

## 基于 AADL 语言的移动软件能耗评估方法\*

杜欣, 王晓红, 倪友聪, 罗增

(福建师范大学 软件学院, 福建 福州 350007)

通讯作者: 王晓红, E-mail: wangpink\_99@foxmail.com

**摘要:** 移动软件往往部署在电量受限的处理器上, 能耗已成为评价这类软件的一个重要质量属性. 与代码级和指令级相比, 在设计级进行能耗评估具有耗时短、成本低的优点, 近年来已成为软件工程学术界和工业界的研究热点. 目前虽已涌现出一些设计级能耗评估方法, 但这些方法大多未对软件构件的内部行为元素进行能耗评估, 导致了精度问题. 针对上述问题, 基于体系结构分析设计语言 AADL 和 StrongARM 处理器构建了一种移动软件能耗评估模型, 进一步定义了面向 AADL 语言的移动软件能耗评估过程, 在此基础上研发了一款能耗评估工具, 进而提出一种基于 AADL 语言的移动软件能耗评估方法. 实验结果表明该方法较已有 AADL 能耗评估方法在精度上有所提高.

**关键词:** AADL; 行为附件; 能耗评估; 移动软件

中文引用格式: 杜欣, 王晓红, 倪友聪, 罗增. 基于 AADL 语言的移动软件能耗评估方法. 软件报, 2015, 26(Suppl. (2)): 272-280. <http://www.jos.org.cn/1000-9825/15038.htm>

英文引用格式: Du X, Wang XH, Ni YC, Luo Z. Approach to estimating energy consumption for mobile software based on AADL language. Ruan Jian Xue Bao/Journal of Software, 2015, 26(Suppl. (2)): 272-280 (in Chinese). <http://www.jos.org.cn/1000-9825/15038.htm>

## Approach to Estimating Energy Consumption for Mobile Software Based on AADL Language

DU Xin, WANG Xiao-Hong, NI You-Cong, LUO Zeng

(Faculty of Software, Fujian Normal University, Fuzhou 350000, China)

**Abstract:** Mobile software is often deployed on processors with limited energy. Energy consumption has been an important quality attribute to evaluate such software. Compared with the code level and instruction level assessment, energy consumption assessment at design level has the advantages of low time consumption and low cost. In recent years, it has become a research focus in academic and industrial fields of software engineering. Currently, most of the methods on energy consumption evaluation did not evaluate the energy consumption of internal behavioral elements of software components, resulting in the problem of low precision. To tackle this problem, this study builds a model of energy consumption evaluation for mobile software based on AADL language and the StrongARM processor. Further, a process for evaluating energy consumption of mobile software is defined based on AADL language. In addition, a tool for evaluating energy consumption is developed. Lastly, a method for evaluating the energy consumption of mobile software is proposed based on AADL language. The experimental results show that the proposed method improves precision compared with existing evaluation method of energy consumption based on AADL.

**Key words:** AADL; behavior annex; energy evaluation; mobile software

随着智能手机和 PDA 等移动设备的日益普及和迅猛发展, 运行在这些设备上的应用软件丰富多样. 据 Google Play 和 App Store 的统计, 在 2014 年 Android 和 iOS 移动应用市场分别都有超过 143 万款移动软件被

\* 基金项目: 国家自然科学基金(61305079); 武汉大学软件工程国家重点实验室开放基金(SKLSE 2014-10-02); 福建省自然科学基金(2015J01235); 福建省教育厅 JK 类项目(JK2015006)

收稿时间: 2015-08-07; 定稿时间: 2015-10-12

应用到生活、运动、健康和医疗等诸多领域.不同于传统的台式机应用和 Web 应用,移动应用往往部署在电量受限的处理器上,能耗作为评价这类软件的一个重要质量属性,已成为用户选择和使用软件时的一个重要考量因素<sup>[1]</sup>.

