

一类智能系统的自主能力嵌入研究与仿真^①

Study and simulation for a class Intelligence system self – optimizing – embedded

柯宗武 田开坤 汪自云 (黄石 湖北师范学院计算机科学与技术系 435002)

摘要:本文提出了有智能系统的自主性嵌入机制与实现方法,并给出一个仿真实例,是指导高性价比系统设计的有效策略。

关键词:自主性 高性价比 嵌入机制 有效策略

1 引言

按照[1]、[2]、[3]、[4]、[5] 对一类有智能系统的有效性优化策略与实现机制,业界应该对基于静态模型和确定算法,甚至基于规则的自适应动机的分析达到了极高的地步。多年来深入与细致的程度也驱使研究路径和实现方法无懈可击。就如同一个复杂系统的可靠性提升企图采纳一切可用的技术一样。然而,事实上不可靠性依然存在。所以,大量的硬件冗余就是系统终极可靠的必然选择。因为多个相同系统同时发生功能故障的概率极小(例如:八套设备有七套同时发生故障的概率 14 年可能一次)。工程上(特别是重点工程)不吝啬安全系数的增加也在于此。

1967 年,Preparata F P., Metze G. 和 Chen R T. [6] 建立了让系统智能环节相互测试,根据测试结果找出不良行为的构想,它是用最小的开销实现系统自治的典范。所以,真正具有高自治度的自主系统才是终极有效的。

然而,在原本用智能支撑的系统内部嵌入自主能力,只是投入软件开销。例如:基于 CPU 的当前进程属性,自主调节主时钟频率;基于被交换的信息属性,自主调度路由;基于过程信号属性,自主选择采样周期等等。都会使系统整体的 QoS 大大提升。

2 自主能力与自主系统

设一个系统的自主能力 P 表示为:

$$P = f(E)$$

$$\text{且 } E = \{e_i\} (i=1,2,\dots,n)$$

即自主能力 P 是由它所能作为的全部自主环节决定的。当然,这里描述的可作为自主环节可能是二值逻辑(刚性的可作为与不可作为)的。比如:CPU 当前处于等待进程,就自动触发主时钟频率环节(有级分频),降低主时钟频率,以降低 CPU 及相关部件的处理速度,节省功耗。显然,也能够基于进程的状态(比如支持长时段的文本打印)无级地动态自调节相关的环境支持(如指令寻址与译码、运算器处理等均可因此而退出高速状态或间歇)。

可以假定:

$E = \{e_i\}$ 集的每一元素都平均地被自主调节,且每一次(ΔT 时段)被作为(自主性发挥)所获取的能耗节省或生命期延长(指有效生命期可量化设备)派生的效益因子就可用 $G = \{g_i\}$ 表示。

所以,N 个可作为环节产生的整体效益 Y 为:

$$Y = N \sum_{i=1}^n g_i$$

则 T 时段内收益为 G:

$$G = \frac{T}{\Delta T} Y \quad (1)$$

凡被考查的系统有自主能力作为所获取的效益与被考查的系统全部 E(自主能力作为环节)成正比。

以上描述可知,一个系统的自主能力获取可刚性地实现。而自主系统的设计则是一个综合全部自主能

^① 本课题得到湖北省教育厅重大科研课题资助(2001/2003-5)

力作为过程的圆融。因为自主系统性能提升不是简单的让每一个自主能力作为环节都分别做到极至,即不是 $\lim_{m \rightarrow \infty} g_i = 1$, 由于 g_i 的提升可能使 g_{i+1} 下降。这类似与明星球队不都是由明星组成一样。

设计一个自主系统很困难, 因为一个系统可自主的环节实在太多, 要考虑 N 个自主环节分别提升自主能力所产生的对其他 $N-1$ 个环节自主能力提升的影响, 且影响程度的静态度量或标定是可行的。即:

