

# 动态数据驱动的交通仿真框架研究与实现

罗永琦 燕雪峰 冯向文 周 勇

(南京航空航天大学计算机科学与技术学院 南京 210016)

**摘 要** 针对现有交通仿真中事先建立的理论模型难以准确预测交通状态发展趋势的问题,提出一种基于动态数据驱动应用系统范型的交通仿真框架。首先在微观仿真模型的基础上建立状态空间模型进行先验状态估计;继而将实测交通数据引入模型以调整和评估状态空间模型;基于交通状态非线性非高斯的特性,选用粒子滤波器,提出并实现了数据同化模型和相关算法,提炼传统粒子滤波器的关键步骤并对其进行改进,以提高状态估计的能力;最后基于微型交通仿真软件 MovSim 实现了上述框架。实验表明:基于该框架的交通状态预测精度得到明显提高,受测量误差和环境噪声的影响小,具有较强的预测稳定性和可靠性。

**关键词** 动态数据驱动,微观仿真模型,状态估计,数据同化,粒子滤波

中图法分类号 TP311 文献标识码 A

## Research and Implementation of Dynamic Data-driven Traffic Simulation Framework

LUO Yong-qi YAN Xue-feng FENG Xiang-wen ZHOU Yong

(School of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

**Abstract** The existing pre-established theoretical models in traffic simulation are hardly able to accurately predict the trend of traffic state development, a traffic simulation framework based on dynamic data-driven application system (DD-DAS) paradigm is put forward. Firstly, a state space model is proposed to conduct a priori state estimate on the basis of microscopic simulation model; then real-time data is dynamically incorporated in the process of simulation to adjust and evaluate the state space model; what's more, given the nonlinear and non-Gaussian characteristic of the application field, particle filter is chosen as the means of implementation of data assimilation models and related algorithms by refining the key step and improving related procedures to make enhancement of the ability of state estimation. Finally, micro-simulation system named Movsim is taken to realize the framework. Experimental results indicate our proposed framework poses a high accuracy of prediction and is less vulnerable to the impact of measurement error and environmental noise, thus, it provides stronger stability and reliability.

**Keywords** Dynamic data driven, Microscopic simulation model, State estimation, Data assimilation, Particle filter

## 1 引言

动态性、开放性和不确定性是交通系统的典型特征,难以建立精确的数学模型来准确预测交通状态。因此,采用仿真方法解决交通问题具有重要的意义。交通仿真<sup>[1]</sup>是研究运用计算机技术建立一个能够代替现实交通系统的模型的过程。传统静态仿真建立模型主要基于线性系统理论方法,包括 Box-Jenkins 模型<sup>[2]</sup>、卡尔曼滤波模型<sup>[3]</sup>、模糊回归模型<sup>[4]</sup>等,该类模型常忽略复杂路况下的突发事件,如车辆故障和关闭车道,无法利用实时数据提高模型预测的精确性。

动态数据驱动应用系统(Dynamic Data Driven Application Systems, DDDAS)<sup>[5]</sup>是一种全新的仿真应用模式,其本质想法是在仿真运行过程中动态加入反映系统真实状态的实时信息并做出响应,提高模型分析预测能力,增强结果的可靠

性。可见,DDDAS 能够弥补传统仿真的时效性差和模型依赖离线历史数据的缺点,实现仿真和实验动态反馈的要求,目前 DDDAS 在交通领域已经有了一些应用成果。Hoffman 等人<sup>[6]</sup>重点研究复杂环境下如何在仿真过程中不断动态控制自适应传感器获得光谱测量数据对模型进行修正以获得更为准确的预测。Xi 等人<sup>[7]</sup>提出扩展亚当斯模型以优化估计信号灯控制下行人在人行横道的平均延误时间。在空中交通领域,Chen 等人<sup>[8]</sup>给出动态数据驱动的航班延误预测方法,并介绍实时数据如何被收集并注入到预测过程得到更可靠的结果。在海上交通领域,曾庆成等人<sup>[9]</sup>提出并设计基于动态数据驱动的溢油应急决策仿真模型,研究模型状态的更新算法,并将其与一般溢油模型对比以证明该模型可以提高溢油事故预测的准确性。Sun 等人<sup>[10]</sup>以扩展卡尔曼滤波为基础,引入序列蒙特卡洛方法对交通状态进行估计。

本文受国防科工局十二五重大基础科研项目(c0420110005),国家自然科学基金(61139002)资助。

罗永琦(1989—),男,硕士,主要研究方向为计算机仿真,E-mail:luoyongqi-35@126.com;燕雪峰(1975—),男,博士,副教授,主要研究方向为复杂系统仿真、多学科虚拟样机技术、分布交互仿真等;冯向文(1992—),男,硕士,主要研究方向为计算机仿真;周勇(1975—),男,博士,副教授,硕士生导师,主要研究方向为人工智能、专家系统、智能推理。