目前人们已在指令级<sup>[2]</sup>、源程序级<sup>[3]</sup>和系统设计级<sup>[4-6]</sup>提出了一些软件能耗的评估方法.由于系统设计级的能耗评估方法具有消耗资源少、耗时短和代价小的优点,近年来成为一个研究热点<sup>[7]</sup>.针对于系统设计级的能耗评估方法,刘啸滨等人<sup>[8]</sup>提出了一种利用 BP 神经网络在体系结构层估算软件能耗的模型;该能耗模型对有效代码行数、构件数目、平均构件接口复杂度、平均路径复杂度、平均构件耦合度这 5 个体系结构特征量进行度量,并利用 BP 神经网络拟合出软件特征量与软件能耗的非线性函数关系.该方法侧重于体系结构的结构特性,未给出动态行为对软件能耗的影响.Kim 等人<sup>[9]</sup>提出了一种基于 UML 行为模型对移动软件能耗进行评估的方法.该方法基于 UML 语言的顺序图和交互图等行为元素,给出了能耗元素的识别方法,并利用预先定义的能耗模型库,评估各个构件和软件系统的能耗值.针对体系结构建模语言 AADL<sup>[10]</sup>,Senn 等人<sup>[11]</sup>提出了一种移动软件的能耗评估方法.该方法根据处理器的不同硬件参数以及处理器的可调度性,从整体上对软件的能耗进行评估,并基于 PowerPC405 给出该方法的具体实施过程.该方法虽能较好地建模和评估移动软件在硬件上调度的能耗的影响,但尚未充分考虑软件构件内部行为元素对能耗的影响,还存在精度也不高的问题.

针对这一问题,本文基于工业级体系结构分析设计语言 AADL 和 Tan 等人<sup>[12]</sup>已测得指令集能耗值的 StrongARM 处理器,构建了一种移动软件能耗评估模型 EEM4AADL,进一步定义了 AADL 模型能耗评估过程 EEP4AADL,在此基础上研发一款能耗评估工具 EET4AADL,进而提出了一种移动软件能耗评估方法 EE4MS-BA.

本文第 1 节阐述 EEM4AADL 能耗评估模型.第 2 节给出 EEP4AADL 能耗评估过程的核心步骤.第 3 节概要介绍能耗评估工具 EET4AADL 的总体设计.第 4 节给出案例研究的实验结果和分析.最后总结全文.

## 1 EEM4AADL 能耗评估模型

EEM4AADL 能耗评估模型是以 StrongARM 处理器<sup>[12]</sup>的虚拟指令及其能耗值作为基础,通过在构件内部行为的 AADL 附件(AADL behavior annex)<sup>[13]</sup>中识别出能耗行为元素(energy behavior element,简称 EBE),并根据定义 AADL 语言能耗行为元素与 StrongARM 处理器虚拟指令的映射关系而构建的.

### 1.1 StrongARM处理器的虚拟指令及其能耗值

Kim 等人<sup>[9]</sup>将虚拟指令分为虚拟私有指令(VPI)和虚拟系统指令(VSF).VPI 指令是广义的汇编指令,而 VSF 指令是广义的操作系统功能指令.Kim 等人在文献[4]中测得了每条虚拟指令对应的能耗值,见表 1.该虚拟指令是基于 StrongARM 处理器的,可用于运行在该款处理器上移动软件的能耗评估.结合该能耗库中虚拟指令能耗值和移动软件能耗评估模型可获取每个 EBE 的能耗值.

表 1 StrongARM 处理器的虚拟指令所对应的能耗值

VPI	Type	Energy (nJ)	VSF	Parameters	Macro-Model (nJ)
load	char	14.1	fork()	-	132 202.6
	short	28.1	waitpid()	-	25 077.3
	int	13.0	wait()	-	24 986.6
	float	13.0	signal()	-	9 460.6
	double	13.7	fopen()	-	17 886.6
store	char	13.2	fclose()	-	8 262.2
	short	26.4	fread()	x byte	5.1x+49308.5
	int	12.5	fwrite()	x byte	5.6x+32022.3
	float	12.8	shmget()	-	126 358.8
	double	13.2	shmat()	-	11 599.9

### 1.2 AADL语言行为附件中的能耗行为元素

能耗行为元素 EBE 是从 AADL 行为附件元模型<sup>[14]</sup>中提取的用于评估移动软件能耗的基本单元.通过对 AADL 行为附件元模型的分析得出 AADL 中的能耗行为元素,见表 2,根据 AADL 行为附件元模型中类所属的

不同类别(动作语言,控制结构,行为执行动作,声明接口等)来识别是否为 EBE.

