

NUMERICAL MODELING OF SEDIMENT TRANSPORT NEAR THE BED
USING A TWO-PHASE FLOW APPROACH

By

ANDREA EMILIA GONZALEZ

B.S. (Universidad de Chile, Santiago, Chile) 2000

M.E. (The Johns Hopkins University) 2004

DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Civil and Environmental Engineering

in the

OFFICE OF GRADUATE STUDIES

of the

UNIVERSITY OF CALIFORNIA

DAVIS

Approved:



(Fabian A. Bombardelli)



(Bassam A. Younis)



(David H. Schoellhamer)

Committee in Charge

2008

UMI Number: 3336258

Copyright 2008 by
Gonzalez, Andrea Emilia

All rights reserved.

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI[®]

UMI Microform 3336258

Copyright 2008 by ProQuest LLC.

All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest LLC
789 E. Eisenhower Parkway
PO Box 1346
Ann Arbor, MI 48106-1346

Copyright 2008, Andrea Gonzalez

ABSTRACT

The prediction of the motion of sediment particles close to river beds is a notably complex task. Experimental and numerical studies have provided a basic understanding of the processes affecting the bed load; however, there are still several issues that are elusive to the current technology/theory.

Numerical models have become a standard tool to analyze complex engineering problems. Despite intrinsic limitations derived from the assumptions required to make the theoretical models tractable, their value relies on their flexibility and wider range of applicability, as opposed to statistical results obtained from experiments. Bed load transport can be simulated through three sub-models: a) a set of equations describing the particle "free" flight, b) a sub-model to calculate the post-collision particle velocity and rotation, and c) a mathematical representation of the bed roughness. In this dissertation, a new theoretical/numerical model for bed load motion is presented, including different and novel versions of the above sub-models, in three spatial dimensions (3-D).

The "free" flight sub-model includes the effect of several forces over the particle translation (buoyancy, drag, virtual mass, lift, fluid acceleration, Basset and Magnus forces) and also deals with the particle rotation. A new optimized methodology to compute the Basset force is presented, including both the use of a semi-derivative to calculate the Basset integral and the memory time concept, which reduces the integration span of the term. Important savings in computational time are obtained by using this methodology. This sub-model was validated through comparisons with experimental data.

The post-collision velocity and rotation sub-model features the conservation of linear and angular momentum during the rebound, and it enables a straightforward extension to inter-particle collisions. A new 3-D representation of the bed roughness is introduced by using geometric considerations between the moving particle and the bed, and a stochastic model. The bed is assumed to be formed by uniformly packed spheres and the determination of the contact point between the moving particle and the bed is evaluated by using three different algorithms. The proposed sub-models are compared with bed roughness models existing in the literature, and are compared with experimental data in the sand-gravel size range. It is concluded that the Dependent Bed Angle (*DBA*) sub-model provides the best representation of the bed. The importance of the values of the friction and restitution coefficients is also addressed.

The particle tracking model was coupled with a highly-resolved, 3-D, turbulent flow field, to study the effect of the flow turbulence on the particle motion. The velocity field has been obtained for a flat-plate flow. In spite of the intrinsic differences between the boundary layers in flat plates and in open channels, the velocity field proved to be very valuable in gaining insight into small-scale particle fluctuations. The one-way coupling model was validated with experimental observations. Fluid/particle interactions were investigated by defining the particle turbulent intensity and the particle turbulent kinetic energy. To that end, a new filter to separate turbulence effects from the “mean” flow conditions is presented. The particle tracking model was supplemented with a sub-model of inter-particle collision. The effects of particle size, flow velocity, and particle concentration on the particle turbulent intensity and turbulent kinetic energy are elucidated.

The main contribution of this work is the development of a 3-D particle tracking model for bed load with an unprecedented level of detail. Unlike previous modeling efforts, the proposed model includes all forces in the equations describing particle motion. A new methodology to calculate the Basset force was proposed, which leads to important saving in computational time. Additionally, the influence of bed roughness in the particle trajectory is studied in detail and incorporated in the model. Finally, the influence of turbulence is studied through a one-way coupling simulation for multiple particles within a highly resolved turbulent flow field. These contributions lead to a better understanding of the bed load phenomenon and provide a tool for more realistic predictions in engineering applications.

TABLE OF CONTENTS

Chapter 1: Introduction and Motivation	1
1.1 Problem Statement.....	1
1.2 Research Objectives and Description.....	4
1.3 Definitions, Assumptions and Limitations.....	5
1.4 Dissertation Plan.....	7
Chapter 2: Literature Review.....	10
2.1 Important Definitions.....	10
2.2 Two-Phase Flow Studies.....	11
2.3 Forces Considered in Lagrangian/Eulerian Studies.....	15
2.4 Particle-Wall Collisions.....	19
2.5 Inter-Particle Collisions.....	22
2.6 Turbulence Effect in Bed Load Transport.....	23
2.7 Experimental Data Available.....	24
2.8 Global Evaluation and Knowledge Gaps.....	27
Chapter 3: Two-Dimensional Particle Tracking Model.....	29
3.1 General Considerations.....	29
3.2 Two-Dimensional Particle Tracking Model.....	30
3.2.1 Drag, Lift and Virtual Mass Coefficients.....	36
3.2.2 Two-Dimensional Particle-Wall Collision Algorithm.....	40
3.3 Model Validation.....	42

3.4	Relative Importance of the Forces on the Particle Trajectory.....	47
3.5	Analysis of the Magnus Force.....	49
3.6	A New Algorithm to Compute the Basset Term.....	53
3.6.1	Memory Time Concept.....	57
3.6.2	Time Reduction.....	65
3.6.3	Efficiency of the Basset Force Approximation.....	66
3.7	Assessment of Values for the Friction and Restitution Coefficients.....	66
Chapter 4: Three-Dimensional Particle Tracking-Model.....		69
4.1	Three-Dimensional Particle Trajectory Model.....	69
4.2	Particle Rotation.....	74
4.3	Algorithm for Particle-Wall Collisions.....	76
4.3.1	Particle Rebound Sub-Models.....	76
4.3.2	Previous Models for the Treatment of the Surface: a Roughness (Bed-Representation) Model.....	79
4.3.3	A New Bed-Representation Sub-model.....	79
4.4	Assessments of Particle-Wall Collision Models.....	81
4.4.1	Characteristics of the Numerical Tests.....	82
4.4.2	Simulation Results of the Numerical Tests.....	82
4.4.3	Validation of the Selected Collision Model.....	90
4.5	Analysis of Particle Velocity under a Non-Turbulent Velocity Field.....	93
4.6	Motion of Multiple Particles.....	97
4.7	Inter-Particle Collision Model.....	98

Chapter 5: Particle Motion under A Three-Dimensional Turbulent Velocity Field.....	103
5.1 General Considerations.....	103
5.2 High-Resolution Three Dimensional (HR3D) Velocity Field Simulation Description	104
5.2.1 Boundary Layer in a Flat Plate and in Channel Flows.....	106
5.2.2 Dynamic Similarity.....	107
5.2.3 Simulated Flow Characteristics.....	109
5.3 Three-Dimensional Simulation of the Motion of a Single Particle under a Simulated Velocity Field.....	114
5.3.1 Validation of the Model for a Single Particle.....	114
5.3.2 A New Turbulence Filter.....	119
5.3.3 Particle Turbulence intensity and Turbulent Kinetic Energy Computations.....	121
5.4 Three-Dimensional Tracking of Multiple Particles under Simulated Turbulent Velocity Field.....	123
5.4.1 Validation of the Model for Multiple Particles.....	123
5.4.2 Effect of Inter-Particle Collisions on the Particle Turbulence Intensity and Kinetic Energy.....	129
5.4.3 Effect of Particle Size and Flow Velocity on the Particle Turbulence Intensity and Kinetic Energy	130
5.4.4 Effect of Volumetric Concentration on the Particle Intensity and Turbulent Kinetic Energy.....	133

Chapter 6: Summary and Conclusions.....	135
References.....	142
Appendix A: Computational code	152
Appendix B: Analysis of the semi-derivative method to calculate the Basset integral...	174
Appendix C: Simulation results for $R_p=73$	183
Appendix D: Simulation results for $R_p=250$	186

LIST OF SYMBOLS

α = non dimensional variable, equal to $(1 + R + C_m)^{-1}$;

α_b = angle formed by the tangent to the point of impact of the “flying” particle (with a particle on the bed) and the bed of the channel, defined in the y-z plane;

α_d = lateral deviation angle;

α_{in} = collision angle of the particle with the bed slope, equal to $atan(\tilde{w}_p / \tilde{v}_p)$