本文在上述研究基础上,首先介绍粒子滤波器[11]的相关理论并提炼关键技术,接着提出基于粒子滤波器的动态数据驱动交通仿真框架,通过拓展典型微观仿真模型[12]并设计框架的主要模块以支持模型的动态调整,研究实际系统数据映射到模型的方法,重点基于特定应用领域对粒子滤波器进行改进实现加噪声和权值计算、重采样操作等算法,最后在上述建模思想和关键方法的基础上建立并实现动态数据驱动的交通仿真框架。

## 2 基本概念及相关算法

粒子滤波器对非线性非高斯状态估计问题具有强大的处理能力,其核心思想是利用一系列加权值的随机采样粒子近似表征后验概率密度函数,是一种改进贝叶斯滤波算法,能解决在处理非线性滤波问题时随机量必须满足高斯分布的约束条件。状态估计被视为一个概率推理过程,就是从得到的观测值  $y_k = (y_1, y_2, \dots, y_n)$  推理出  $k$  时刻的状态变量  $x_k$ , 主要包含两个过程:

1. 预测过程:使用模型状态转移过程和 C-K(Chapman-Kolmogorov)方程得到状态先验概率:

$$p(x_k | y_{k-1}) = \int p(x_k | x_{k-1}) p(x_{k-1} | y_{k-1}) dx_{k-1} \quad (1)$$

2. 更新过程:根据观测值  $y_k$ , 通过贝叶斯定理更新先验概率以得到后验概率:

$$p(x_k | y_k) = \frac{p(y_k | x_k) p(x_k | y_{k-1})}{p(y_k | y_{k-1})} \quad (2)$$

贝叶斯滤波需要复杂积分计算,很难得到后验概率的封闭解析式。因此,现有非线性滤波器采用近似计算方法解决积分问题,利用状态空间中的大量样本点近似逼近估计变量的后验概率分布,将积分问题转换为有限样本点的求和问题。粒子滤波器的基础是序贯重要性采样方法(Sequential Importance Sampling, SIS),该方法通过引入一个已知的且容易采样的重要性概率密度函数(通常符合正态分布)生成大量采样粒子并利用这些加权和逼近求解。则式(2)可简化为:

$$p(x_k | y_k) \approx \sum_{i=1}^N w_k^i \delta(x_k - x_k^i) \quad (3)$$

相应的权值定义为:

$$w_k^i \propto \frac{p(x_k^i | y_k)}{q(x_k^i | y_k)} \quad (4)$$

通常需要对权值进行归一化处理,当采样粒子数目很大时,式(3)可以近似逼近真实的后验概率密度函数。经过归一化后其状态估计为:

$$x_{k, estimate} = \sum_{i=1}^N x_k^i w_k^i \quad (5)$$

然而对 SIS 算法而言,多次迭代后粒子权值方差变大,大量计算累积到多数权值小的粒子上,出现样本匮乏的缺陷。为克服权值退化的现象,引入重采样算法,其基本思想是舍弃权重小的样本,复制权重大的样本。具体过程为:根据粒子集  $[\{x_k^i, w_k^i\}_{i=1}^N}]$  中权值的比对方法,得到新粒子集。综上所述,粒子滤波器的流程如图 1 所示。

