L_k < T_j + T_{hj}$ 。

根据实时控制算法的第二条有: $L_j \geq T_{hj}, L_k \geq T_{hk}$, 则得到:

$$T_j + T_{hj} \leq T_j + L_j < T_k + T_{hk}; T_k + T_{hk} < T_k + L_k < T_j + T_{hj}$$

这显然矛盾,因此假设不能成立,故“ $T_c + L_c - T_{hi} - T_i$ ”成立。

3 HLA 中的乐观时间同步算法设计

前文中提到的 HLA 的三条时间管理规则为维护联邦时间的一致性,是通过滞后 TSO 消息的发送,直至时间约束关系得到满足后(即确定 TSO 消息具有最小时间戳),才将消息发送给特定的联邦成员,这种严格按时间戳次序执行的策略在分布计算的时间同步研究领域称为保守同步控制。

与保守同步算法相对应,HLA 支持一种称为乐观同步的机制来实现消息的快速发送,乐观同步机制允许 RTI 无需确定消息是否具有最小时间戳,就将该消息发送给对应的联邦成员,而当后来的消息具有更小的时间戳时,则通过反绕(rollback)操作的机制来纠正消息的次序。一般称采用乐观同步机制的联邦成员为乐观联邦成员,反之则称为保守联邦成员,HLA 的乐观时间管理机制主要包括乐观消息发送服务和反绕操作服务:

(1) 乐观消息发送规则:乐观联邦成员调用 HLA 提供的乐观消息请求服务将对应 RTI 内部消息队列中所有的消息读入到联邦成员内部。

(2) 反绕操作规则:乐观联邦成员恢复历史状态,对这一段时间中发送的所有消息发送对应的“反消息”以取消这些消息。RTI 接收到“反消息”后检测 RTI 内部的消息队列,如对应的消息还在 RTI 内部,则删除该消息;如该消息已经发送给联邦成员,则将反消息也发送到该联邦成员,由该联邦成员进行处理;联邦成员接收到反消息后,如果对应消息还没有处理,则删除该消息,否则又调用反绕操作。

至于乐观联邦成员在何种情况下启动反绕操作,在文[1]中没有给出一致的说明,这和所采用的乐观同步控制的具体算法相关联,结合我们设计的算法,我们认为应该按规则 3 进行:

(3) 反绕启动规则:乐观联邦成员一旦发现新消息的时间戳小于最近被处理(不是被接收的)消息的时间戳,则调用反绕操作。此规则保证了乐观联邦成员处理消息的时间戳次序。

在同一个 HLA 仿真联邦(federation)中,乐观联邦成员和保守联邦成员可以同时存在,因此 RTI 应该采取一定的措施保证两类联邦成员之间的和谐且透明的共处。其中的一个主要难点在于保守联邦成员不能处理“反消息”,而对于乐观联邦成员而言,其并不能区分与之交互的联邦成员的乐观或保守特性(这是透明性要求),因此其发送消息时也只能采取统一的模式,在这种情况下要保证保守联邦成员不接收“反消息”,即要保证其所处理的所有消息都不会被撤销。

为解决此问题,一种思路是引入全局虚拟时间(Global Virtual Time,定义为所有联邦成员时间、所有未处理消息时间戳中的最小值)作为乐观联邦成员时间,显然 GVT 作为 HLA 全系统的逻辑时间的最小值,使得所有消息的时间戳大于 GVT,从而保证小于 GVT 的(历史)消息不会被撤销。但这只是问题的一个方面,保守联邦成员的要求满足了,但对乐观联邦成员,这种模式下的反绕操作无法保证消息的时间戳处理次序(规则 3 不能实现)。经过细致的分析和对仿真过程的理解,结合我们提出的反绕操作规则,提出了一种乐观时间推进算法,概括为三条原则:

(4) 联邦成员时间设置规则:乐观联邦成员采用 GVT 作为联邦成员时间。

(5) 联邦成员时间特性规则:乐观联邦成员同时具有逻辑时间校正和约束特性。

(6) 联邦成员发送消息规则:乐观联邦成员发送消息的时间戳要不小于所处理消息的最大时间戳(必然不小于 GVT,即联邦成员时间)加上 Lookahead。

其中最关键的是第 6 条规则,这条规则对乐观联邦成员处理超前消息方面加以约束,保证了乐观联邦成员在反绕操作的支持下,实现消息的时间戳处理次序,同时也保证保守联邦成员不接收乐观消息。在具体实现中,乐观联邦成员可以选择性地滞后处理不必要的超前消息,只要该消息对所发送消息没有影响,则对发送消息的时间戳没有特殊的限制。

在这六条规则限制下必然有:

定理 5, 设乐观成员的 Lookahead 为 L , 则在 HLA 仿真的任何时刻,乐观成员发出的时间戳不大于 $GVT + L$ 的(历史)消息必然不会被取消。