ε = turbulent dissipation of the flow;

ε_x = factor indicating the proportion of the velocity in the stream-wise direction;

ε_y = factor indicating the proportion of the velocity in the span-wise direction;

δt = time step;

δ_0 = boundary layer thickness at the inlet;

$\Gamma(\cdot)$ = Gamma function;

θ = angle of the channel bed with respect to a horizontal plane;

θ_b = angle formed by the tangent to the point of impact of the “flying” particle (with a particle on the bed) and the bed of the channel, defined in the x-z plane;

θ_{crit} = maximum angle where the moving particle can hit the bed;

θ_{in} = collision angle of the particle with the bed slope, equal to $atan(\tilde{w}_p / \tilde{u}_p)$;

μ = Dynamic viscosity;

ρ = fluid density;

ρ_s = particle density;

τ = dummy variable for integration;

τ_o = bed shear stress;

τ_f = Flow characteristic time;

τ_B = Basset force time scale;

τ_p = Particle relaxation time;

τ_* = Shields parameter, equal to $u_*^2 / (g R d_p)$;

τ_{*c} = Critical Shields parameter for the initiation of motion;

ν = kinematic viscosity;

$\vec{\tilde{\omega}}$ = dimensionless particle angular rotation vector immediately before the collision;

$\vec{\hat{\omega}}$ = dimensionless particle angular rotation vector immediately after the collision;

$\tilde{\omega}_y$ = particle angular rotation along the span-wise axis;

$\tilde{\omega}_x$ = particle angular rotation along the stream-wise axis immediately before the collision with the bed;

$\tilde{\omega}_y$ = particle angular rotation along the span-wise axis immediately before the collision with the bed;

$\tilde{\omega}_z$ = particle angular rotation along the wall-normal axis immediately before the collision with the bed;

$\hat{\omega}_x$ = particle angular rotation along the stream-wise axis immediately after the collision with the bed;

$\hat{\omega}_y$ = particle angular rotation along the span-wise axis immediately after the collision with the bed;

$\hat{\omega}_z$ = particle angular rotation along the wall normal axis immediately after the collision with the bed;

$\vec{\omega}$ = non dimensional particle rotation vector;

$\vec{\omega}_r$ = non dimensional relative particle rotation vector with respect to the fluid vorticity;

Ω = particle angular velocity in the transverse direction;

a = particle radius;

\vec{c} = relative velocity vector of the mass centre between two particles;

\vec{c}_{fc} = slip velocity vector between particle surfaces;

C = Scaling factor for the largest value among the particle relaxation time, the flow characteristic time and the Basset-force scaling;

C_D = drag coefficient;

C_L = lift coefficient;

C_m = added mass coefficient;

C_t = rotational coefficient;

C_v = volumetric concentration of particles;

d_p = particle diameter;

DBA = Dependent Bed Angle roughness sub-model;

e = restitution coefficient;

f = friction coefficient;

Fr = Froude number, defined as $Fr = \frac{u}{\sqrt{gl}}$

g = acceleration of gravity;

- H = dimensionless particle jump height;
- H_c = channel depth;
- IBA = Independent Bed Angle roughness sub-model;
- \vec{J} = impulsive force exerted over colliding particles;
- J_n = normal component of the force exerted over colliding particles;
- J_t = tangential component of the force exerted over colliding particles;
- l = arbitrary length scale;
- k_s^+ = dimensionless wall roughness, defined as $k_s^+ = (k_s u_*) / \nu$
- K = turbulent kinetic energy of the flow;
- k_s = measure of the roughness of the wall;
- L = dimensionless particle jump length;
- \vec{n} = normal unit vector directed from particle centre to centre;
- N = number of terms considered in the semi-derivative sum;
- N_{back} = number of jumps considered in the computation of the Basset term;
- q = volumetric sediment transport rate;
- q^* = dimensionless volumetric bed load rate $q^* = \frac{q}{\sqrt{g R d_p^3}}$;
- r_1 = geometrical parameter linked to θ_b and θ_{in} ;
- r_2 = geometrical parameter linked to α_b and α_{in} ;
- r_{min1} = minimum value of r_1
- r_{min2} = minimum value of r_2

r_{max1} = maximum value of r_1

r_{max2} = maximum value of r_2

R = submerged specific gravity of sediment, equal to $(\rho_s / \rho - 1)$;

Re = Reynolds number, defined as $Re = \frac{ul\rho}{\mu}$;

Re_p = particle Reynolds number equal to $w_s d_p / \nu$;

Re_δ = HR3D Reynolds number, defined as $(U_m \delta_0 / \nu)$;

R_p = explicit particle Reynolds number equal to $(R g d_p^3)^{0.5} / \nu$;

t = time coordinate;

t^* = inter-particle collision time;

\hat{t} = tangential unit vector in the direction of the slip velocity;

TKE_f = turbulent kinetic energy of the fluid;

TKE_p = turbulent kinetic energy of the particle;

T_{back} = Time interval during the history of the particle affects the current particle velocity;

\vec{u} = particle velocity vector;

\vec{u}^f = particle velocity fluctuation vector;

\vec{u}^a = mean particle velocity vector, averaged over the turbulence;

u^a = mean particle velocity in the stream-wise direction, averaged over the turbulence;

$\vec{\tilde{u}}_p$ = dimensionless particle velocity vector immediately before the collision;

$\vec{\hat{u}}_p$ = dimensionless particle velocity vector immediately after the collision;

$\vec{u}_r^a = \vec{u}^a - \vec{v}^a$ is the particle relative velocity vector;

$\left| \vec{u}_r^a \right|_B$ = magnitude of the particle relative velocity vector evaluated at the bottom of the particle;

$\left| \vec{u}_r^a \right|_T$ = magnitude of the particle relative velocity vector evaluated at the top of the particle;

u = arbitrary velocity scale;

u_* = wall-friction velocity;

u^f = particle velocity in the stream-wise direction;

\vec{u}' = dimensionless velocity fluctuation of the particle vector;

u_f = dimensionless stream-wise fluid velocity;

u'_f = fluid turbulent intensity in the stream-wise direction;

$\left| \tilde{U} \right|_{in}$ = magnitude of the particle velocity vector (dimensionless) immediately before the collision with the wall;

$u_n|_{in}$ = dimensionless wall-normal particle velocity immediately before the collision with the bed;

$u_n|_{out}$ = dimensionless wall-normal particle velocity immediately after the collision with the bed;

u_p = dimensionless stream-wise particle velocity;

\tilde{u}_p = dimensionless stream-wise particle velocity immediately before the collision with the bed;

\hat{u}_p = dimensionless stream-wise particle velocity immediately after the collision with the bed;

u'_p = dimensionless particle turbulent intensity in the stream-wise direction;

$u_t|_{in}$ = dimensionless tangential particle velocity immediately before the collision with the bed;

$u_t|_{out}$ = dimensionless tangential particle velocity immediately after the collision with the bed;

\vec{u}_{pi} = dimensionless particle i velocity vector immediately before the collision particle j;

\vec{u}_{pi} = dimensionless particle i velocity vector immediately after the collision particle j;

$|\vec{u}_r|$ = magnitude of the particle relative velocity vector (dimensionless) measured at its center;

$|\vec{u}_r|_B$ = magnitude of the particle relative velocity vector (dimensionless) evaluated at the bottom of the particle;

$|\vec{u}_r|_T$ = magnitude of the particle relative velocity vector (dimensionless) evaluated at the top of the particle;

U_m = unperturbed stream-wise velocity of the fluid far away from the wall;

U_{filter} = turbulence filter;

UDA = Uniformly Distributed Angle roughness sub-model;

\vec{v} = fluid velocity vector;

\vec{v}^a = fluid mean velocity vector, averaged over the turbulence;

\vec{v}' = fluid velocity fluctuation vector;

\bar{v}_f = dimensionless fluid velocity vector,

v_p = dimensionless span-wise particle velocity;

\tilde{v}_p = dimensionless span-wise particle velocity immediately before the collision with the bed;

\hat{v}_p = dimensionless span-wise particle velocity immediately after the collision with the bed;

v'_f = dimensionless fluid turbulent intensity in the span-wise direction;

v'_p = dimensionless particle turbulent intensity in the span-wise direction;

w^a = mean particle velocity in the stream-wise direction, averaged over the turbulence;

w_p = dimensionless wall-normal particle velocity;

w_s = particle fall (limit) velocity;

\tilde{w}_p = dimensionless wall-normal particle velocity immediately before the collision with the bed;

\hat{w}_p = dimensionless wall-normal particle collision immediately after the collision with the bed;

w'_f = dimensionless fluid turbulent intensity in the wall-normal direction;

w'_p = dimensionless particle turbulent intensity in the wall-normal direction;

x = stream-wise direction;

y = span-wise direction;

z = wall-normal direction.