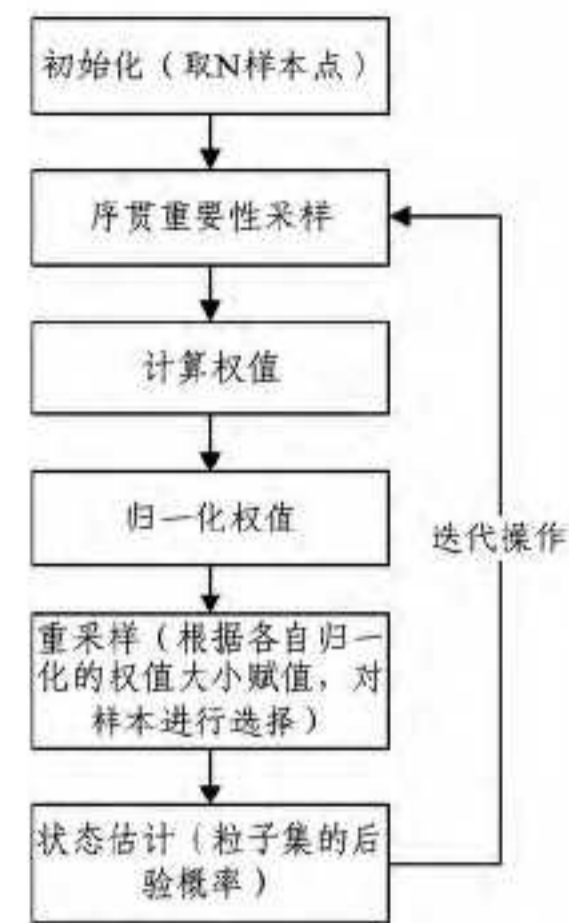


图 1 粒子滤波算法流程

## 3 动态数据驱动的交通仿真框架

构建动态数据驱动应用系统将囊括多个组成部分,应用领域需求和相关特性、系统模型与评估算法、模型与测量模型的耦合、软硬件资源以及运行环境。不同于传统仿真串行化顺序执行的模式,动态数据驱动的模式强调系统的动态适应性,表现在:适应以传感器为主要测量设备的动态数据输入;适应计算需求的变化;适应模型的动态调整和仿真结果的动态反馈。传感器网络的发展和计算机性能的提高使得实时数据的采集和传输成为可能,正是这些反映系统真实动态特性的数据及基于数据的操作为动态数据驱动模式的应用提供了有力支持。

因此,动态数据驱动交通预测框架可以克服基于线性系统理论进行建模的不足和基于知识理论的智能模型参数选择的困难。DDDAS 在危机管理、灾害预测和环境检测等拥有充足数据的领域得到广泛的应用,给框架在交通领域的应用奠定了坚实的理论和实践基础。图 2 给出基于动态数据驱动系统范型的交通仿真系统框架。