第 i 个效益因子 g_i 提升对于其他 $N-1$ 个效益因子的影响度 R 可标定为:

$$R = \sum_{i=1}^{n-1} g_i | \text{Effective} |$$

也可能某些影响为零、为负(正作用), 但是, 对于一个自主系统, 以上的分析必须是动态实时的, 否则就失去意义。由于系统综合效率 $E_s = \alpha E + \beta E_c$, 其中 α, β 是系统资源和交互资源占系统总成本的比例 ($\alpha + \beta = 1$), E 为处理机效率 (E 等于加速比/处理机数), E_c 为成本效率 (Cost-effective)。每一个自主环节(智能单元)嵌入, 事实上只是算法工作量 W (Workload) 增加, 可导致 E 提升。

为了有效地描述自主系统的性能或整体自主度, 给出以下相关的约束:

- 自主系统的可靠性保障
- 自主系统的效率提升
- 自主系统的可实现性界定

2.1 自主系统的可靠性 P_e

设 $P_e = 1$ 为完全可靠(平均无故障时间趋向无穷), 因为嵌入的自主机制 C_i 自身可靠, 所以 C_i 嵌入可能导致的系统可靠性 P_e 下降仅仅来自于软件进程增加或分支选择引入。则规约定义为:

(1) 由于分支或进程派生引入的时延 ΔT 必须满足 $(t + \Delta T) \leq t^*$, (t 为原有进程(作业)的实际时段, t^* 为原有进程允许时限(Dead-time))。

(2) 任何派生进程中均保持响应规定进程原有的调度策略(类似于响应实时软中断, 可接受中断陷阱(Interrupt trap))。

(3) 嵌入机制被激活期 (ΔT) 以外必须安全释放, 以利于原有进程平滑回归。

在维护以上规约的前提下, 原有系统嵌入自主机制后仍能保持可靠性 P 不变。

2.2 自主系统的效益

显见的效益表现在可能的耗损缩减和有限生命期

的部件或系统由于指令性松弛而获取的寿命延长。为实现稳定的效益, 可给出以下规约:

(1) 自主机制嵌入与相应的效益因子标定(静态)。且动态效益的获取一般可源于系统所面向进程的属性, 比如, 长时段的打印作业所引入的自主机制可获取更大的效益。

(2) 为支持自主机制所产生的边际开销(比如, CPU 主频分频环节的引入)与所获得的效益选优, 原则上基于系统属性和任务属性。即:

边际开销 M 很小于自主机制效益总和 T , $M < T$
 $(T = \sum_{i=1}^{\infty} T_i)$

因为, 刚性的效益积累是巨大的, 如果面向海量系统则更如此。

(3) 由于自主机制被激活, 基于进程属性所产生的部件指令性松弛, 甚至待机, 可能由于系统类化后而大大简化设计, 简化度 D 满足:

$D = f(w, t)$ w 为类化系统的属性, t 为基于可自主作为的时间。

(4) 一个确定的系统, 由于自主机制嵌入所产生的整体效益由(1)式给出, 但静收益 $N < G$ 。

对于显见的静收益可量化, 而一般性量化很困难。

2.3 自主系统的可实现性

基于 Von. Neumann 机结构, 任何进程的运行都是指令序列的刚性链接, 并且系统面向的作业属性由于作业的一次提交便被系统映象, 随着固定作业的周而复始, 自主机制可随时被激活, 并不断牵引系统趋向更加自主。当然, 以下的规约应该是必须的。

(1) 作业属性 W 是可以段分的。 $W = \sum_{i=1}^n w_i$, 且存在或更多的 w_i 对系统效率的需求极低或可被忽略, 能很好地支持自主机制嵌入或自主系统生成。

(2) 如果设系统智能环节的底层开放(物理资源开放)度为 $Open$, 则:

自主系统的可实现性 $S = f(Open)$, $S \propto Open$

3 过程实例与仿真

根据上节的分析, 可以进行四个方面的仿真研究。即面向过程的实时或相对特定过程的实时行为在线交互式环境设计研究; 松散耦合群集系统(Cluster System)行为实时表征与策略指导过程研究; 过程现场三