LIST OF FIGURES

Figure 3-1: Definition sketch for saltating particle in 2-D.....	35
Figure 3-2: Comparison of different expressions for the drag coefficient.....	37
Figure 3-3: Wall collision parameters. Two-dimensional scheme. Lateral view.....	40
Figure 3-4: Comparison of predictions of the particle free-flight sub-model with data obtained by Niño and García (1998b) for sands. Distances are made non-dimensional by using the particle diameter. Single jump case. $d_p=0.6$ mm; $u_* = 0.025$ m/s.....	45
Figure 3-5: Comparison of predictions of the particle free-flight sub-model with data obtained by Niño and García (1998b) for sands. Distances are made non-dimensional by using the particle diameter. Single jump case. $d_p=0.8$ mm; $u_* = 0.027$ m/s.....	45
Figure 3-6: Comparison of predictions of the particle free-flight sub-model with data obtained by Niño and García (1998b) for sands. Distances are made non-dimensional by using the particle diameter. Single jump case. $d_p=0.7$ mm; $u_* = 0.032$ m/s.....	46
Figure 3-7: Comparison of predictions of the particle free-flight sub-model with data obtained by Niño et al. (1994) for gravels. Distances are made non-dimensional by using the particle diameter. Single jump case. $d_p = 30$ mm; $u_* = 0.22$ m/s.....	47
Figure 3-8: Force analysis. $R_p = 73$. $\tau_* / \tau_{*c} = 2$. a) Particle going upwards. b) Particle going downwards.....	49
Figure 3-9: Force analysis. $R_p = 250$. Run 1 Set Niño. $\tau_* / \tau_{*c} = 2$. a) Particle going upwards. b) Particle going downwards.....	49

Figure 3-10: Comparison of simulations with experimental data for the case of a particle moving in a flume and rebounding with the wall (2-D model). The figure shows results associated with: a) the particle jump height (H), b) the particle jump length (L) and c) the particle stream-wise mean velocity. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. $R_p = 250$51

Figure 3-11: Variation of the Magnus force along the wall-normal direction, for different values of the relative shear stress. a) Particle going upwards. b) Particle going downwards..... 52

Figure 3-12: Comparison of the average jump height (a) and average jump length (b) computed using the semi-derivative approximation for the Basset force. Multiple jumps (stochastic collisions). $R_p = 100$. $\tau_* = 0.056$ 61

Figure 3-13: Dimensionless jump variables versus number of jumps in the past (N_{back}) included in the Basset force computation. Multiple-jump simulation using stochastic collision model. $R_p = 100$. $\tau_* = 0.056$. Dotted lines represent the reference range. a) Dimensionless stream-wise component of the velocity. b) Dimensionless wall-normal component of the velocity. c) Dimensionless jump height. d) Dimensionless jump length.....63

Figure 3-14: Dimensionless jump variables versus number of jumps in the past (N_{back}) included in the Basset force computation. Multiple-jump simulation using stochastic collision model. $R_p = 100$. $\tau_* = 0.11$. Dotted lines represent the reference range. a) Dimensionless stream-wise component of the velocity. b) Dimensionless wall-normal

component of the velocity. c) Dimensionless jump height. d) Dimensionless jump length..... 64

Figure 3-15: Dimensionless jump variables versus number of jumps in the past (N_{back}) included in the Basset force computation. Multiple-jump simulation using stochastic collision model. $R_p=100$. $\tau_s=0.28$. Dotted lines represent the reference range. a) Dimensionless stream-wise component of the velocity. b) Dimensionless wall-normal component of the velocity. c) Dimensionless jump height. d) Dimensionless jump length..... 65

Figure 3-16: Effect of the restitution coefficient e on the particle trajectory after colliding with the wall. The value of the friction coefficient is considered constant and equal to 0.4. $R_p=73$. a) Dimensionless particle jump height averaged over 100 jumps. b) Dimensionless particle jump length, averaged over 100 jumps..... 67

Figure 3-17: Effect of the friction coefficient on the particle trajectory after colliding with the wall. The value of the restitution coefficient e is considered constant and equal to 0.8. $R_p=73$. a) Dimensionless particle jump height averaged over 100 jumps. b) Dimensionless particle jump length averaged over 100 jumps..... 67

Figure 4-1: Definition sketch for saltating particle in 3-D..... 71

Figure 4-2: Parameter definition for a 3-D particle-wall collision. Tsuji et al. (1985)....77

Figure 4-3: Three dimensional collision of a particle with the bed. The bed is composed by uniformly packed spheres placed one next to the other. The particle diameter d_p is equal for the moving and the resting spheres. Side view. Definition of α_{in} and α_b 80

Figure 4-4: Comparison of simulations with experimental data for the case of a particle moving in a flume and rebounding with the wall (Run 6, Set Niño). The figure shows results associated with: a) the particle jump height (H), b) the particle jump length (L) and c) the particle stream-wise mean velocity. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. $R_p=73$ 83

Figure 4-5: Overlap Area Determination: $R_p = 73$, Simulation Run 6. Set Niño. The overlap area is defined is depicted as the grey area in the figure..... 84

Figure 4-6: Comparison of simulations with experimental data for the case of a particle moving in a flume and rebounding with the wall (Run 6, Set Niño). The figure shows results associated with: a) the particle jump height (H), b) the particle jump length (L) and c) the particle stream-wise mean velocity. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. $R_p=250$ 86

Figure 4-7: Particle angular velocity simulations. a) Niño and Garcia (1998a) for $R_p=73$. b) Lee and Hsu (1996) for $R_p=2126$ 91

Figure 4-8: Take off angle comparison. Simulation and experimental results. $R_p=73$. Run 6 Set Tsuji..... 92

Figure 4-9: Comparison of the absolute value of the deviation angle obtained in the numerical simulation and experimentally by Niño and García (1998a). a) Absolute value of the deviation angle α_d of particle trajectories as a function of the flow conditions. b) Cumulative probability distribution of the absolute value of the deviation angle..... 93

Figure 4-10: Single particle velocity time series in a velocity field defined by a semi-logarithmic law. $R_p=73$, $\tau_*/\tau_{*c}=2.0$. a) stream-wise component. b) span-wise component. c) wall-normal component..... 94

Figure 4-11: Velocity variation over the time for one particle $R_p=73$, $\tau_*/\tau_{*c}=2$ with the corresponding particle elevation. a) Stream-wise component. b) Span-wise component. c) Wall-normal component..... 95

Figure 4-12: Dimensionless particle rotation along the different axis. $R_p=73$, $\tau_*/\tau_{*c}=2$. a) Stream-wise component. b) Span-wise component. c) Wall-normal component..... 97

Figure 4-13: Three dimensional particle trajectory. No particle-particle collision $R_p=73$, $\tau_*/\tau_{*c}=1.5$ 98

Figure 4-14: Inter-particle collision sketch. Definition of parameters..... 100

Figure 4-15: Particle 1 (dark line) and Particle 2 (light line) trajectories. Both lines represent the position of the center of each sphere. Blank arrows represent the dimensionless time where the both particles collide. Filled arrows represent the dimensionless time where Particle 2 hits the wall. $R_p=73$, $\tau_*=0.048$ 101

Figure 4-16: Snapshot of particles trajectories, showing a inter-particle collision at time $t=0.888$ 102

Figure 5-1: Computed stream-wise velocity component in the simulation domain. The extensions of the different zones are only approximate.....104

Figure 5-2: Comparison between the mean stream-wise fluid velocity obtained from the HR3D simulation and known expressions for the velocity profile in a turbulent open-channel. Solid lines represent analytical expressions; circles represent data obtained from numerical simulation data..... 110

Figure 5-3: Turbulence intensities and turbulent kinetic energy from the numerical simulation (circles) compared with the experimental regressions suggested by Nezu and Nakagawa (1993) (solid line). a) Stream-wise component. b) Span-wise component c) Wall-normal component d) Turbulent kinetic energy of the flow..... 112

Figure 5-4: Wave-number spectrum of the velocity component at the turbulent region. a) stream-wise, b) wall normal and c) span-wise direction.....113

Figure 5-5: Comparison of simulations with experimental data for the case of a particle moving in a flume and rebounding with the wall. The figure shows results associated with: a) the particle jump height (H). b) the particle jump length (L). c) the particle stream-wise mean velocity. d) the mean particle spinning. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. Single-particle simulation $R_p=73$ 116

Figure 5-6: Comparison of simulations with experimental data for the case of a particle moving in a flume and rebounding with the wall. The figure shows results associated with: a) the particle jump height (H), b) the particle jump length (L) and c) the particle stream-wise mean velocity. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. Single particle simulation $R_p=250$ 117

Figure 5-7: Comparison of simulations with experimental data for the case of multiple particles moving in a flume, rebounding with the wall and colliding among themselves.