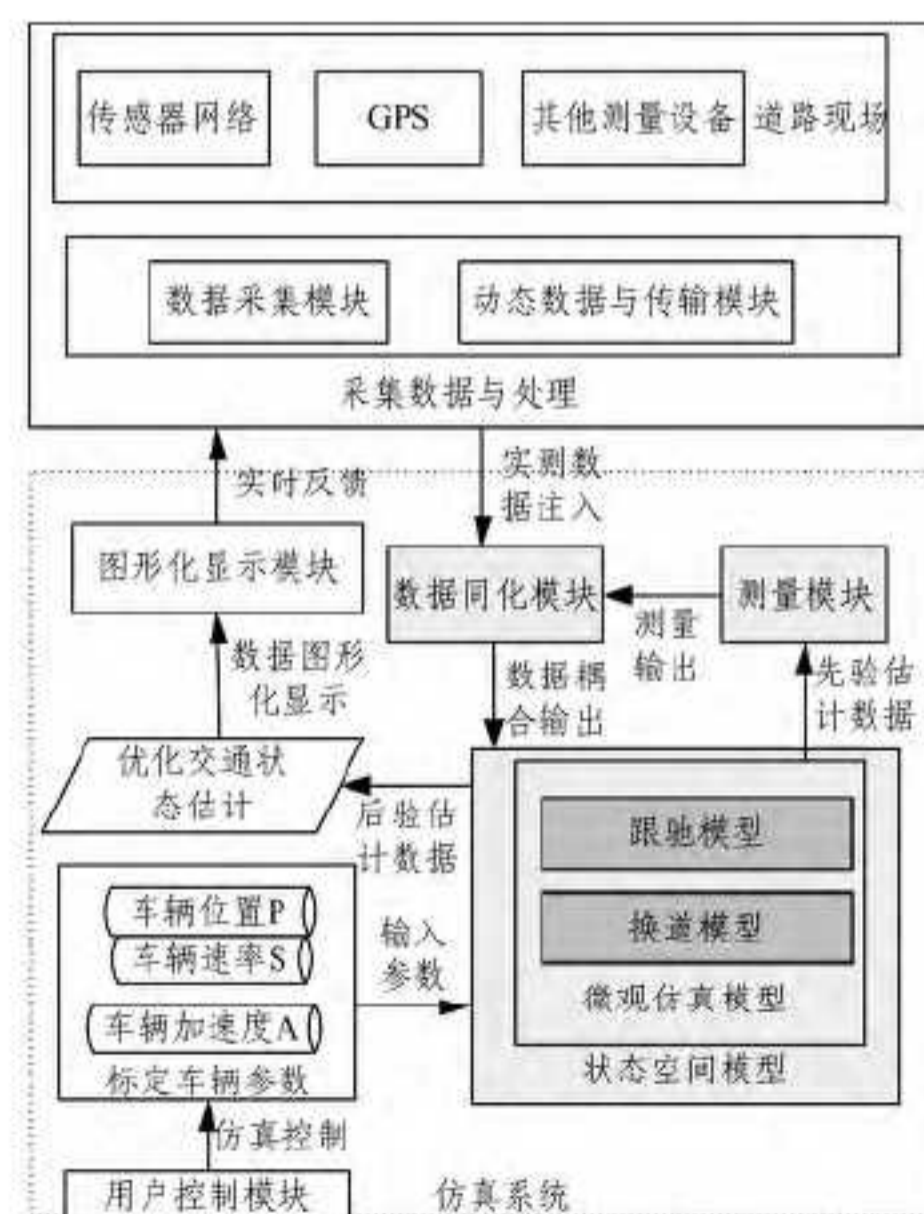


图 2 动态数据驱动交通仿真框架

从动态数据驱动应用系统应用的相关描述看,设计的框架符合构建 DDDAS 的要求,仿真模型可同时利用历史数据

与实测数据对交通状态进行仿真与预测。首先,标定仿真模型参数,交通参数是进行交通状态判别的基础,利用大量的历史数据选定的参数变量为车辆速度、加速度以及位置。接着,构建状态空间模型,通过现有的微观交通仿真模型中各个车辆的状态参数转移方程辅以静态的路网模型建立描述车辆状态变量变化规律以及之间的相互作用的模型,它是交通状态预测的基础。然后,以基于动态数据驱动的数据融合为主线,在仿真过程中动态加入车辆实测数据并研究实测数据到仿真模型的映射方法调整和评估状态空间模型。其中测量模块完成将实测数据与仿真模型建立映射的功能,它将实时数据中具有可比性的数据进行抽离并辅以一定的转换,使得系统能够处理各类数据实现多个映射。而数据同化模块的目的是用实际测量的结果来更新模型预测的结果,本质是将观测数据和模型预测通过某种方法结合得到更加客观的分析。显然,数据同化模块是整个框架的核心部分,也是 DDDAS 的本质体现,结合粒子滤波器的理论介绍将其作为同化模块的实现方法,其关键技术及算法分析在下节介绍。最后将融合数据作为状态空间模型的输入进一步进行交通状态优化预测,提高模型预测精确性。由于测量误差和环境噪声的存在,且框架的主要模块处理涉及大量的计算,因此,选用高性能计算机完成计算任务,同时将框架搭建在多计算机运行环境下实现并行处理。

#### 4 动态数据驱动的交通仿真数据同化及算法

##### 4.1 数据映射算法

由于状态空间模型输出数据无法与实时监测数据建立直接映射关系,在仿真系统中需要设计数据映射算法。典型的微观模型主要是跟驰模型,在本文中特指智能驱动模型 IDM (Intelligent Driver Model)<sup>[13]</sup>。该模型在空间上进行离散化,将道路每隔一定距离进行分割作为运行基本结构单元,车辆流运行过程由若干路段运行组构成,且所有车辆共享道路这个临界资源,以传感器覆盖的某个路段距离  $S_d$  为例介绍数据映射算法。道路表面铺设速度和位置传感器分别对车辆进行测速和测距。假设某个时刻  $t$ ,传感器监测的速度序列为  $S_t = (S_1, \dots, S_n)$ ,位置序列为  $P_t = (P_1, P_2, \dots, P_m)$ ,其中  $m > n$ 。数据映射算法如下:

1. 该路段所在的区域记为  $k[a, b]$ ,遍历位置序列根据区域的临界值  $a$  和  $b$  确定此路段区域内的车辆个数,即确定速度序列  $S_t$  的维数  $n$ ,需要注意的是此位置序列记录的为车辆车头处的位置,若需要车辆重心位置则需要根据车辆属性设置位置偏移量;

2. 选取平均速度作为表征交通状态的映射变量,定义检测数据序列到测量值的映射为:

$$SM = \frac{1}{n} \sum_{id=1}^n S_{id} \quad (6)$$

其中,  $SM$  代表该路段车辆平均速度,  $S_{id}$  为速度序列中的车辆号为  $id$  的车的速度;