证明 设 t 时刻 HLA 仿真联邦的 GVT 为 $GVT(t)$, F 为任一乐观联邦成员, Lookahead 为 L , Msg

为 F 发送的任一(历史)消息, 其时间戳 $TS_s < = GVT(t) + L$ 。

根据规则 6, F 发送 M_{sg} 时, 其所处理消息的最大时间戳 TS_p 必然不大于 $GVT(t)$, 即 $TS_p < = GVT(t)$;

根据 GVT 的定义, 当前新消息的时间戳 TS 必然不小于 $GVT(t)$, 即 $TS > = GVT(t)$;

则有 $TS > = GVT(t) > = TS_p$; 根据规则 3 知, M_{sg} 必然不会被撤销。

定理 6 在 HLA 的时间管理机制下, 具有逻辑时间约束特性的保守联邦成员必然不会接收“反消息”。

证明 设 t 时刻 HLA 仿真联邦的 GVT 为 $GVT(t)$, F 为任一具有逻辑时间约束特性的保守联邦成员。

根据规则 4, 乐观联邦成员的成员时间为 $GVT(t)$, Lookahead 为 L , 则 F 的 LBTS 满足: $LBTS < GVT(t) + L$, 故根据 HLA 的保守时间推进机制得知: F 接收(来自乐观联邦成员的)消息的时间戳必然小于 $GVT(t) + L$, 根据定理 5, 此时 F 所接收的来自乐观联邦成员的消息必然不会被撤销, 即不会接收“反消息”。

从上述分析可知, 乐观联邦成员是通过 GVT 来实现乐观联邦成员和保守联邦成员的协调共处的, 而对于分布仿真情况下, GVT 的准确值是难以得到的(因为 RTI 并不为进行 GVT 的估算进行额外的消息传递), 因此乐观联邦成员必须对 GVT 进行局部估算。我们提出的估算方法为:

(7) GVT 估算规则: 乐观联邦成员采用 LBTS 作为 GVT 的估计值 \overline{GVT} 。

因此乐观联邦成员在调用乐观消息请求服务后, RTI 推进其成员时间为 LBTS 和 TSO 消息时间戳二者中的最小值。在规则 7 的 GVT 的估计算法下仿定理 5 得到:

定理 7 设乐观成员的 Lookahead 为 L , 其对 GVT 的估计为 \overline{GVT} ; 则在 HLA 仿真的任何时刻, 乐观成员发出的时间戳不大于 $\overline{GVT} + L$ 的消息必然不会被取消。

证明 设 t 时刻 HLA 仿真联邦的 \overline{GVT} 为 $\overline{GVT}(t)$, F 为任一乐观联邦成员, Lookahead 为 L , M_{sg} 为 F 发送的任一(历史)消息, 其时间戳 $TS_s < = \overline{GVT}(t) + L$ 。

根据规则 6, F 发送 M_{sg} 时, 其所处理消息的最大时间戳 TS_p 必然不大于 $\overline{GVT}(t)$, 即 $TS_p < \overline{GVT}(t)$; 因为当前新消息的时间戳 TS 必然不小于 LBTS, 根据 $\overline{GVT}(t)$ 的定义, 有 $TS > \overline{GVT}(t)$, 则有 $TS > \overline{GVT}(t) > TS_p$; 根据规则 2 知 M_{sg} 必然不会被撤销。

由定理 7 取代定理 5, 定理 6 的证明同样成立。因此这 7 条规则组成的乐观时间同步算法, 在减小乐观联邦成员消息传输延迟的同时还保证了消息处理的 TSO 次序, 并和原有的保守联邦成员透明共处, 符合 HLA 的时间管理机制。

4 小结

HLA 现在是分布仿真研究的一个热点, 而其中的时间管理机制又是仿真系统的核心与难点。尤其在我国, 到目前为止还没有成功开发一个完备的 HLA 支撑系统, 对其中的若干问题还存在认识模糊乃至有争议的地方。我们主要是根据对相关材料的理解, 并结合正在开发的分布仿真支撑系统介绍我们的研究成果, 旨在同国内同行交流。

参考文献

- 1 HLA Time Management Design Document Version 1.0, DMSO, 1996
- 2 High Level Architecture Run-Time Infrastructure Programmer's Guide, DoD, 1997
- 3 Darrin West, Wide Virtual Time Application to Parallel Discrete Event Simulation and the HLA RTI, 1998, summer simulation conference
- 4 邱晓刚. 复杂大系统的面向对象仿真实理论和方法研究. 国防科技大学[博士学位论文], 1998
- 5 Blair R C. Tools for the Development and Distribution of Web-Based HLA Simulations, 1998, summer simulation conference
- 6 Celso Hisata. An Investigation of The Use of World Views With. Shared Variables In Time Warp, Simulation Practice and Theory, 1997-5