The figure shows results associated with: a) the take-off angle after a collision (θ_{out}), b) the absolute value of the lateral dispersion angle (α_d) and c) the cumulative probability distribution of the absolute value of the deviation angle. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. $R_p=80$ 118

Figure 5-8: Single particle velocity time series using a high resolution three dimensional velocity field. $R_p=73$, $\tau_*/\tau_{*c}=1.5$. a) stream-wise component. b) span-wise component. c) wall-normal component..... 119

Figure 5-9: Single particle velocity and the new turbulence filter proposed. $R_p=73$, $\tau_*/\tau_{*c}=1.5$ 120

Figure 5-10: Distribution of the particle turbulence intensity and particle turbulent kinetic energy for a single particle moving in a simulated velocity field. $R_p=120$. a) Wall-normal variation of the parameters for $\tau_*/\tau_{*c}=1.5$. b) Variation of the turbulent kinetic energy as the flow condition changes..... 122

Figure 5-11: Comparison between flow and particle turbulence intensity and turbulent kinetic. $R_p=120$, $\tau_*/\tau_{*c}=3$. a) Stream-wise component. b) Span-wise component. c) Wall-normal component. d) Particle turbulent kinetic energy.....124

Figure 5-12: Simulated dimensionless bed load transport rates. Comparison with formulae by Meyer-Peter and Muller (1948), Engelund and Fresse (1976), Fernandez Luque and van Beek (1976) and Parker (1978)..... 125

Figure 5-13: Comparison of simulations with experimental data for the case of multiple particles moving in a flume, rebounding with the wall and colliding among themselves.

The figure shows results associated with: a) the take-off angle after a collision (θ_{out}), b)

absolute value of the lateral dispersion angle (α_d) and c) cumulative probability distribution of the absolute value of the deviation angle. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. $R_p=80$ 126

Figure 5-14: Time series of the velocity components of a single moving particle using a high resolution three dimensional velocity field, when multiple particles are simulated. The dotted circle shows the time when a collision with other particle occurs. $R_p=73$, $\tau_s/\tau_{*c}=1.5$. a) Stream-wise component. b) Span-wise component. c) Wall-normal component.....127

Figure 5-15: Multiple particle simulation with no inter-particle collision. Comparison between flow and particle turbulence intensity and turbulent kinetic energy and their variation in the vertical. $R_p=73$, $\tau_s/\tau_{*c}=1.5$. a) No inter-particle collision. b) Including inter-particle collision..... 130

Figure 5-16: Particle turbulence intensity and particle turbulent kinetic energy vertical profile for multiple particle moving in a simulated velocity field. $R_p=80$. a) Stream-wise component of the turbulence intensity. b) Span-wise component of the turbulence intensity. c) Wall-normal component of the turbulence intensity. d) Particle turbulent kinetic energy..... 131

Figure 5-17: Particle turbulence intensity and particle turbulent kinetic energy vertical profile for multiple particle moving in a simulated velocity field. $R_p=120$. a) Stream-wise component of the turbulence intensity. b) Span-wise component of the turbulence intensity. c) Wall-normal component of the turbulence intensity. d) Particle turbulent kinetic energy..... 132

Figure 5-18: Particle turbulence intensity and particle turbulent kinetic energy vertical profile for multiple particle moving in a simulated velocity field. $R_p=120$, $\tau_*/\tau_{*c}=2.5$. Concentration variation. a) Stream-wise component of the turbulence intensity. b) Span-wise component of the turbulence intensity. c) Wall-normal component of the turbulence intensity. d) Particle turbulent kinetic energy..... 134

LIST OF TABLES

Table 2-1: Summary of two-phase flows studies. The three last works are large-scale models.....12

Table 2-2: Summary of Lagrangian/Eulerian studies for particle motion. Forces included.....16

Table 2-3: Characteristics of experimental results of saltating particles..... 26

Table 3-1: Comparison of the dimensionless jump height (H) and length (L) due to the change in the model discretization time dt . $R_p=100$, $\tau_*=0.056$44

Table 3-2: Relative error of the quadratures associated with the semi-derivative approach and the Second Euler-Maclaurin summation formula, when used to compute the Basset integral with $u_r(\tau) = \tau^3 / 3$ (for different numbers of points/terms). The analytical solution for this case is given by: $\frac{-2}{15} \sqrt{t-\tau} (8t^2 + 4t\tau + 3\tau^2)_0^t$, which was evaluated at $t=100$. The relative error was computed as the absolute value of the difference between the numerical and the analytical results of the Basset integral multiplied by 100 and divided by the analytical value, for both methodologies.....55

Table 3-3: Relative error of the quadratures associated with the semi-derivative approach and the Second Euler-Maclaurin summation formula, when used to compute the Basset integral $du_r(\tau)/d\tau = \cos(\tau)$ (for different numbers of points/terms). The analytical solution for this case is given by:

$$\int_0^t \frac{\cos \tau}{\sqrt{t-\tau}} d\tau = -\sqrt{2\pi} \left\{ \cos(\tau) C \left[\sqrt{\frac{2}{\pi}} \sqrt{t-\tau} \right] + \sin(\tau) S \left[\sqrt{\frac{2}{\pi}} \sqrt{t-\tau} \right] \right\}_0^t, \text{ with the}$$

Fresnel integrals given by $C(\tau) = \int_0^{\tau} \cos\left(\frac{\pi z^2}{2}\right) dz$ and $S(\tau) = \int_0^{\tau} \sin\left(\frac{\pi z^2}{2}\right) dz$; this

integral was evaluated at $t=100 \pi/2$. The relative error was computed as the absolute value of the difference between the numerical and the analytical results of the Basset integral multiplied by 100 and divided by the analytical value, for both methodologies..... 56

Table 3-4: Decrease in simulation time (in percentage) versus N_{back} . Run considers 500 particle jumps.....65

Table 4-1: Comparison of results of 3-D particle-tracking model in terms of change in the particle jump characteristics when complete and simplified lift force expressions are used. Δ represents the percentage of change of the simplified expression relative to the complete expression.....73

Table 4-2: Comparison of results of 3-D particle-tracking model in terms of change in the particle jump characteristics when complete and simplified lift force expressions are used. Δ represents the percentage of change of the simplified expression relative to the complete expression..... 73

Table 4-3: Values of coefficient C1, C2 and C3. Computation of C_t coefficient.....75

Table 4-4: Set of values of friction and restitution coefficients used in the particle-wall collision model comparison..... 82

Table 4-5: Simulation summary..... 82

Table 4-6: Statistical analysis using the overlap area indices. $R_p = 73$ 85

Table 4-7: Statistical analysis using the root mean squared error (RMSE). $R_p = 250$87

Table 5-1: Frequency of inter-particle collisions. $R_p = 120$, $C = 0.1\%$ 128

Table 5-2: Frequency of inter-particle collisions. $R_p = 120$, $\tau_*/\tau_{*c} = 2.5$ 128

ACKNOWLEDGEMENTS

I am most grateful to Professor Fabian A. Bombardelli who has been very supportive and whose scientific insight, motivation and encouragement were inspiring. In addition, Fabian granted me freedom, protection, support and guidance during these years at the University of California, Davis. For all this, I am truly thankful.

I would like to thank the members of my committee, Professor Bassam Younis and Professor Dave Schoellhamer, for their fruitful discussions and helpful conversations, whose advise definitely improve the outcome of this work.

I also would like to thank the people I worked with these years: Sanjeev Jha, Dane Behrens, Arash Massoudieh and Ryan Moniz. Everyone of them helped to make the work environment friendly and productive.

I would like to gratefully acknowledge the financial support provided by the Civil and Environmental Engineering Department at the University of California, Davis, the University of California Water Resources Center and the Cooperative Institute for Coastal and Estuarine Environmental Technology.

I would like to thanks all my friends, who have made my life in the United States so enjoyable. Specially, Maria Eugenia González, Carlos Zuritz, Maria Elena Faricelli and Josue Medellín, for all the memorable evenings spent together.

Finally, and most importantly, I would like to thank to my family. My husband, Marcelo, has given me his love, friendship, encouragement and assistance. With his support, this adventure has become a wonderful life experience. To my son, Rafael, for give me the most beautiful smile every day and the strength to pursue my dreams.