根据表 2 中行为附件元模型中能耗行为的不同分类得出能耗行为元素列表见表 3,其中 AL(action language)为 AADL 行为附件元模型中定义的执行行为语言,BA(behavior action)为具体执行的行为,CS(control structure)为 AADL 行为附件元模型中定义的控制结构.

表 2 识别 Behavior Annex 元模型中的 EBE

元模型	类型	EBE ?
Assignment Action	Action Language	Y
If/Else Statement	Control Structure	Y
Loop Statement	Control Structure	Y
Dispatch Condition	Behavior Action exec	Y
Subprogram Call Action	Behavior Action exec	Y
Parameter Label	Non-Behavior Action	N
Data Holder	Interface	N
Port Send Action	Behavior Action exec	Y
Binary Adding Operators	Action Language	Y
Multiplying Operators	Action Language	Y
Logical Operators	Action Language	Y
Relational Operator	Action Language	Y
Behavior State	Interface	N

表 3 能耗行为元素列表

分类	EBEs
AL	Bit operator(+, -, *, /, %, &, !, ==, >, <=, mod .etc ); Function operator(data read, write .etc)
BA	sub program call, port queue , port freeze, port send dispatch, port fresh, sub program access .etc
CS	(if ,else, end ,while or until ,loop, switch .etc)statement

### 1.3 AADL语言能耗行为元素与StrongARM处理器虚拟指令的映射关系

通过能耗行为元素 EBE 转换成简单的汇编指令来构建 EBE 与虚拟指令之间的映射关系.例如,AL 中进行 ‘+’ 运算需要先 load 数据,然后进行 add 运算并将运算结果 store 起来,因此 ‘+’ 可实现为 3 条虚拟指令,分别为 load, store 和 add.表 4 给出了 AADL 能耗行为元素与虚拟指令之间的映射关系集合.

表 4 能耗行为元素与虚拟指令之间的映射关系

元素 指令	ForOrForA llStatement	binaryAddingO perators: +	relationOp arator: <=	Subprogram CallAction	SharedDa taAction	logicalOpe rators: and	...	DispatchC ondition	EventData PortHolder
load	✓	✓	✓			✓			
store	✓	✓	✓			✓			
add		✓							
...									
compare	✓		✓						
branch	✓			✓				✓	
bitAND						✓	...		
fork()				✓				✓	
signal()								✓	✓
msgsnd()				✓	✓				✓
msgrcv				✓					✓
fread()					✓				

## 2 EEP4AADL 能耗评估过程

能耗评估过程 EEP4AADL 的总体流程如图 1 所示.该过程通过转换器 STG 将 AADL 行为模型转换成状态变迁图,再利用识别器识别出其中的能耗行为元素,最后利用生成器生成 AADL 行为模型的能耗评估报告.

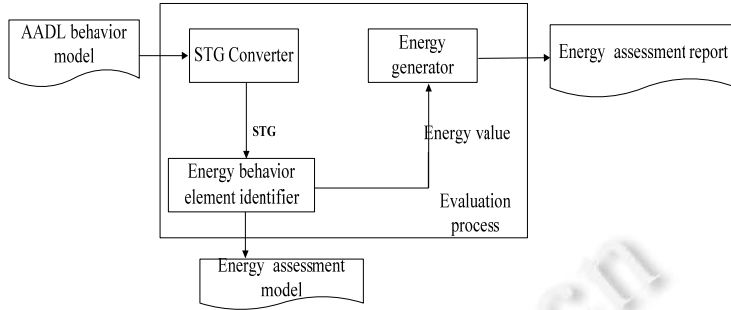


图 1 EEP4AADL 能耗评估框架

2.1 AADL行为模型到STG模型的转换

AADL 行为模型是结合 AADL 基本组件模型和 AADL Behavior Annex 构建的用于描述系统内部以及系统之间的信息交互行为设计模型.AADL Behavior Annex 通过定义状态(states)和变迁(transition)来描述系统各组件的行为,变迁可以定义触发条件(condition)及变迁的动作(action).AADL 行为模型和行为状态之间的变迁可以用一个有限状态自动机模型描述,定义如下:

定义 1. 用有限状态自动机描述的 AADL 行为模型表示为  $BM=(BS, BE, TR, IBS, FBS)$ ,其中,

- (1)  $BS$  是所有状态的集合,  $BS=\{bs_1, bs_2, bs_3, \dots, bs_m\}$ ;
- (2)  $BE$  是变迁的事件集合,包括变迁触发条件和变迁动作集合,  $BE=BE_a \cup BE_c$ , 变迁的动作  $BE_a=\{be_{a1}, be_{a2}, be_{a3}, \dots, be_{ak}\}$ , 变迁的触发条件  $BE_c=\{be_{c1}, be_{c2}, be_{c3}, \dots, be_{ck}\}$ ;
- (3)  $TR$  是所有状态间变迁的集合,状态转移函数  $TR: BS \times BE \rightarrow BS, TR(bs_i, be_i)=be_k$ ;
- (4)  $IBS$  是初始行为状态集, AADL 行为附件中规定有且只有一个初始行为状态,因此这个集合只有一个元素,即  $IBS=\{bs|bs \text{ 是初始行为状态}\}$ ;
- (5)  $FBS$  是终止行为状态集合.

STG 模型由节点、弧和变迁组成,变迁定义了节点迁移的方向,弧上定义了进入变迁的约束条件和变迁后需要执行的动作.STG 模型可以用一个五元组进行描述,其定义如下:

定义 2. 用 5 元组表示 STG 模型为  $SM=(N, T, F, C, A)$ ,其中,

- (1)  $N$  是节点集合,  $N=\{n_1, n_2, n_3, \dots, n_n\}$ ;
- (2)  $T$  是变迁集合,  $T=\{t_1, t_2, t_3, \dots, t_m\}$ ;
- (3)  $F$  是弧的集合,  $F=F_m \cup F_{m'}, F_{m'}=N \times T$ , 描述从节点到变迁的集合,  $F_m=T \times N$ , 描述从变迁到节点的集合;
- (4)  $C$  是约束条件集合,描述从节点到变迁的约束条件,  $C=\{c_1, c_2, c_3, \dots, c_n\}$ ;
- (5)  $A$  是执行动作集合,描述从变迁到节点后的执行动作  $A=\{a_1, a_2, a_3, \dots, a_n\}$ .

根据定义 1 和定义 2,比较 AADL 行为模型与 STG 模型的相似性,实现 AADL 行为模型元素到 STG 模型元素的转换函数  $BM\_SM$  可表示为

定义 3. AADL 行为模型元素到 STG 模型元素的转换;

- (1)  $BM\_SM(bs)=n, BM\_SM(BS)=N$ , 实现 AADL 行为状态到 STG 中节点的转换;
- (2)  $BM\_SM(be_c)=c, BM\_SM(BE_c)=C$ , 实现了 AADL 变迁的触发条件到 STG 变迁的约束条件的转换;
- (3)  $BM\_SM(be_a)=a, BM\_SM(BE_a)=A$ , 实现了 AADL 变迁的执行动作到 STG 变迁后需要执行的动作转换;
- (4)  $BM\_SM(TR)=F$ , 即  $BM\_SM(TR)=F_m \cup F_{m'}$ , 实现了将 AADL 中行为状态之间的变迁转换为 STG 中节点的进入弧(变迁到节点)和输出弧(节点到变迁).

图 2 给出了 AADL 行为模型与 STG 模型的转换具体例子.图 2(a)中定义了组件 Component1 和 Component2 及其行为描述,图 2(b)为依照上述转换方法得到的 STG 模型.Component1 中状态 start 和 exe 转换成 STG 中的对应节点 start 和 exe,即  $BM\_SM(start)=start, BM\_SM(exe)=exe$ ;触发条件  $be_{c1}=x>1 \text{ and } y<10$ ,转换成 STG 中的约

束条件  $c_1=x>1, y<10$ , 即  $BM\_SM(be_{c1})=c_1$ ; 变迁后的动作  $be_{a1}=exe\_action$  转换成 STG 中的进入动作  $a_1=exe\_action$ , 即  $BM\_SM(be_{a1})=a_1$ . Component2 的转换同理也遵循定义 3 所述转换函数.

