

移动粒子半隐式方法的计算效率优化

孙中国, 梁杨杨, 席光

(西安交通大学能源与动力工程学院, 710049, 西安)

摘要: 针对无网格法计算效率普遍不高的局限性, 以移动粒子半隐式(MPS)方法为例, 将计算耗时较多的程序模块作为优化对象, 以降低计算等级和减少计算次数为优化手段, 从搜索算法、数学运算和计算实现形式3个主要方面进行了改进. 提出了用低级运算来代替高级运算、用方位比较来代替距离比较的位置比较消冗法来优化搜索算法, 消除了大部分无效计算, 大大降低了计算的复杂程度和计算量, 使得该方法的计算效率得到显著提高, 增强了MPS方法的应用范围和工程时效性. 该算法具有非常好的普适性, 容易在类似算法中移植, 还有随着计算规模的增大优化效果更加明显的特点. 同时, 发现多种优化方式的组合使用并非效果叠加, 需考虑它们的兼容关系.

关键词: 无网格法; 移动粒子半隐式方法; 计算效率; 搜索算法

中图分类号: O351.2 **文献标志码:** A **文章编号:** 0253-987X(2012)05-0001-06

Optimization Study on Computational Efficiency of Moving Particle Semi-Implicit Method

SUN Zhongguo, LIANG Yangyang, XI Guang

(School of Energy and Power Engineering, Xi'an Jiaotong University, Xi'an 710049, China)

Abstract: The moving particle semi-implicit (MPS) method was improved to raise the computational efficiency of meshless methods from the search algorithm, the mathematical manipulation and the calculation execution. In the improvement, the program modules which consume the most computing time in the calculation were regarded as the optimization objects, and reducing the calculation level and the calculation amount was viewed as the optimization means. A new search algorithm was proposed to eliminate most of the useless calculation and reduce the complexity and the calculation amount by adopting low level mathematical manipulations and the position comparison. The results show that the MPS method improved has higher computational efficiency and wider application in engineering. It is suggested that the compatibility between different optimization methods should be considered because their combination may not yield the superposition of effects.

Keywords: meshless method; moving particle semi-implicit method; computational efficiency; search algorithm

移动粒子半隐式(MPS)法^[1-2]作为近年来发展迅速的一种无网格法, 以其打破节点之间的固定拓扑结构、适用于任意大变形流动问题的显著特点

(相对于传统网格法), 成为研究的热点之一并得到了广泛的应用^[3-7]. MPS法是一种基于配点法的无网格粒子法, 计算区域及边界由一系列粒子进行离

收稿日期: 2011-08-26. 作者简介: 孙中国(1981—), 男, 讲师; 席光(通信作者), 男, 教授, 博士生导师. 基金项目: 国家自然科学基金资助项目(51106125); 高等学校博士学科点专项科研基金资助项目(20090201110059); 中央高校基本科研业务费专项资金资助项目.

网络出版时间: 2012-02-25

网络出版地址: <http://www.cnki.net/kcms/detail/61.1069.T.20120225.0850.001.html>

散表达,计算过程中这些流体粒子既满足连续性方程又在动量方程的作用下变换速度和位置,通过时间层推进的方式逐步更新粒子的流动信息来描述非定常流动的整个过程. 具体算法及表达式可参考文献[1-11],本文不做详述. 该方法计算条理清晰,但是计算过程相对比较烦琐,计算效率也并不是很高. 例如,在每个计算时间层内,算法都需要对参与计算的所有粒子重复性地进行粒子数密度的遍历计算,尤其是在物理模型较为复杂和粒子数目庞大的情况下,这一过程需要花费较长的时间,时间层推进缓慢,使得整个计算过程时间往往非常长,超出工程应用可以接受的范围. 因此,对算法要进行必要的改进和优化,尽量减少计算冗余,提高计算效率,这对 MPS 法具有非常重大的意义.