Finally, to my mom, Emilia, my dad, Nelson, and my sisters, Claudia and Karina, for their unconditional love and support from the distance. Muchas gracias!

A mis raíces, mi familia.

CHAPTER 1

INTRODUCTION AND MOTIVATION

1.1 Problem Statement

Since the beginning of mankind, sedimentation processes have been associated with multiple human activities, such as water supply, irrigation, and flood control, among others. Hundreds of billions of dollars are currently devoted annually to control sediment-related problems, solely in the United States. Approximately \$500 billion are spent yearly, only in dredging US harbors and waterways for navigation (García, 1999). The loss of storage capacity in reservoirs of US dams can sum up to \$100 million per year (Julien, 1998). These and other expenses provide a clear motivation for directing efforts to improving the current understanding of sediment motion in water bodies.

Sediment transport encompasses important processes that occur in many environmental and industrial conditions. Several contaminants of concern in aquatic systems are hydrophobic, which means they adsorb to solid material located either in the bed or in suspension in the water (Bonzogo et al., 1996; Shresta and Orlob, 1996; Gardner and Theis, 1996; Wang et al., 2004; Massoudieh et al., 2006; Massoudieh et al., 2008). Suspended sediments can also be associated with pollution driven by pathogens: virus may attach to colloids in coastal areas (Redman et al., 1997). Because of this a detailed knowledge of the transport and fate of sediments is necessary to quantify the transport and fate of contaminants. Considering the risk to biota and human health driven by toxic chemicals, the study of sediment transport becomes especially important.

In addition, in industrial applications such as pneumatic transport of fine materials and the hydraulic transport of minerals (slurry), the efficiency of the transport is strongly dependent upon the interaction of particles and flow. In this case, a more thorough knowledge of these processes could lead to a decrease of construction and operational costs in those facilities.

Sediment transport in rivers can be classified in three categories: a) suspended load, b) wash load, and c) bed load (Julien, 1998; Garcia, 1999). In bed load, the grains maintain a quasi permanent contact with the bed in a very narrow region, usually defined as a few grain diameters thick (Garcia, 1999).

The analysis of bedload is a fairly complex task. Depending on the size of the particle and the particular flow condition, sediments are transported along the river bed by rolling, sliding and saltating (Hu and Hui, 1996). According to previous observations, the percentages in each mode depend upon the flow intensity, which can be expressed through the parameter $\Theta = \tau_0 / (\rho_s - \rho) g d_p$, where τ_0 corresponds to the bed shear stress, ρ_s and ρ denote the density of the sediment and water respectively, g is the acceleration of gravity and d_p is the particle diameter. While rolling accounts for 80% of the bedload motion when Θ is less than 0.8, the fraction of rolling drastically decreases as τ_0 increases (Hu and Hui 1996). Saltation accounts for nearly 50% of the bedload motion when Θ is equal to 0.15, and it is more than 60% when Θ is equal to 0.2 (Hu and Hui 1996). Therefore, saltation is considered to be the main form of bed load motion in most natural conditions (Einstein, 1950; Sekine and Kikkawa, 1992, Lee et al., 2000). The bed load transport has been widely studied since the beginning of the past century, using both empirical and analytical approaches.

Despite the progress in measurement techniques, it is still difficult to perform detailed observations of flow and particle motion near the bed, whether in the laboratory or in situ. Empirical formulas for bed load transport are therefore restricted to certain ranges of flow and particle characteristics, and extrapolation to conditions beyond those ranges is not reliable. Nevertheless, these formulas are widely used in practice, even outside of their ranges of applicability.

The numerical simulation of sediment transport in general has proven to be a promising tool to estimate the amount of material put into motion and to provide information on bed changes. However, the accuracy of those models is still limited (Duan et al., 2004) and there is a lack of knowledge of the small-scale interactions of particles with the flow (Haratty et al., 2003). Existing numerical models are able to give quantitative predictions in some cases, and are restricted to qualitative predictions in real conditions of practical interest (see Duan et al., 2004; Olsen, 2003).

In particular, several models for bed load transport have been presented recently for dilute mixtures (Sekine and Kikawa, 1992; Niño et al., 1994; Niño and García, 1998b). (In this study, a mixture is said to be dilute if the concentration of particles in water is smaller than 1 or 2% by volume.) Within this approach, it has been assumed that particles are driven by the flow and there is no interaction between particles and the surrounding fluid. Notwithstanding this hypothesis, the predictions of those models have been quite satisfactory.

Recently, some authors have incorporated aspects of the two-phase flow theory to describe particle-laden flows (in a general sense). This theory has been successfully applied in different problems in the fields of chemical, nuclear and mechanical

engineering (Drew, 1976; Elghobashi and Abou-Arab, 1983; Gore and Crowe, 1989; Ahmadi and Ma, 1990; Squires and Eaton, 1994). This is a theory in which the liquid and solid phases obey basic conservation laws coupled through interactions forces (Jiang et al., 2004).

Two-fluid models have been also used with success in improving the prediction of particle concentrations in sediment-laden flows in open channels (Greimann et al., 1999; Greimann and Holly, 2001; Hsu et al., 2003; Jiang et al., 2004). Considering these accomplishments, is in the writer's opinion that the use of the concepts of two-phase flow theory in bed load transport problems will increase our knowledge about this type of particle-laden flow. This theory provides the tools to understand the small-scale interactions between the sediment and the fluid, a subject that has been addressed only in a limited fashion in the case of particles moving close to river beds.

1.2 Research Description and Objectives

The main objective of this work is to develop a two-phase flow model to integrate fluid and particle motion using an Eulerian-Lagrangian approach close to the channel bed. This work focuses on the water flow close to the wall and the transport of sediment in bed load motion, and it includes processes that have been disregarded in most existing models, most notably inter-particle collision.

The specific objectives of this study are:

- To develop and validate an Eulerian-Lagrangian model to simulate the saltation of particles close to the bed. The Lagrangian model is intended to be coupled with highly-resolved velocity fields, using Large Eddy Simulations (LES) or Direct

Numerical Simulations (DNS). Validation of this model is accomplished via-comparison with experimental data.

- To optimize the computational effort of the Lagrangian model to calculate particle trajectories and velocities, by revisiting the formulations for all forces; with special emphasis is put on the Basset force.
- To examine existing bed-representation sub-models, to propose a simple yet realistic new approach and to validate it via comparison with experimental data.
- To gain more understanding about the response of saltating particles to the velocity fluctuations in the flow (i.e., to turbulence). To that end, a highly-resolved flow simulation, considering one-way coupling between phases, is developed.

1.3 Definitions, Assumptions and Limitations

In the present work, sediment particles are assumed to be perfect spheres saltating near the bed.

Their trajectories are calculated using a Lagrangian approach. The saltation process is numerically simulated via two separate stages of motion: 1) the free flight of particles through the flow and 2) the collision with the wall.

In the first stage, the particle trajectory is defined by the hydrodynamic forces acting on it. Models for this stage have been extensively developed by several researchers (Maxley and Riley, 1983; Mei et al., 1991; Niño and García, 1994; Lee and Hsu, 1994; Schmeeckle and Nelson, 2003; Lukerchenko et al., 2006, Lee et al., 2006, González et al., 2006; Bombardelli et al., in press). In this work, the effect of buoyancy, lift, virtual

mass, drag, Basset, Magnus, and fluid acceleration forces are considered. Electrochemical forces, that are present in the case of very small particles, have been left out of the analysis.

The second stage relates to the process whereby the particle hits the wall and eventually rebounds. The wall-particle collision event was represented by using two sub-models: a) a set of equations to calculate the particle velocity after the rebound and b) a sub-model to represent the bed roughness. The proposed model does not consider any mechanism to prevent the particle motion after a rebound; therefore, if the flow conditions allow the particle motion, the particle will move with a saltating trajectory without stopping.

To study the effect of the spatial and temporal variations in the flow, a highly resolved 3-D (HR3D) turbulent velocity field was used. This velocity field was obtained by simulating a boundary-layer flow on a smooth flat plate. Despite differences between a boundary layer in a flat plate and in an open-channel flow (the latter akin to rives and flumes; see, Nieuwstadt and Bradshaw, 1997; Ashrafian, 2004), the mean stream-wise velocity has been found to be accurately represented by the semi logarithmic law of the wall in both flows (Gad-El-Hak 2000). Wei et al. (2005) present "compelling evidence" of the similitude of both flows in a large region, and therefore it is assumed in this work that the use of the HR3D field of a flat plate is a satisfactory velocity field for the purpose of this study..