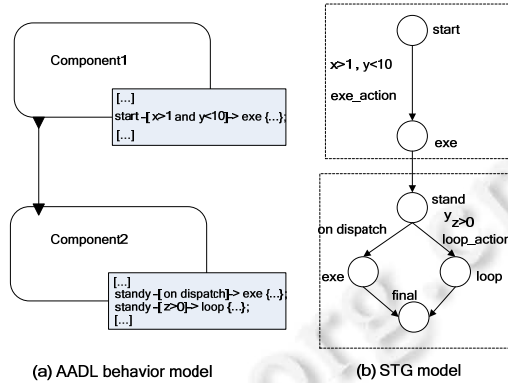


图2 AADL 行为模型转换成 STG 模型

## 2.2 能耗行为元素的识别

能耗行为元素的识别算法的描述见算法 1, 输入为包含  $n$  个节点的 STG 模型, 输出为识别后的能耗行为元素数组 EBEs[]. 在算法 1 中, EM 表示能耗评估模型, Successor\_Node 表示后续节点. 一个节点的后续节点可能有多个, 这样就构成了不同的遍历路径, 可分别针对每条路径进行遍历, 最后得到所有节点上的能耗行为元素.

**算法 1** 能耗元素识别算法.

输入: 包含  $n$  个节点的 STG 模型;

输出: 识别的能耗行为元素列表 EBEs[];

```

1  for each  $i$  in  $(1, n)$  do
2    if ( $N_i$  is the Start_Node) then
3      Identify_EBE_from_EM( $N_i$ );
      /*后续节点大于一个时*/
4    if (the Successor_Node of  $N_i > 1$ ) then
5      select  $N_k$  is one join node,  $K$  for all
6        Successor_Nodes;
7      for each  $k$  in  $(1, K)$  do
8        Identify_EBE_from_EM( $N_k$ );
9         $N_k =$  Successor_Node;
10     else
11       Identify_EBE_from_EM( $N_i$ );
12        $N_i =$  Successor_Node;
13     end if;
14   else
15      $N_i =$  Successor_Node;
16   end if;
17 end for;
18 return EBEs[];
/*识别能耗元素函数*/

```

```

19 Identify_EBE_from_EM(N);
20 for each j in (1,m) do
21     compare the Node_Type with ej∈EM;
22     if the Node_Type correspond to ej then
23         add ej into EBEs[];
24     end if;
25 end for;

```

### 2.3 能耗值的计算

根据算法 1 遍历 STG 模型的每个节点,得到所有的能耗行为元素后,可通过公式(1)计算出系统消耗的总能耗  $E_{total}$ .其中, $E_{vpi}$  和  $E_{vsf}$  分别表示虚拟私有指令(VPI)和虚拟系统指令(VSF)的能耗值.

$$E_{total} = \sum_{i=1} EBE_i \left( \sum_{j=1} E_{vpi_j} + \sum_{k=1} E_{vsf_k} \right) \tag{1}$$

## 3 EET4AADL 能耗评估工具

EET4AADL 能耗评估工具用于支撑上一节提出的评估过程,其总体架构分为 UI、Domain 和 Resource 三层,如图 3 所示.