目前,比较常用的搜索优化算法有 Monaghan 建立的关联链表(Link-List)技术^[12]和 Hemquist 等开发的运算量树状链表 NNPS 技术^[13]等,这些方法通称为动态搜索方法;国内的于志安^[14]等也对搜索算法优化进行了相关研究. 上述方法通过动态存储或划分搜索区域,在一定程度上减少了计算机 CPU 的计算量,提高了计算效率,但同时大量链表结构数据的存储和频繁更新也大大增加了内存的存储量. 不仅如此,有些技术在实际应用中操作起来没有统一的标准,对较复杂的问题,经验性占了较大的比重,优化效果并不如预期的那样明显,也就是优化的普适性相对较弱. 本文将对 MPS 法从多方面(包括搜索算法)进行改进和优化,以提高其计算效率.

1 优化模式及环境

1.1 优化的程序范本

经分析,MPS 法中最耗时的计算过程为遍历搜索(或粒子数密度计算)过程,而在整个程序中有多个模块含有这一计算过程,本文采用模块优化和测试的方式,以二维溃坝模拟(12 225 个粒子参与计算)程序为程序范本,将耗时较多的程序模块作为优化重点进行考察研究,如表 1 所示. 其中模块 3 并不

包含遍历搜索过程,在此仅作为参考比较对象.

表 1 MPS 法待优化程序模块

模块	名称	耗时/s	程序功用
1	density	41	粒子数密度计算
2	function	70	Poisson 方程系数计算
3	iccg	2	Poisson 方程求解
4	gradient	71	压力梯度求解
5	laplace	70	扩散项求解

1.2 优化的思路及检测工具

程序模块本身的计算效率、程序整体有效计算比率和计算机的执行效率是直接关系到算法实现效率的 3 个重要因素,也是本文优化的主要目标. 其中,为了提高模块计算的效率,本文以数学运算为对象,采用降低运算等级为主要优化手段;提高有效计算比率方面则从改进搜索算法入手,建立新的搜索规则,去除多余无效的计算步骤;最后,借助计算机软硬件的发展,采用并行计算方式来提高程序的执行速度. 具体优化目标、对象和手段如图 1 所示.

研究计算效率的检测手段是在进行相同运算时记录运算所进行的时间,通过运行时间的比较来显示程序运行效率的高低. 本文使用 Fortran90 版本程序语言,采用 system_clock()函数作为基本研究手段来进行时间记录和测量. 定义时间数组 $t(0:n)$,其中 n 为需要测量的时间点数,在需要进行测试的程序段前和段后调用该时间函数 call system_clock($t(i)$)和 call system_clock($t(i+1)$),读取程序运行到 $t(i)$ 和 $t(i+1)$ 时刻的计算机系统时间,计算该段程序花费的时间 $\Delta t=t(i+1)-t(i)$,时间计量单位为 s($t>1$ s)或 ms($t<1$ s).

1.3 计算机软硬件环境

本文计算所用的计算机为戴尔工作站,运行环境为 Windows 下的 Fortran 语言编译器,具体硬件性能参数及运行环境如表 2 所示. 其中,在进行数学运算优化和搜索算法优化的过程中使用了比较基础的编译环境1,在进行计算形式优化时,由于要使用

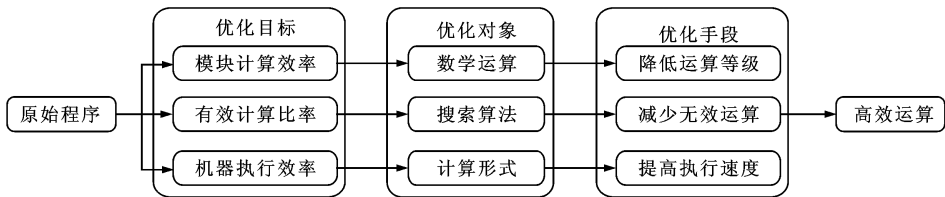


图 1 计算效率优化的目标、对象及手段

OpenMP 并行运算环境,采用了比较高级的编译环境 2,相关运行结果将进行横向比较.