This work promotes a mechanistic view of sediment transport using the two-phase flow theory regardless of the range of particle sizes; this approach aims at avoiding the use of limiting empirical formulas as much as possible. Given the random nature of the

transport of sediments, it is impossible to eliminate completely the use of empirical formulas. Values of coefficients associated with surface forces over the particle (such as lift and virtual mass) and from the particle collision (i.e., friction and restitution coefficients) are considered in this study as known values (from previous studies). It is worth to pointing out that there is still a great debate surrounding those values.

1.4 Dissertation Plan

The results obtained in this work are organized in five chapters, as follows. Chapter 2 presents a literature review of the existing particle tracking models, wall-particle and inter-particle collision algorithms, bed roughness representations and the use of spatially-temporally variable velocity fields in the study of bed load transport. A summary of the available experimental data used for validation purposes, and a review indicating the knowledge gaps that the writer addresses in this dissertation, are also included.

Chapter 3 presents the results obtained with a two-dimensional (2-D) particle tracking computational code, including a rebound sub-model and a bed roughness representation. The model was validated through comparison of numerical predictions with experimental data. To optimize the computational cost of the particle tracking model, a new methodology to calculate the Basset force was developed and it was tested for particles of both small and large size.

In Chapter 4, a three-dimensional (3-D) particle tracking model is presented, which includes the description of both the particle translational and rotational velocity at every moment. An assessment of existing sub-models for bed roughness representation is

introduced in this chapter together with a new sub-model. The validation of the best sub-model is accomplished by comparing its performance with experimental data. The computational code also considers the motion of multiple particles and an algorithm to treat the inter-particle collisions.

Chapter 5 presents the results of the proposed 3-D model coupled with a highly resolved 3-D (HR3D) turbulent velocity field. After its validation, the effect of the turbulence on the particle motion is studied in detail. The interaction between particles, the effect of particle size, volumetric concentration of particles and flow conditions on particle turbulent parameters are discussed in this chapter. A new filter to separate the fluctuating component of the particle velocity from the "mean" value is introduced in this chapter.

Finally Chapter 6 summarizes the contributions of this study, provides the final conclusions of this dissertation and presents guidelines for future work.

Specific topics addressed in this dissertation have been communicated in papers published or in press in several international proceedings and journals.

- González, A.E., Bombardelli, F.A. and Niño, Y. (2006). "Towards a Direct Numerical Simulation (DNS) of particle motion close to river beds". in River Flow 2006, Proc. of the Int. Conf. on Fluvial Hydraulics, Lisbon, Portugal, R. M. L. Ferreira, E. C. T. L. Alves, J. G. A. B. Leal, and A. H. Cardoso (Eds.), Vol. 1, pp. 799-806.
- González, A.E., Bombardelli, F.A. and Niño, Y. (2006). "Improving the prediction capability of numerical models for particle motion in water bodies". in

The 7th Int. Conf. on Hydrosience and Engineering (ICHE-2006), Sep 10 – 13, Philadelphia.

- González, A.E., Bombardelli, F.A., Calo, V. and Niño, Y. (2007). "Simulation of the particle motion close to a rigid bed". presentation at the ASME Applied Mechanics and Materials Conference, June. Austin, Texas.
- Bombardelli, F.A. and González, A.E. (2008). Discussion of "Analytical approach to calculate rate of bank erosion" by Duan, J.G. *Journal of Hydraulic Engineering*, 134 (2), 280-281.
- Bombardelli, F. A., González, A. E., and Niño, Y. I. (2008). "Non-linear, Lagrangian theoretical models for particle motion close to solid boundaries". *Journal of Hydraulic Engineering*, in press.

Recent submissions:

- González, A.E., Bombardelli, F.A. and Moniz, R. (2008) "Generalized algorithms for particle motion and collision with river beds". submitted to the *Computer and Geosciences Journal*.
- González, A.E., Bombardelli, F.A. and Calo, V. (2008). "Particle saltating motion under a highly resolved 3D turbulent velocity field ". future submission.

CHAPTER 2

LITERATURE REVIEW

This chapter presents the state of the art in the modeling of particle-laden flows. Specifically, this chapter reviews previous works which employ the two-phase flow theory for particle motion; it presents an assessment of the different forces used in previous Eulerian/Lagrangian studies (with emphasis on the Basset force) and discusses works dealing with different wall-particle and inter-particle collision algorithms. The chapter also includes information concerning previous experimental setups that were used to validate the model presented in this dissertation. Finally a global evaluation of the state of the art is introduced.

2.1 Important Definitions

A flow is considered dilute if the effect of particle-particle interactions is not significant (Crowe et al., 1998). Particle-particle interactions can refer to two separate mechanisms: a) particle-particle collisions (where the particle can rebound, shatter or coalesce) and b) particle-particle fluid-dynamic interactions (where the proximity of the particles affects their fluid dynamic forces). Dilute flow will generally include *one-way coupling* (where the disperse phase is affected by the continuous phase) and *two-way coupling* (where the disperse-phase also affects the continuous phase). The *three-way coupling*, refers to mechanism b) above where particles may affect locally the flow field of neighboring particles, thereby promoting time-varying changes in the local structure of

the flow. (This effect adds to the two-way coupling.) In addition, particles may suffer from frequent collisions in non-dilute mixtures, effect known as *four-way coupling*.

Two different methods usually have been employed to describe two-phase flows. The Lagrangian approach is a natural extension of particle mechanics, and it focuses on material particles as they move through the flow. Each particle in the flow is labeled, or identified by its original position. On the other hand, in the Eulerian approach, one concentrates on what happens at a spatial window and it is especially useful to describe how a fluid behaves (Kundu and Cohen, 2004). These two different approaches have been used to describe liquid-particle flows under the two-phase flow theory: Eulerian-Eulerian and Eulerian-Lagrangian. While in both cases the carrier fluid is treated as a continuum, the disperse phase can be analyzed as a continuum (Eulerian-Eulerian approaches) or as a set of particles that need to be followed individually (Eulerian-Lagrangian approaches).

2.2 Two-Phase Flow Studies

Several authors have contributed with numerical simulations of particle-laden flows. Different approaches have been presented in order to solve the system of partial differential equations concerning the particle and fluid motions. A brief summary is presented in Table 2-1.

One-way coupling has been the main level of interaction used by several authors in order to describe the particle motion in a turbulent flow. Sommerfeld (1992, 2003) used a $K-\varepsilon$ turbulent closure in order to treat the fluid turbulence; Dorgan and Loth

(2004) and van Haarlem and Boersman (1998) performed DNS to treat the continuous phase and used a Lagrangian approach for the particle motion.

Table 2-1: Summary of two-phase flows studies. The three last works are large-scale models

Author	Year	Eul/ Lag or Eul/ Eul	One /Two Way Cou- pling	Colli- sions	Dilute Mix- ture	Set-up	Particle size	Gas/ Water	Method
Amoudry et al.	2008	E-E	Two	Yes	No	Open- channel	D=0.21- 0.46 mm	Water	K-e
Boivin et al.	1998	E-L	One /Two	No	Yes	Isotrop- ic tur- bulence	D=0.11- 0.35 η	Gas	DNS
Cheng & Pereira	2000	E-L	One /Two	Yes	Yes	Pipe	D=400- 520 μm	Water	K-e
Dorgan & Loth	2004	E-L	One	Elastic Wall Collisio ns	Yes	Channel	D=y+	Gas	DNS
Elghobash i & Truesdell	1993	E-L	Two	No	Yes	Isotrop- ic tur- bulence	D \ll η	Gas	DNS
Ferrante & Elgobashi	2003	E-L	Two	No	No	Isotrop- ic tur- bulence	D=0.04- 0.6 η	Gas	DNS
Greimann et al.	1999	E-E	Two	No	Yes	Open- channel	D=0.13- 1.3 mm	Water	Anali. Exp
Greimann & Holly	2001	E-E	Two	Yes (no near the bed)	Yes	Open- channel	D=0.26- 1.3 mm	Water	Anali. Exp
Hsu et al.	2003	E-E	Two	No	Yes	Open- channel	D=0.13- 0.24 mm	Water	Anali. Exp
Kartushins -ky & Michaeli- des	2004	E-E	One	Yes	No	Pipe	D=243- 501 μm	Gas	K-e
Kartushins -ky & Michaeli- des	2007	E-L	Four	Yes	No	Channel	D=24- 88 μm	Gas	K-e
Lun & Liu	1997	E-L	Two	Yes	Yes	Channel	D=500 μm	Gas	K-e
Portela & Oliemans	2003	E-L	One /Two	No	Yes	Pipe	Zero- volume	Gas	DNS /LES
Schabani et al.	2005	E-L	One	Yes	Yes	Open- channel	D=500 mm	Water	Law of the wall