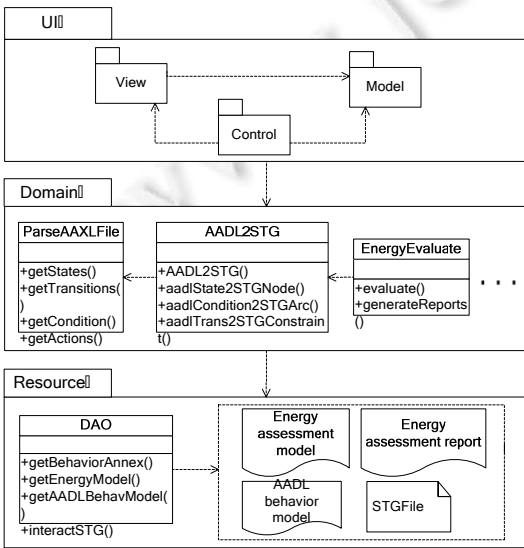


图 3 EET4AADL 能耗评估工具总体架构

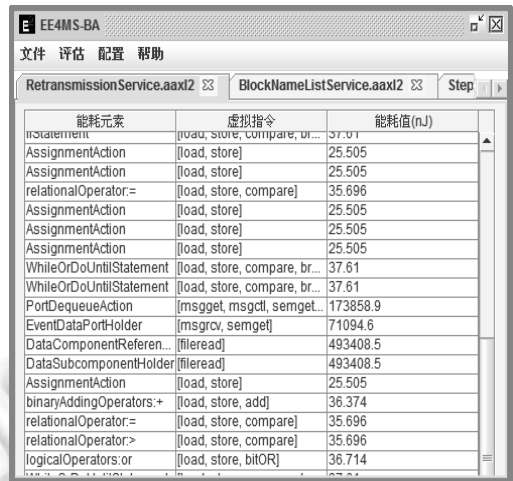


图 4 EET4AADL 能耗评估工具的主界面

UI 层采用 MVC 模式<sup>[15]</sup>,主要实现导入待评估的 AADL 行为模型、输出评估结果等功能界面,图 4 给出了 EET4AADL 工具的输出界面;Domain 层主要实现 AADL 行为模型到 STG 模型的转换、能耗行为元素识别和评估报告生成等业务功能;而 Resource 层主要封装了对 AADL 行为模型、能耗评估模型、STGFile 和评估报告等文件资源的访问和使用.

## 4 案例研究

### 4.1 案例简介

本文所选取案例为一种蓝牙低功耗计步数据同步协议 BLE-Step<sup>[16]</sup>.BLE-Step 协议主要包括重传服务

(retransmission service,简称 RS),获取计步数据块信息列表服务(block name list service,简称 BS),获取特定块计步数据服务(step data by block service,简称 SS)3 个部分,通过这 3 种服务可以实现计步腕表与智能手机之间的低功耗计步数据传输.前期工作已经完成了对这 3 种数据传输服务的 AADL 建模,如图 5 所示给出了 BLE-Step 的 AADL 模型,图 6 给出了重传服务的 AADL 行为描述;同时,我们已经在 iOS 及 Android 平台环境下的程序实现,并已经运用于实际项目中.因此,可以方便地对 BLE-Step 协议进行源程序级和系统设计级的能耗评估和实验对比.

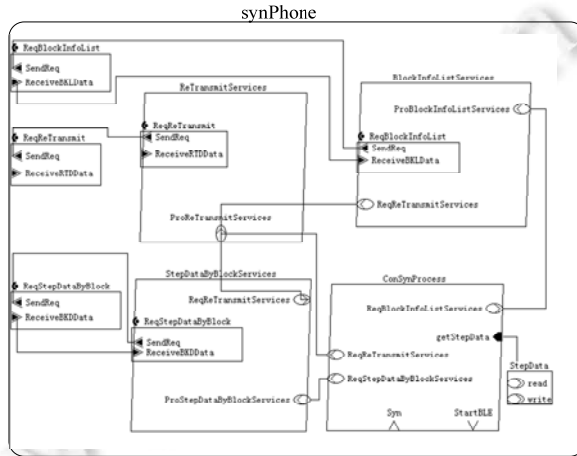


图 5 BLE-Step 简介

### 4.2 实验结果

为验证本评估方法的可行性,本文做了如下对比分析实验:基于源程序级的能耗评估方法评估 BLE-Step 的能耗分别与本文提出的 EE4MS-BA 方法评估的能耗和 Senn 等人<sup>[11]</sup>的方法评估的能耗进行比较,计算误差.本文采用的源程序级能耗评估平台为 EMSIM 功耗仿真平台,这是一款指令级高精度移动系统功耗模拟器<sup>[12]</sup>.

本文构建的评估工具可以得到 AADL 行为模型各组件的能耗总和,并与 EMSIM 和 Senn 等人<sup>[11]</sup>方法的评估结果进行比较,得出表 5 和表 6 的实验结果.其中 EE4MS-BA 为基于 AADL 语言的移动软件能耗评估工具评估的结果.对于重传协议 RS 用源程序级设计的评估工具 EMSIM 评估的结果为 6 875 636.561nJ,用基于设计级 EE4MS-BA 的能耗评估工具评估的结果为 6 588 879.826 nJ,两种工具的误差(EE4MS-BA 设计级与 EMSIM 代码级比较的误差)为 4.17%,而用基于设计级 Senn 等人的能耗评估工具评估的结果为 6 537 962.847 nJ,两种工具的误差(Senn 设计级与 EMSIM 代码级比较的误差)为 4.91%.