表 2 计算机软硬件环境

计算机型号	Dell Precision T5400
中央处理器	Intel Xeon E5430 8×2.66 GHz
内存	8.00 GB
硬盘存储器	HITACHI 300 GB/Ultrastar SAS, 15 000 r/min
操作系统	Microsoft Windows XP Professional×64 Edition SP2
编译环境 1	Microsoft Developer Studio Fortran PowerStation 4.0
编译环境 2	Microsoft Visual Studio 2005 8.0.5 Intel Fortran Compiler Integration

2 计算效率优化

2.1 数学运算优化

在采用计算机进行数学运算的时候,浮点运算的时间要高于整数运算的时间,乘除法高级运算的时间要高于加减法等简单运算的时间.一般情况下,有 $t_{\text{整数}} < t_{\text{浮点}}, t_{\text{加减}} < t_{\text{乘除}} < t_{\text{乘方}} < t_{\text{开方}}$. 因此,在程序编制过程中,要尽可能减少浮点运算及乘除和开方等高级运算出现的次数. 在本文实验用计算机上测试 Fortran 语言里上述运算分别运行的时间,在变换计算数值的情况下,通过对上述运算都进行 10^{10} 次相同运算并对多次实验结果求平均值,整型数据的加减和乘除分别耗时 22 s 和 24 s,实型数据的加减和乘除分别耗时 28 s 和 29 s. 这说明在计算机中乘除运算较加减法复杂,耗时略高,而就数据类型而言,实型由于存储相对较复杂也占用了较多的时间. 对于乘方和开方运算,这一耗时分别为 60 s 和 283 s. 对于实型的加减、乘除、乘方和开方运算,它们消耗计算机资源的时间比大致为 1 : 1 : 2 : 9.5. 这说明降低高级运算,尤其是乘方和开方的次数,将对提高计算效率非常有帮助,也是优化的重要对象.

MPS 法中计算粒子数密度的遍历过程要实现距离的计算和比较,其计算复杂度为 $o(N^2)$,其中 N 为参加计算的粒子数目,随着粒子数目的增加其复杂程度将会以指数递增,计算量异常巨大. 因此,通过降低运算的等级来降低这一过程的计算复杂度,将会大大改善 MPS 法的计算效率.

在粒子数密度的计算过程中,要先计算粒子之间的距离并判断是否在截断半径之内,然后再进行下一步的运算,原粒子间距的计算方程为

$$r = \left((x - x_0)^2 + (y - y_0)^2 \right)^{1/2} \tag{1}$$

在程序语言中直接表达的基本算法为

$$r = \text{sqrt}((x - x_0) ** 2 + (y - y_0) ** 2) \tag{2}$$

式中:sqrt()为平方根函数;**为乘方运算符. 乘方运算在这里就可以进行运算降级处理,变成乘法运算. 方程(2)可改写为

$$r = \text{sqrt}((x - x_0) * (x - x_0) + (y - y_0) * (y - y_0)) \tag{3}$$

此时,再通过程序运行时调用 system_clock() 函数来计算运行时间. 结果表明,各个模块的运算时间有大幅度的缩短,如图 2 中改进 1 所示. 改进后运算时间的降低幅度可达到 87%左右.

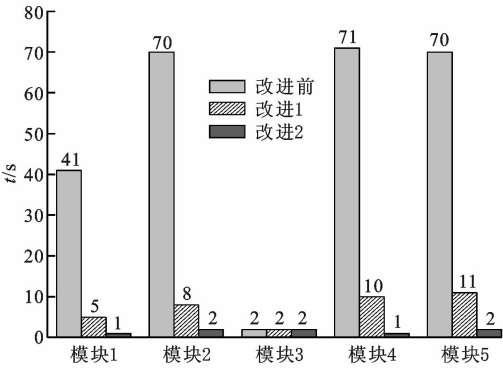


图 2 各程序模块消耗的计算时间比较

2.2 搜索算法优化

通常来讲,计算量基本由两个因素决定,一个是构成计算的基本单元的复杂度,另一个就是这些基本单元要重复使用的次数. 2.1 节中将基本运算进行降级的过程实际上是对基本单元的复杂程度进行了降级优化,本小节则着手通过减少复杂运算的次数来提高计算效率,即减少距离计算的次数,以此来降低计算成本较高的开方运算的次数.