Author	Year	Eul/ Lag or Eul/ Eul	One /Two Way Cou- pling	Colli- sions	Dilute Mix- ture	Set-up	Particle size	Gas/ Water	Method
Schmee- ckle and Nelson	2003	E-L	One	Yes	Yes	Channel	D=5 mm (mean)	Water	Exp (Flow)+ DNS
Sommer- feld	2003	E-L	One	Yes	Yes	Channel	D=30- 195 μ m	Gas	K-e
Sommer- feld	1992	E-L	One	Yes	Yes	Channel	D=45- 108 μ m	Gas	K-e
Squires & Eaton	1990	E-L	One	No	Yes	Isotro- pic tur- bulence	$D \ll \eta$	Gas	DNS
Sundaram & Collins	1999	E-L	Two	No	Yes	Isotro- pic tur- bulence	D=100 μ m	Gas	DNS
Ten Cate et al.	2004	E-L	Two	Yes	No	Isotro- pic tur- bulence	$D > \eta$	Gas	DNS
van Haarlem et al.	1998	E-L	One	No	Yes	Channel	$D \ll \eta$	Gas	DNS
Yamamo- to et al.	2001	E-L	One /Two	Yes	Yes	Pipe	D=7-70 μ m	Gas	LES
Yeganeh et al.	2000	E-L	Two	Yes (soft sphere model)	No	Open- channel	D=5 mm	Water	K-e
Young & Leeming	1997	E-E	Two	No	Yes	Pipe	D=0.28- 0.5y+	Gas	RANS

On the other hand, two-way coupling has been employed in some occasions to capture interesting phenomena related to particle diffusion, inter-particle interactions and wall-bouncing effects (Taniere et al., 2004; Portela and Oliemans, 2003).

Portela and Oliemans (2003) developed a code for DNS and LES of particle-laden turbulent gas flow using an Eulerian-Lagrangian approach. They considered in their simulations a very small fraction of point heavy particles (dilute condition); thus, the effect of the particle volume was not taken into account on the mass balance of the fluid. They also assumed that the distance between particles was large compared with the

particle size, neglecting hydrodynamic coupling. Portela and Oliemans considered drag as the only force acting on the particles.

Most of the studies mentioned above have involved the transport of very fine particles through a gas, mainly for industrial applications, where the particle size has been assumed to be relatively uniform. Those studies have usually dealt with the transport of particles in close conduits, such as pipes (Portela and Oliemans, 2003; Young and Leeming, 1997) and channels (Lun and Liu, 1997; Dorgan and Loth 2004).

Comparatively, little research has been performed regarding the transport of particles (sediment transport) in open channels using the two-phase flow theory. Some studies have been developed for dilute suspensions, using an Eulerian-Eulerian point of view. Greimann et al. (1999) and Greimann and Holly (2001) used this approach to obtain an analytical expression of the vertical concentration profile for suspended sediment in an open channel. Using the same approach, Hsu et al. (2003) used the boundary layer approximation to obtain concentration profiles that provided closer predictions to measurements than the single-phase Rouse formula. Jiang et al. (2004) included additional dispersion terms that predicted more accurately the sediment concentration distribution along the water depth.

On the other hand, the Eulerian-Lagrangian approach has not been much used in sediment transport problems either. Chen and Pereira (2000) developed a two-phase flow code using the K- ϵ model to simulate the movement of the particles in an axisymmetric sudden expansion pipe. Schmeeckle and Nelson (2003) focused on providing a detailed collision model for different particle sizes, to represent in a realistic way how natural particles interact. However, they used a fluid velocity field obtained from experiments

rather than simulating it with the two-phase flow equations. Yeganeh et al. (2000) used the Eulerian/Lagrangian approach to study sheetflows under a high bottom shear stress by using a 2-D model with inter-particle collisions. Amoudry et al. (2008) predicted the time evolution of sediment transport for sheetflows in unsteady conditions

2.3 Forces Considered in Lagrangian-Eulerian Studies

When using a Lagrangian approach to describe the trajectory of a single particle moving in a flow field, the Second Newton's law is applied. Generally speaking, the acceleration of the particle is determined by the sum of all the external forces affecting it. Those forces include in the body particle forces induced by the gravitational field (buoyancy) and the forces induced by pressure and shear stresses imposed by the fluid on the particle surface. Forces coming from pressure differences and stresses have been the subject of much research, and usually a point-force approach is used to evaluate those forces (Dorgan and Loth, 2007). These forces include the quasi-steady drag force; the lift force generated by the shear stress and the particle rotation (usually called Magnus force); the virtual or added mass force which accounts for the work required to change the momentum of the surrounding fluid as the particle accelerates; the fluid acceleration force which accounts for the forces that would exist in the absence of the particle, due to acceleration of fluid and the hydrostatic pressure gradient; and the unsteady-drag force, usually called "history" or Basset force, which addresses the temporal delay in the development of the boundary layer surrounding the particle as a consequence of changes in the relative velocity (Crowe et al. 1998).

A particularly important theoretical equation for particles moving in a low Reynolds number flow field was developed by Maxey and Ridley (1983); this equation (which models the motion of a small particle moving in unbounded fluid, within the Stokes range) has been the basis for most of the research done in this field. Depending on the complexity of the flow simulation, particle size and the physical size of the problem to simulate, the effect of some of the previously listed forces has been disregarded by several authors, as Table 2-2 summarizes.

The references with the symbol (*) correspond to application of particles moving in pneumatic systems (except Mordan and Pinton, 2000), where usually the emphasis of the study is put in how to model the complexity of the flow field. The remaining 13 references deal with applications of bed load transport, where usually only the drag, buoyancy and lift force are considered to determine the particle trajectory. Basset, Magnus and virtual mass force are usually neglected, sometimes without solid arguments or clear-cut evidence.

Table 2-2: Summary of Lagrangian/Eulerian studies for particle motion. Forces included.

Autor	Year	Flow method	Forces	Particle size	Particle Material
Armenio & Fiorotto (*)	2001	Pseudo-spectral method	Drag (linear), virtual mass, fluid acceleration, Basset	$d_p = 0.005-0.01$ mm	$\rho_p / \rho = 2.65-2650$
Chen & Pereira (*)	2000	K-e	Drag, lift, virtual mass, buoyancy	$d_p = 0.4-0.5$ mm	glass beads
Dorgan & Loth (*)	2004	DNS	Drag (linear), buoyancy	$d_p = 0.024$ mm	solid sphere $\rho_p / \rho = 1$
Mondant & Pinton	2000	Still water	Drag (linear), buoyancy, virtual mass, Basset	$d_p = 0.25-3$ mm	glass, steel & tungsten particles

Autor	Year	Flow method	Forces	Particle size	Particle Material
Portela & Oliemans	2003	DNS-LES	Drag	$d_p = 0.66 \text{ y+}$	$\rho_p / \rho = 1000$
Sommerfeld (*)	1992	2-D K-e model	Drag, lift due to slip-shear, gravity, Magnus	$d_p = 0.05-0.11 \text{ mm}$	glass beads
Sommerfeld (*)	2003	Potential law 1/7 + measured rms values	Drag, lift, buoyancy, Magnus	$d_p = 0.03-0.7 \text{ mm}$	glass beads
Tsuji et al. (*)	1997	Boundary layer eq.	Drag, buoyancy, lift, Magnus	$d_p = 1 \text{ mm}$	polystyrene beads
van Haarlem & Boersma (*)	1998	DNS	Drag (linear)	$d_p = 0.15-0.25 \text{ y+}$	point spheres
Vojir & Michaelides (*)	1994	Sinusoidal flow, uniform flow w/wo turbulence	Drag, buoyancy, virtual mass, Basset, fluid acceleration	-	-
Yamamoto et al. (*)	2001	LES	Drag, lift, Magnus, gravity	$d_p = 0.007-0.07 \text{ mm}$	$\rho_p / \rho = 0.7-8.8$
Harada & Gotoh	2006	Logarithmic expression + Monte Carlo Method for turbulence	Drag, buoyancy, virtual mass	Sand range	natural sediment
Hu & Hui	1996	2-D Logarithmic expression	Drag, buoyancy, lift, virtual mass, fluid acceleration, Magnus	$d_p = 2-3.2 \text{ mm}$	natural sediment
Lee & Hsu	1994	2-D Logarithmic expression	Drag, buoyancy, lift	$d_p = 1.36-2.47 \text{ mm}$	natural sediment
Lee et al.	2000	2-D Logarithmic expression	Drag, buoyancy, lift	$d_p = 6 \text{ mm}$	natural sediment
Lee et al.	2006	2-D logarithmic expression	Drag, buoyancy, lift	$d_p = 0.039-0.068 \text{ mm}$	natural sediment
Lukerchenko et al.	2006	2-D Logarithmic expression	Drag, buoyancy, virtual mass, Basset, Magnus	$d_p = 15 \text{ mm}$	natural sediment
Lun & Liu	1997	K-e	Drag, buoyancy, lift, Magnus	$d_p = 1 \text{ mm}$	polystyrene bead
Niño & Garcia	1994	2-D Logarithmic expression	Drag, buoyancy, lift, virtual mass, basset, Magnus	$d_p = 15-31 \text{ mm}$	natural sediment