EE4MS-BA 设计级与 EMSIM 代码级比较获取计步数据块信息列表服务 BS 和获取特定块计步数据服务 SS 的误差分别为 8.79%和 8.18%,而 Senn 设计级与 EMSIM 代码级比较获取计步数据块信息列表服务 BS 和获取特定块计步数据服务 SS 的误差分别为 8.85%和 8.21%.

获取计步数据块信息列表服务 BS 和获取特定块计步数据服务 SS 的误差明显大于重传服务 RS 的误差,这主要是由于对重传服务行为的描述较为复杂,涉及的能耗行为元素比较精细,因而误差较小.因此,用 AADL 行为附件对系统的行为进行描述时,其系统的内部行为的复杂程度以及行为描述的精细程度直接决定了最后对 AADL 行为模型的能耗评估误差.同时,从这 3 种服务的评估结果可以看出本评估方法的平均误差小于 10%,具有可行性.

因此,本文提出的能耗评估方法 EE4MS-BA 评估的能耗较 Senn 等人<sup>[11]</sup>的与基于源程序级的能耗评估方法所得评估值更为接近,表明本文方法的精度较 Senn 等人的有所提高.

```

annex behavior_specification {**
variables
    t1:base_types::integer;           t: DataType::timeoutdata;
    m:base_types::integer;           m1:base_types::integer;
    finish:base_types::boolean;
        mCount:base_types::integer;
    TimeOut_LostData:base_types::boolean;
        result:DataType::message_type;
states
    init : initial complete state;      com : complete state;
    parse : state;          fin: final state;
transitions
    init -[on dispatch ]-> parse {
        t1:= t * ReTransmitData.RTI.ReTransmitPageCount;
        m1:= m * ReTransmitData.RTI.ReTransmitPageCount;
        SendReq!(ReTransmitData.RTI) };
    parse -[on dispatch ]-> com {      mCount:=1;
        for(i:base_types::integer in
            1..ReTransmitData.RTI.ReTransmitPageCount){
            ReceiveRTDDData?(ReTransmitData.RTD) } };
    if(TimeOut_LostData=true)      finish:=false
    else                            finish:=true
    end if;
    while(finish=false) {
        do { SendReq!(ReTransmitData.RTI);
            for(i:base_types::integer in
                1..ReTransmitData.RTI.ReTransmitPageCount){
                ReceiveRTDDData?(ReTransmitData.RTD) };
                mCount:=mCount+1 }
            until (finish=true or mCount>m1)
        } };
    parse-[]->fin{
        if(finish=true)              result:="Success"
        else                          result:="Failure"
        end if                        };
    com -[on dispatch timeout ]-> init {result:="Failure" }; **};
    
```

图 6 重传服务的行为描述

表 5 EMSIM 和 EE4MS-BA 的实验对比结果

案例	Code-Level (nJ)		Model-Level (nJ)		误差 (%)
	EMSIM		EE4MS-BA		
RS	6 875 636.561		6 588 879.826		4.17
BS	3 514 652.690		3 205 573.868		8.79
SS	9 595 517.194		8 810 501.510		8.18

表 6 EMSIM 和 Senn 方法的实验对比结果

案例	Code-Level (nJ)		Model-Level (nJ)		误差 (%)
	EMSIM		Senn 方法		
RS	6 875 636.561		6 537 962.847		4.91
BS	3 514 652.690		3 203 525.256		8.85
SS	9 595 517.194		8 807 419.652		8.21

## 5 总结与展望

针对目前已有的设计级能耗评估方法大多未对软件构件的内部行为元素进行能耗评估,导致了精度问题,本文提出了一种基于 AADL 语言的移动软件能耗评估方法,并基于该方法开发了一款用于评估移动软件能耗的工具.通过构建的评估模型和本文定义的移动软件能耗评估过程确定移动软件的耗能热点,为未来进行移动软件能耗的优化研究奠定了基础.

下一步的工作将主要针对本方法的评估结果对移动软件的能耗优化提出相应的策略.