目前程序中所采用的搜索方法,每一步都要按照式(3)先对整个搜索区域内的所有目标粒子 (x_i, y_i) 与周围粒子 (x_j, y_j) 之间的距离进行计算,然后再与截断距离进行比较,如图 3 所示. 假设参与计算的粒子数目为 N ,每个粒子进行粒子数密度计算要进行 $N-1$ 次距离计算,那么在整个搜索区域完成计算,需要进行 $N(N-1)$ 次开方运算,在这里可以认为计算的复杂度为 $o(N^2)$.

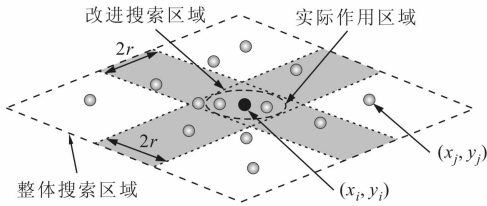


图3 搜索算法优化遍历粒子区域分类图

本文采用了新的搜索算法——位置比较消冗法 (Position Comparison to Eliminate Redundancy, PCER),即用先位置比较再计算距离的形式来代替先计算距离再进行比较的过程.顺序的颠倒将会使计算量发生非常大的变化.如图3所示,在计算目标粒子 (x_i, y_i) 与周围粒子 (x_j, y_j) 之间的距离之前,先计算它们在 x 和 y 坐标上位置差值的绝对值 $|x_i - x_j|$ 和 $|y_i - y_j|$;然后,与截断半径 r 进行大小的逻辑判断 $|x_i - x_j| < r, |y_i - y_j| < r$.如果两个不等式都不成立,说明该粒子位于两条灰色带状区域之外,与目标粒子间的距离绝对大于 r ,没有可能与目标粒子产生关系,可以不进行距离计算;如果两个不等式分别成立,说明该粒子的位置位于两条灰色的带状区域中,可能位于截断半径内,与目标粒子产生关系;如果两个不等式同时成立,该粒子则位于以目标粒子为中心,边长为 $2r$ 的正方形区域里,即图中所示的改进搜索区域.这个区域与粒子实际作用区域(如图3的圆形区域)相比,面积上相差无几,也就是说改进后的搜索区域已经达到了极限小的程度.随后的计算仅对位于改进搜索区域内的粒子进行距离计算.

由于该区域的面积要远远小于整体搜索区域,也就是说该区域内的粒子数目也按比例少很多,开方运算的次数也相应地减少.从定量的角度来讲,每个时间层内,开方运算的次数大致减少为 NM ,其中在二维情况下, M 约为12~15.由于 $N \ll M$,所以可以认为改进后的计算复杂度为 $o(N)$,与改进前相比,理论上的计算复杂程度产生了指数降阶.

不仅如此,由于边长为 $2r$ 的正方形区域的面积非常有限,相对于整体计算区域而言,整体计算区域越大,该区域的相对比重就越小.换句话说,在计算规模越大、参与计算的粒子数目越多、搜索区域越大的情况下,该搜索改进算法的效果就越明显,优化结果如图2中的改进2所示.在第1次改进的基础上,程序模块的运行时间再次大幅降低,降幅达到75%~90%左右,运行时间减少到只有1~2 s左右.

将上述两种优化形式综合起来对溃坝算例进行整体运行测试,取100步时间步长的迭代过程为测试对象,结果表明,未改进前程序运行需要1117 min,而改进后这一结果仅为54 min,节省了约95.25%的时间.这对将MPS法应用到工业实际中产生了非常重要的作用.此外,作者推测,若结合本方法的优点与Link-List方法耦合,可能会有更好的效果,本文不做详述.