3. 根据道路总长度和采样间隔确定分段数,各个路段均按式(6)求得平均速度。

##### 4.2 自适应平移噪声算法

加噪声是指对初始化操作后的粒子的车辆位置、速度和加速度添加不确定性因素,使得粒子间存在差异性,增强系统

样本多样性。在经典粒子滤波算法中,系统噪声通常采用的是高斯白噪声,其方差固定,然而通过观察实际车辆的运行情况发现在车速较大的情况下,车辆间隔较大,其添加噪声的空间比较充裕,此时,可对其添加较大的速度和位置噪声;相对而言,在车速较小的条件下,后续车辆由于前行车辆降速而被迫减速以避免碰撞,车辆间距变小,其可添加噪声的空间变小,甚至当两者相距超过撞车的最小安全距离时不对其添加噪声。因此设计自适应平移噪声算法,其添加位置噪声如下:

$$\theta_n(t) = s(t, v_1, v_2, \rho)U \quad (7)$$

其中,式(7)为粒子  $n$  的粒子在时刻  $t$  添加的噪声量,  $U$  为服从  $0 \sim 1$  上的均匀分布的随机数,  $s$  为通过设定的速度临界值,  $v_1$  和  $v_2$  为确定的位置偏移量  $\rho$  的函数。相应的速度噪声和加速度噪声均可以通过速度和加速度乘以相应的服从均匀分布的随机数系数后加自身原有值来添加,其速度求解公式如下:

$$v_n(t) = v(1 + U_1 R_{v_0}) \quad (8)$$

式(8)代表粒子  $n$  在  $t$  时刻添加速度噪声后的求解函数。其中,  $U_1$  为服从  $0 \sim 1$  上的均匀分布不同随机数,其后为速度固定系数值,根据实验进行调整。加速度做类似求解。

##### 4.3 权值计算算法

确定权值函数主要是合理利用实测数据和测量模块的测量值设定衡量粒子优劣的标准,给准确反映状态信息的粒子赋予较大权值,给含有较大误差且不能反映状态信息的粒子赋予较小权值。传统粒子滤波器采用正态分布函数作为权值分布的概率密度函数,无法突出展现大噪声和小噪声粒子的权值差别,不能很好地利用当前的观测值对不同程度的噪声粒子加以区分。鉴于此,采用更能明显反映粒子区别的反比例函数作为概率密度函数对算法进行改进,提出基于平均速度的智能传感器权值计算算法。假设  $N_s$  为每隔一定采样距离均匀布设传感器的数量,  $X_j (j=1, 2, \dots, N_s)$  表示第  $j$  个传感器的位置,  $R_j(t_k)$  表示在  $t_k$  时刻的第  $j$  个传感器测得的速度数据,  $V_j(t_k)$  表示在  $t_k$  时刻的第  $j$  个传感器的平均速度,  $weight[t_k, i]$  表示第  $i$  个粒子  $t_k$  时刻的权重。其算法流程如下:

1. 根据实测数据确定拥堵区域, For  $i=1, \dots, N_s$ , 由  $R_j(t_k)$  计算  $V_j(t_k)$ , 设置状态门限值  $V_{thread}$ , IF  $V_j(t_k) < V_{thread}$  即为拥堵区域,记  $[X_i, X_{i+1}]$  为确定的拥堵区域之一,考虑交通系统链式反应特性,将  $[X_i, X_{i+1}]$  前  $1/4$  路段和后  $1/8$  路段作为选定的拥堵区域。

2. 求得当前时刻的经过加噪声处理后的各个粒子在选定区域内的平均速度,分别与步骤 1 的该区域内的实测平均速度相减求得其绝对值差,需要注意的是两者的绝对值差为 0,且此时该区域的车流速度不为 0 时,赋予权值时默认将其设置为 1。

3. 设定权重比例惩罚因子,通常情况下,离故障发生距离越近,其在计算权值时占据的权重比例越大,反之,其比重越小。所以在计算权重时对拥堵区域进行划分,其前  $1/4$  路段求绝对值差后对平均速度差乘以低权重惩罚因子  $lowWeight$ , 中间路段速度差乘以高权重惩罚因子  $highWeight$ , 其余乘上中权重惩罚因子  $middleWeight$ 。

4. 利用迭加原理的每个粒子的平均速度差乘上相对应的惩罚因子后进行求和,记为  $Y_{threadnum}$ ;

5. 采用反比例函数为步骤 4 赋予权重, 具体为  $weight[t_k, i] = 1 / (Y_{threaddum})$ 。

通常需要将各粒子计算的权值进行归一化操作。其目的是将需要处理的数据统一到同一参考系下, 方便对数据进行处理。其基本操作是将各个粒子的权值除以所有粒子的权值之和, 作为归一化后的权值  $norm[t_k, i]$ , 即  $norm[t_k, i] = weight[t_k, i] / sum(weight[t_k, i], i)$ 。

#### 4.4 重采样算法

重采样算法是解决粒子权值退化问题、增强系统粒子多样性的方法, 其根本思想是复制权值大的粒子, 舍弃权值小的粒子。结合特定的应用领域采用层次重采样的方式, 每一次重采样过程中, 权值较大的粒子只对其区域的局部粒子有影响, 这既能在仿真过程中保持粒子的多样性, 又能挑选出足够的较优粒子。其具体过程如下:

1. 划分采样空间, 设置  $q(0) = norm(t_k, 0), u(0) = 1/N$ ; For  $i=1, \dots, N-1, q(i) = q(i-1) + norm(t_k, i), u(i) = u(i-1) + u(0)$ , 即  $u$  为均匀等分空间,  $q$  为按粒子权重大小划分空间。

2. 设置变量  $count = 0$ , For  $i = 0, \dots, N-1$ , while ( $q(count) < u(j)$ ) 则  $count++$ ,  $tempState(j) = TrafficState(t, count)$ , 此步骤为分段赋值, 其中  $TrafficState(t_k, count)$  为第  $count$  个粒子的  $t_k$  时刻的状态, 用临时变量  $tempState(j)$  记录第  $j$  个粒子的此刻的状态。

3. 为  $N$  个粒子赋最新状态 For  $i=0, \dots, N-1, TrafficState(t, k) = tempState(k)$ 。

4. 由于重采样操作计算复杂度较高, 设定阈值  $RT$ , 当有效样本个数小于阈值时进行重采样操作。通常情况下计算有效样本式如下:

$$N_{eff} = \frac{1}{\sum_{i=1}^N [norm(t_k, i)]^2} \quad (9)$$

## 5 实验及结果分析

为了验证 DDDAS 的工作机制及提出框架的有效性, 采用微观仿真系统软件 MovSim 为平台进行框架的验证和实现。实验静态参数设置如下: 初始速度  $V_0 = 24m/s$ ; 道路长度  $RLen = 1000m$ ; 车辆流率  $Q = 1500$ ; 车辆长度  $Vlen = 6m$ , 车辆间安全距离  $VSafe = 5m$ ; 仿真间隔  $timeStep = 0.2s$ ; 粒子数目  $N = 100$ 。实时数据通过仿真拥堵状态并采用 Json 数据格式保存每一仿真时刻车辆号、速度和位置信息来模拟。实验过程如下:

1. 运行原始仿真, 记录每个仿真步长的仿真数据 DATA1, 作为原始对比数据;

2. 模拟拥堵现象, 在距终点  $1/5$  处关闭一条车道, 运行原始仿真, 记录每个仿真步长的仿真结果 DATA2, 作为实测数据 SENSOR DATA;

3. 运行原始仿真, 初始化  $N$  个粒子, 对每个粒子执行加噪声、权值计算、归一化和重采样算法, 记录每个粒子的仿真步长数据为 DATA3<sub>i</sub>。