## References:

- [1] Cuervo E, Balasubramanian A, Cho D, Wolman A, Saroiu S, Chandra R, Bahl P. MAUI: Making smart phones last longer with code of float. ACM, 2010.
- [2] Sinha A, Chandrakasan AP. JouleTrack: A Web based tool for software energy profiling. ACM, 2001.
- [3] Qu G, Kawabe N, Usami K, Usami K, Potkonjak M. Code coverage-based power estimation techniques for micro processors. Journal of Circuits, Systems, and Computers, 2002,11(5):557-574.
- [4] Kim D, Hong J. ESUML-EAF: A framework to develop an energy-efficient design model for embedded software. Software & Systems Modeling, 2013. 1-18.
- [5] Hu J, Li XD, Zheng GL, Wang CH. Modeling and analysis of power consumption for component-based embedded software. Emerging Directions in Embedded and Ubiquitous Computing. Springer-Verlag, 2006. 795-804.
- [6] Ouni B, Rekhissa HB, Belleudy C. Inter-Process communication energy estimation through AADL modeling. IEEE, 2012.
- [7] Capra E, Francalanci C, Slaughter SA. Measuring application software energy efficiency. IT Professional, 2012,14(2):54-61.
- [8] Liu XB, Guo B, Shen Y, Xiong B, Wang JH, Wu YS, Liu YB. Embedded software energy modeling method at architecture level. Ruan Jian Xue Bao/Journal of Software, 2012,23(2):230-239 (in Chinese with English Abstract). <http://www.jos.org.cn/1000-9825/4145.htm> [doi: 10.3724/SP.J.1001.2012.04145]
- [9] Kim DH, Kim JP, Hong JE. A power consumption analysis technique using UML-based design models in embedded software development. In: Proc. of the SOFSEM 2011: Theory and Practice of Computer Science. Springer-Verlag, 2011. 320-331.
- [10] Yang ZB, Pi Lei, Hu K, Gu ZH, Ma DF. AADL: An architecture design and analysis language for complex embedded real-time systems. Journal of Software, 2010,21(5):899-915 (in Chinese with English Abstract). <http://www.jos.org.cn/1000-9825/3700.htm> [doi: 10.3724/SP.J.1001.2010.03700]
- [11] Senn E, Laurent J, Juin E, Diguët JP. Refining power consumption estimations in the component based AADL design flow. IEEE, 2008.
- [12] Tan TK, Raghunathan A, Jha NK. EMSIM: An energy simulation framework for an embedded operating system. IEEE, 2002.
- [13] Dissaux P, Bodeveix JP, Filali M, Gauflillet P, Vernadat F. AADL behavioral annex. 2006.
- [14] AADL Behavior Annex Marco-Model. 2012. <http://aadl.enst.fr/javadoc/aadlba/1.0.0/>
- [15] Krasner GE, Pope ST. A description of the model-view-controller user interface paradigm in the smalltalk-80 system. Journal of Object Oriented Programming, 1988,1(3):26-49.
- [16] Ni YC, Cong S, Ting Z, Ye P, Wang CY, Zeng L. A profile for step data transmission based on bluetooth low energy. In: Proc. of the IEEE Int'l Conf. on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, 2013.

## 附中文参考文献:

- [8] 刘啸滨,郭兵,沈艳,熊冰,王继禾,伍元胜,刘云本.嵌入式软件体系结构级能耗建模方法.软件学报,2012,23(2):230-239. <http://www.jos.org.cn/1000-9825/4145.htm> [doi: 10.3724/SP.J.1001.2012.04145]
- [10] 杨志斌,皮磊,胡凯,顾宗华,马殿富.复杂嵌入式实时系统体系结构设计与分析语言:AADL.软件学报,2010,21(5):899-915. <http://www.jos.org.cn/1000-9825/3700.htm> [doi: 10.3724/SP.J.1001.2010.03700]



杜欣(1979—),女,新疆石河子人,博士,副教授,CCF 高级会员,主要研究领域为基于搜索的软件体系结构,演化计算,分布式计算.



倪友聪(1975—),男,博士,副教授,CCF 会员,主要研究领域为基于搜索的软件设计.



王晓红(1991—),女,硕士,主要研究领域为基于搜索的软件体系结构.



罗增(1989—),男,硕士,主要研究领域为基于搜索的软件体系结构.