2.3 并行计算优化

对MPS法计算效率的优化形式,除了上面提到的算法本身优化以外,另一个很重要的部分就是计算形式的优化,即计算机的硬件条件和执行过程.在硬件条件一定的情况下,可以通过提高计算机的执行效率,即并行计算来提高计算效率.并行计算有很多种类,如MPI、OpenMP或者两者的结合等等.本文采用OpenMP作为并行运算环境,对MPS法的计算执行过程进行优化.

OpenMP作为操作简单、应用广泛的并行运行环境是应用于共享内存式并行系统的多线程程序编译指令.OpenMP将原本复杂的并行结构和算法进行高度的抽象描述,在不用对目标原程序基本算法进行修改的前提下,程序员可以只用简单的在程序相应位置植入专用语句来表明并行意图,编译器就会自动对程序进行相应的模块划分和CPU指定等并行化操作.OpenMP提供的这种对并行过程的高度抽象大大降低了并行编程实现的难度和复杂程度,不仅如此,OpenMP还可以自动调整线程粒度和负载平衡,为适应不同的并行系统配置提供了更强的灵活性.这些简单高效且极易操作的特点也正是OpenMP被广泛认可和应用的原因.

首先,将未进行任何优化的原程序在编译环境2下进行计算,考察由于Fortran编译器的改变对计算效率的影响程度.随后,对上述实验程序中耗时较多的模块添加OpenMP语句进行并行化处理,分别在单核串行、双核并行、四核并行和八核并行的情况下运行,研究目标模块的运行时间,研究结果及其之间的比较如图4所示.

整体而言,在单核串行情况下各模块的运行时间较在编译环境1下发生了非常大的变化,如模块1的运行时间由编译环境1下的41 s减少到编译环境2下的1.4 s,而模块4和模块5的变化则更加剧烈,分别从71 s和70 s减少到813 ms和1 062 ms.这说明编译环境2所采用的高版本Fortran编译器的内部算法有很大的改进,提高了程序的计算效率.

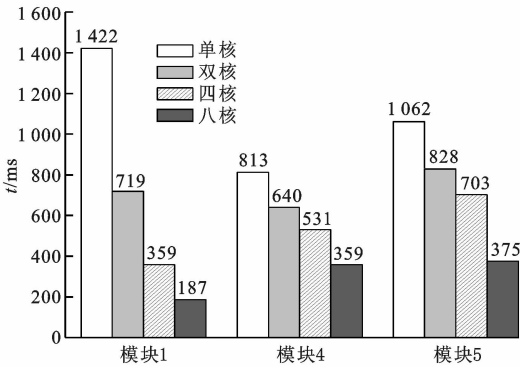


图 4 原程序采用 OpenMP 并行计算的运行时间

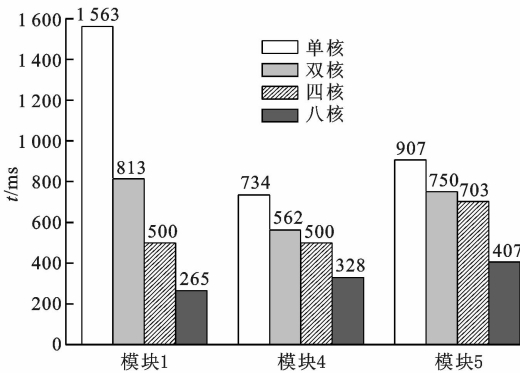


图 5 优化后的程序采用 OpenMP 并行计算的运行时间

但是,编译器对程序计算效率提高的程度并不一致,这与算法本身的结构有较大关系.因此,本文认为在不断改进和优化算法本身的基础上,还要充分利用软件行业的技术进步,采用先进和高版本的编译器也是提高计算效率的一个重要途径.