采用 Swing 技术进行图形化显示, 可以通过设定不同的时刻和粒子号看到相对应的车辆密度曲线。典型粒子车辆密度曲线如图 3 所示。

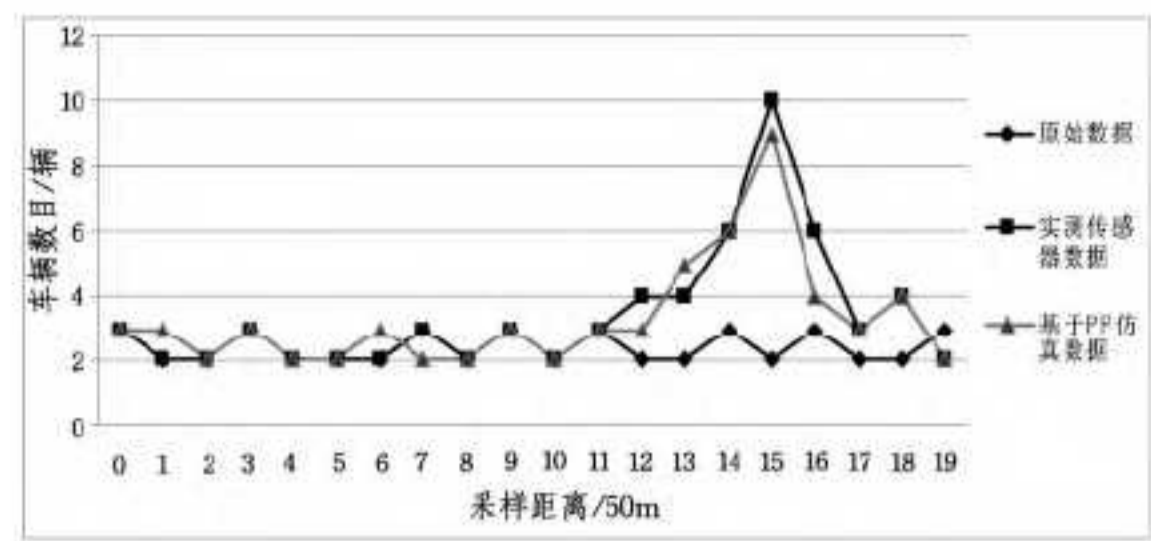


图 3 时刻 120 车辆密度曲线

在堵车点 800m 周围区域, 采用 PF 框架的仿真结果明显受到实测数据 Sensor data 的影响, 产生了明显的堵车路况。堵车程度略微小于观测数据, 但远大于未采用 PF 框架的仿真结果。可见, 动态数据驱动的交通状态预测值与实际测量值的误差相对较小, 预测效果要明显优于先验估计值。同时, 随着时间的推移, 交通量预测值与实际测量值的误差更小, 预测精确度更高, 其预测的波动性并未随着时间的推移产生较大的影响。可见基于动态数据驱动范型的框架具有较强的预测稳定性和可靠性。

结束语 本文基于动态数据驱动应用系统范型提出一种动态数据驱动的交通预测框架, 该框架是一种全新交通状态预测方法, 在利用状态空间模型进行预测的同时, 车辆运行过程的各种实时数据被采集并馈送到预测过程中, 用于动态地调整先验预测值。实验表明所提出的动态数据驱动的交通仿真框架有很高的预测精度, 为交通部门预测未来交通的变化趋势, 减少交通拥堵的状况和相关决策的制定提供了有力支持。接下来的工作主要沿以下方面开展: (1) 优化框架来提高动态预测的精确性; (2) 探索多种噪声添加方法, 使粒子间区分度变大。

## 参考文献

- [1] Wang T, Tang S, Pang P. 3D urban traffic system simulation based on geo-data[C]// 2nd International Conference on Information Technology, Research and Education, 2004. IEEE, 2004: 59-63
- [2] Box G E P, Jenkins G M, Reinsel G C. Time series analysis: forecasting and control (4th Edition)[M]. Beijing: Mechanical Industry Press, 2011
- [3] Huang Y, Xu L, Luo Q, et al. Urban expressway travel time prediction method based on fuzzy adaptive kalman filter[J]. Appl. Math, 2013, 7(2): 625-630
- [4] 赵建有, 周孙锋, 崔晓娟, 等. 基于模糊线性回归模型的公路货运量预测方法[J]. 交通运输工程学报, 2012, 12(3): 80-85
- [5] Darema F. Dynamic data driven application systems[R]. NSF Workshop on Dynamic Data Driven Application Systems, March 2000
- [6] Uz Kent B, Hoffman M J, Vodacek A, et al. Feature matching and adaptive prediction models in an object tracking DDDAS[J]. Procedia Computer Science, 2013, 18: 1939-1948
- [7] Xi H. DDDAS-based multi-scale framework for pedestrian behavior modeling and interactions with Drivers[C]// Proceedings of the Winter Simulation Conference. 2012: 370-371
- [8] Chen H, Wang J, Feng L. Research on the dynamic data-driven application system architecture for flight delay prediction[J]. Journal of Software, 2012, 7(2): 263-268

(下转第 480 页)

同,如果以确定的 CPU 的向量化宽度进行优化,则可能对 MIC 端程序性能产生负面影响。

#### b) 访存方式

访问 cache 的速度远大于内存,因此需要利用局部性原理,尽量集中处理相关的数据。顺序访问的速度远大于随机访问,如果不能满足顺序访问,需要调整访问顺序或数据排布顺序,尽量达成顺序访问。计算与访存交叉,运行时间可以互相掩盖,减少总体运行时间。

#### 9) 开发 MIC 端并行版本

基于 CPU 端并程序,开发 MIC 端并行版本。在需要移植到 MIC 端的代码段前添加 offload 语句,表示将代码段卸载到 MIC 协处理器中运行,并使用 in、out、inout 标示传输的数据、传输的长度、传输的方向,对 offload 代码中用到的函数使用 `__attribute__((target(mic)))` 进行标识。由于篇幅有限,具体语法请参照相关文档。

在此版本的测试过程中,由于基础版本已在 CPU 端通过测试,因此此处测试应偏重于 offload 语句以及传输的正确性。通过先期在 CPU 端的开发和测试,减少了此处的工作量,避免了使用协处理器调试程序的困难,提高了工作效率。

#### 10) 优化 MIC 端并行版本

MIC 设备与 CPU 设备虽然相似,但仍有一定的不同之处,因此 MIC 平台程序需要进行针对性的优化。主要优化点有:

##### a) 并行线程数调整

根据实际测试情况,选择合适的并行线程数,由于 MIC 平台硬件设计,理论上每核心至少运行 2 个线程,通常将并行线程数设置为 MIC 每核心 3 或 4 个线程。

##### b) 向量化调整

由于 MIC 平台向量化长度大于 CPU 平台,因此应针对 MIC 平台的 512bit 向量化宽度进行针对性的优化。虽然在 CPU 版本优化过程中已针对性地做了一定优化,但仍需在真实硬件环境下进行相应的测试和调整。

##### c) 数据传输优化

对 CPU 和 MIC 之间的数据传输进行优化,包含减少不必要的数据传输,计算与传输异步执行等。对一部分应用而言,可能需要对某些非热点函数进行移植,虽然降低了该函数的计算效率,但减少了数据传输时间,进而提高了程序整体的运行效率。

如果热点部分的移植不能满足需求,可以采用 CPU+

MIC 协同计算的方式。即采用多线程或多进程的方式,部分进程/线程使用 CPU 进行热点计算,部分进程/线程控制 MIC 协处理器进行热点计算,以充分利用计算资源。如果仍然无法达到要求,可以将程序改为集群版本。即采用 MPI 等通信方式,将程序扩展为多节点应用,每个节点内使用 CPU 与 MIC 协同计算的方式,充分利用多节点计算资源。

结束语 本文在介绍多核处理器和协处理器发展趋势和 MIC 架构的基础上,对 MIC 应用模式进行了介绍和比较,详细描述了 offload 模式,提出了 offload 模式的适用范围,并提出了一种基于 offload 模式的开发策略。依照本文提出的开发策略,可以快速移植开发基于 offload 模式的应用,减少调试、优化的成本,提高开发效率。

能够看出,MIC 作为新推出的并行计算架构,拥有并发度高、学习曲线平缓等特点。可以得出结论,基于 MIC 平台的 offload 编程模式可以与传统 CPU 平台并行方式互为补充,进一步提升硬件并发处理能力,为并行计算应用带来低迁移成本的性能提升。

## 参考文献

- [1] 王思东,张清,等. MIC 高性能计算编程指南[M]. 北京:中国水利水电出版社,2012
- [2] 唐天兵,谢祥宏,等. 多核 CPU 环境下的并行遗传算法的研究[J]. 广西大学学报:自然科学版,2009,8(34):546-550
- [3] 陈国良. 并行计算:结构、算法、编程[M]. 北京:高等教育出版社,2003(8)
- [4] Akher S, Roberts J. 多核程序设计技术:通过软件多线程提升性能[M]. 李宝峰,富弘毅,李韬,译. 北京:电子工业出版社,2007(3)
- [5] Nvidia. NVIDIA CUDA programming guide[OL]. <http://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html>,2013
- [6] Sanders J, Kandrot E. CUDA by example: an introduction to general-purpose GPU programming[M]. Addison-Wesley,2011
- [7] Jeffers J, Reinders J. Intel Xeon Phi Coprocessor High Performance Programming[M]. morgan kaufmann,2013
- [8] Intel. Intel Xeon Phi system software developers guide[OL]. <http://software.intel.com/sites/default/files/article/334766/intel-xeon-phi-systemssoftwaredevelopersguide-0.pdf>,2013
- [9] Intel. Intel C++ Compiler XE 13.1 User and Reference Guide [OL]. <http://software.intel.com/sites/products/documentation/doclib/stdxe/2013/composerxe/compiler/cpp-mac/index.htm>,2013
- [10] Horowitz S R. Highway traffic state estimation using improved mixture kalman filters for effective ramp metering control[C]// Proceedings of 42th IEEE Conference on Decision and Control. 2003:6333-6338
- [11] Liu J S, Chen R. Sequential Monte Carlo methods for dynamic systems[J]. Journal of the American statistical association, 1998,93(443):1032-1044
- [12] Suh W, Henclewood D, Greenwood A, et al. Modeling pedestrian crossing activities in an urban environment using microscopic traffic simulation[J]. Simulation,2013,89(2):213-224
- [13] Godbole V. Intelligent Driver Mobility model and traffic pattern generation based optimization of reactive protocols for vehicular ad-hoc networks[J]. International Journal of Information and Network Security (IJINS),2013,2(3):207-215

(上接第 462 页)

- [9] 曾庆成,杨路燕. 基于动态数据驱动的港口水域溢油仿真模型与算法[OL]. <http://www.paper.edu.cn/releasepaper/content/201301-475>,2013-01-10
- [10] Horowitz S R. Highway traffic state estimation using improved mixture kalman filters for effective ramp metering control[C]// Proceedings of 42th IEEE Conference on Decision and Control. 2003:6333-6338
- [11] Liu J S, Chen R. Sequential Monte Carlo methods for dynamic