当采用 OpenMP 进行并行计算,参与计算的 CPU 为双核时,图 4 中所示的 3 个模块的运行时间都有显著的降低,模块 1 的并行时间减小最多,加速比将近 2,而模块 4 和模块 5 的减少程度相对有限;当参与计算的 CPU 增加到四核时,计算时间继续减少,模块 1 的计算效率提高程度仍然十分显著,而后两者依然仅有少量改进;当 CPU 增加到八核时,模块 1 的加速趋势基本不变,模块 5 的加速比基本接近 2,模块 4 改进仍然不多,但运行时间的绝对值已经降得很低.这个现象从侧面说明了各种程序结构对并行计算的适应性是不同的.通过对基于 OpenMP 的并行化改进计算结果进行比较,发现 MPS 法很适宜采用并行方法来提高计算效率,且加速效果非常明显.当然,结果同时也表明加速的效果还与程序的具体算法结构有关.

随后,本文还将经过 2.1 节和 2.2 节优化后的程序也在编译环境 2 中进行并行运算,结果发现在单核串行执行的情况下,采用高版本的 Fortran 编译器使得程序模块的运行时间有部分减少,如图 5 所示.模块 4 和模块 5 的运行时间由 1 s 和 2 s 降低为 734 ms 和 907 ms,有明显的降低,但是模块 1 的运行时间却由原来的 1 s 增加到 1.5 s 左右.随着参与并行的 CPU 核数的增加,各模块的运行时间整体上都是在不断减小的.对于优化后的程序,OpenMP 对模块 1 的加速能力仍然要高于模块 4 和模块 5,这一点与图 4 中所示的趋势完全一致.

以八核并行为例,对比图 4 和图 5 的计算结果

可以看出,整体来讲优化后的程序在采用 OpenMP 进行并行计算时,程序的运行时间接近甚至不如优化前的并行效果,如表 3 中模块 1 和模块 5 所示.

表 3 编译环境 2 下程序模块的并行优化结果比较

模块	算法优化前			算法优化后			综合结果
	运行时间/ms		加速比	运行时间/ms		加速比	
	单核	八核		单核	八核		
1	1 422	187	7.58	1 563	265	5.88	恶化
4	813	359	2.26	734	328	2.34	优化
5	1 062	375	2.83	907	407	2.23	恶化

本文认为这是因为优化前的程序虽然计算量庞大,但是计算结构非常简单,便于 OpenMP 进行区域划分和指定 CPU 进行同步计算,而优化后的程序虽然计算量大大降低,但是也伴随着计算结构复杂程度的明显提升,这并不利于 OpenMP 进行分块操作和 CPU 指派,所以造成了上述结果.因此,尽管各种优化方式都可以单独为提高 MPS 法计算效率提供帮助,但是这些方法之间也需要有较好的兼容性,这样才能将优化效果进行叠加或者放大.由于本文使用了商用语言软件作为计算环境,其内部执行结构无法获知,因而未进行更加详细的研究,但作为比较重要的一个方面,下一阶段可采用开源结构的编译指令继续深入研究.

3 结 论

(1)本文基于数学运算优化和搜索算法优化,结合利用先进的计算机硬件和程序语言,再使用 OpenMP 技术进行并行计算,全方位多角度地对 MPS 法的计算效率进行了优化且收到非常明显的效果,节省计算时间达到 95%左右.

(2)提出了新的搜索算法——位置比较消冗法(PCER),该搜索算法采用位置比较来代替距离比较的思想,消除了大量无效计算并且不增加算法的复杂度和存储运算量.该算法有非常好的普适性,容易在类似算法中移植,还有随着计算规模的增大优化效果更加明显的特点.

(3)算法优化和并行计算等多种优化方式的结合使用需要注意各种方式的特点和相互影响,以达到兼容匹配的最佳优化效果.

综上,计算效率的大幅提高使 MPS 法计算对象的离散粒子可以更加细小,有利于计算向更高精度迈进;同时,还可以扩大计算对象的规模,使计算的适用范围更加广泛,极大地提高了 MPS 法的经济性和实用性.

参考文献:

- [1] KOSHIZUKA S, OKA Y. Moving-particle semi-implicit method for fragmentation of incompressible fluid [J]. Nuclear Science and Engineering, 1996, 123(3): 421-434.
- [2] 席光,项利峰.自由表面流动的移动粒子半隐式模拟方法[J].西安交通大学学报,2006,40(2):249-252.
XI Guang, XIANG Lifeng. Numerical simulation of fluid free surface flow using moving particle semi-implicit method[J]. Journal of Xi'an Jiaotong University, 2006, 40(2):249-252.
- [3] 徐刚,文洋.自由面流动模拟的 MPS 算法研究[J].哈尔滨工程大学学报,2008,29(6):539-543.
XU Gang, WEN Yang. An investigation of MPS algorithm for free surface flow simulation[J]. Journal of Harbin Engineering University, 2008, 29(6): 539-543.
- [4] 李绍武,于志安,熊赞.基于 MPS 无网格方法的孤立波传播的数值模拟[J].海洋学报,2007,29(2):830-835.
LI Shaowu, YU Zhi'an, XIONG Zan. Numerical simulation of solitary wave propagation based on MPS method[J]. Acta Oceanologica Sinica, 2007, 29(2): 830-835.
- [5] 段日强,姜胜耀,张佑杰,等.二维数值模拟蒸汽膜覆盖的液滴水力学特性[J].中国科学:E辑,2005,48(2):180-190.
DUAN Riqiang, JIANG Shenyao, ZHANG Youjie, et al. Two-dimensional numerical simulation of hydrodynamic behaviors of drop covered with vapor film[J].

Science in China: Series E, 2005, 48(2):180-190.

- [6] 潘徐杰,张怀新.用移动粒子半隐式法模拟液舱横摇晃动现象[J].上海交通大学学报,2008,42(11):1094-1097.
PAN Xujie, ZHANG Huaixin. Moving particle semi-implicit method for simulation of liquid sloshing on roll motion[J]. Journal of Shanghai Jiaotong University, 2008, 42(11):1094-1097.
- [7] 张帅,解利军,郑耀.一种稳定的移动粒子半隐式法[J].水动力学研究与进展,2010,25(2):239-246.
ZHANG Shuai, XIE Lijun, ZHENG Yao. A stable moving particle semi-implicit method [J]. Chinese Journal of Hydrodynamics, 2010, 25(2):239-246.
- [8] 梁杨杨,孙中国,席光.液滴斜向撞击液面的 MPS 法数值模拟[J].西安交通大学学报,2010,44(9):93-98.
LIANG Yangyang, SUN Zhongguo, XI Guang. Numerical simulation of droplet oblique impact on liquid film with moving particle semi-implicit method [J]. Journal of Xi'an Jiaotong University, 2010, 44(9):93-98.
- [9] SUN Z G, XI G, CHEN X. Mechanism study of deformation and mass transfer for binary droplet collisions with particle method [J]. Physics of Fluids, 2009, 21(3): 032106.
- [10] 孙中国,席光,王焕然.带运动部件的不可压缩流动 MPS 法数值模拟[J].工程热物理学报,2008,29(10):1676-1678.
SUN Zhongguo, XI Guang, WANG Huanran. Numerical simulation of incompressible flow with moving component using the MPS method[J]. Journal of Engineering Thermophysics, 2008, 29(10):1676-1678.
- [11] SUN Z G, XI G, CHEN X. A numerical study of stir mixing of liquids with particle method[J]. Chemical Engineering Science, 2009, 64(2): 341-350.
- [12] MONAGHAN J J. Particle methods for hydrodynamics[J]. Computer Physics Reports, 1985, 3(2):71-124.
- [13] HERNQUIST L, KATZ N. TREESPH: a unification of SPH with the hierarchical tree method[J]. Astrophysical Journal Supplement Series, 1989, 70(6): 419-446.
- [14] 于志安.水动力学数值模拟 MPS 方法的改进及应用[D].天津:天津大学,2007.

(编辑 荆树蓉)