Autor	Year	Flow method	Forces	Particle size	Particle Material
Niño & Garcia	1998b	2-D Logarithmic expression	Drag, buoyancy, lift, virtual mass, Basset, Magnus	$d_p = 0.5-0.8$ mm	natural sediment
Schmeeckle and Nelson	2003	Experimental measurements	Drag, buoyancy	$d_p = 2-7$ mm	natural sediment
Shabani et al.	2005	3-D logarithmic expression	Drag, buoyancy, virtual mass	$d_p = 0.05$ mm	natural sediment
Sekine & Kikkawa	1992	3-D logarithmic expression	Drag, buoyancy, virtual mass	$d_p = 0.3-0.6$ mm	natural sediment
Yeganeh et al.	2000	2-D K-e	Drag, buoyancy, virtual mass	$d_p = 5$ mm	glass beads

Although several authors have disregarded the Basset force in their models of bed load transport (see for instance Wood and Jenkins, 1973, Lee and Hsu, 1994, Schmeeckle and Nelson, 2003, Shabani et al, 2005, Harada and Gotoh, 2006) there is recent evidence that the Basset force becomes important for relatively small particle sizes, i.e., for relatively small explicit particle Reynolds numbers, $R_p = (R g d_p^3)^{0.5} / \nu$, where R is $(\rho_s / \rho) - 1$, ρ and ρ_s denote fluid and particle density, respectively, g is the acceleration of gravity, d_p is the particle diameter and ν is the kinematic viscosity of water. Comparisons of numerical results with laboratory observations have shown that, while the Basset force is negligible for gravels moving as bed load (with R_p of the order of 20,000; Niño and García, 1994), it becomes extremely important for sands (with R_p of the order of 100; Niño and García, 1998b). When the Basset force is neglected in the case of sands, the length of a single particle jump can be under-predicted by about 40%, and the jump height can be under-predicted by about 15% (see Fig. 2 in Niño and García 1998b and Figs. 3-4 to 3-6 in this dissertation). These differences can accumulate for

multiple jumps and lead to very large errors in computations of transport. Mordant and Pinton (2000) in turn performed laboratory tests of spheres settling in water, with diameters ranging from 0.5 mm to 6 mm, and fall (limit) velocities varying from 0.07 to 1.16 m/s. They found that the Basset force needs to be included in the Lagrangian models in order to correctly describe the particle acceleration when the particle Reynolds number is smaller than 4,000 (given by $Re_p = w_s d_p / \nu$, where w_s is the particle fall (limit) velocity). This result is in accord with the findings of Niño and García (1994, 1998). In addition, consistent with the above results, Armenio and Fiorotto (2001) found that the Basset force is appreciable for Re_p of the order of, and smaller than, 1, for a large range of density ratios.

In Lagrangian models, the evaluation of the Basset force can be extremely time-consuming and requires the storage of the relative acceleration of the particle, which can make the simulation very demanding in terms of computer memory. Michaelides (1992) recast the linear equations of particle motion using Laplace transforms. He also employed “canonical” velocity fields to simplify the analysis, and presented a novel procedure to reduce the computational cost of including the Basset force. Obviously, his procedure does not apply to non-linear equations, or to random fields which prevail in flumes and rivers.

2.4 Particle-Wall Collisions

A complete collision model usually can be divided in two sub-models: a) a series of equations to describe the particle velocity after the rebound and b) a representation of the bed roughness. The simplest collision sub-model would be to assume that the particle

rebounds in a completely elastic way on a flat bed and, therefore, that the particle velocity after hitting the wall is exactly the same than the particle velocity before hitting the wall, changing only the direction of the *velocity component in the wall-normal direction*. However, this method neglects the loss of energy associated with the collision.

Using equations for the conservation of momentum and energy during the collision, and/or geometrical considerations, several authors have derived expressions for the velocity of a particle after hitting a wall (Matsumoto and Saito, 1970a; Tsujimoto and Nakagawa, 1983; Tsuji et al., 1985; Niño and García, 1994, Lukerchenko et al., 2006). Two of these models, proposed by Tsuji et al. (1985), and Niño and García (1994), have been widely used in different applications of particles moving close to a rigid wall (Tsuji et al., 1987, Sommerfeld, 1992; Niño and García, 1998a,b; Sommerfeld and Huber, 1999; Lee et al., 2000; Kartushinsky and Michaelides, 2004; Lee et al., 2006). In spite of the general agreement in the physics behind the collision equations, there is still considerable debate regarding how to reproduce the randomness in the saltating process that particles naturally experience.

The randomness of the collision phenomenon is driven by the effect of the shape of the bouncing particle (Matsumoto and Saito, 1970a; Schmeeckle and Nelson 2003) and by the intrinsic roughness of the wall (Matsumoto and Saito, 1970b; Tsujimoto and Nakawaga, 1983; Tsuji et al., 1987; Sekine and Kikkawa, 1992; García and Niño, 1992). In reality, both mechanisms contribute, but numerical simulations become increasingly complicated if non-spherical particles are considered.

One of the first models which considered the influence of the wall roughness and the effect of particle shape in particle-wall bouncing was that proposed by Matsumoto

and Saito (1970b). In this study, the wall was represented by a sinusoidal shape, where the phase of the roughness was randomly specified. A different approach to represent the wall roughness was proposed by Tsujimoto and Nakagawa (1983). The authors assumed that the bed was composed by uniformly-packed spheres aligned at the bottom of the channel. The random nature of the collision was given by the point of contact between the moving particle and the bed particles.

Subsequent modeling has increasingly created more complex bed configurations. Sekine and Kikkawa (1992) considered that the bed was composed by similar diameter spheres. The centroid of each sphere composing the bed protruded above or was depressed below the mean bed elevation, by an amount that varied randomly. Scotti (2006) assumed that the bed is composed by ellipsoids, randomly oriented at the bottom. All previous models required a detailed description of the geometry of the wall, which in some cases could lead to high computational costs.

On the other hand, several authors have provided different methodologies to simulate the roughness of the bottom, without describing the exact geometry of the bed. The effect of the bed roughness is usually incorporated in the collision model as a change in coordinate system due to the existence of an angle between the coordinate system associated with the channel and the real point of contact between the flying particle and the bed. Sommerfeld (1992) adopted some concepts from the work of Tsuji et al. (1985) and replaced the wall (assumed to be plane) by a virtual wall, where its inclination was assumed to be *uniformly distributed* in the range $(-4^\circ, +4^\circ)$ (Tsuji et al., 1987) and *normally distributed* between -4 and 4° (Sommerfeld, 1992). García and Niño (1992) and Niño and García (1994) provided a stochastic estimation of the angle of collision by

using a random generator to produce values of r (an auxiliary variable) with uniform probability for given values of the incident angle. The definition of r is the result of considering the bed formed by uniformly-packed spheres.

2.5 Inter-Particle Collisions

Two approaches are usually followed to model inter-particle collisions: “hard-sphere” and “soft-sphere” methods. Those techniques are employed to compute the velocity, energy, and spin after the collision of particles as a function of the incoming values of velocity, energy, and spin.

In hard-sphere systems, particle deformation during the collision is neglected and the trajectoryies of the particles are determined by momentum-conserving binary collisions. The interactions between particles are assumed to be pair-wise, i.e, only binary collisions are considered, which are also assumed to be instantaneous. For not too dense systems, the hard-sphere models are considerably faster than the soft-sphere models (Deen et al., 2007).

In more complex situations, the particles may interact via short or long-ranges forces, and the trajectories are determined by integrating the Newtonian equation of motion. In cases where the particle concentration is high (e.g., fluidized beds, sheet flows), the use of spring/dashpot systems to describe the collision between particles have been widely used (Gotoh and Sakai, 1997; Yeganeh et al., 2000; Schabani et al., 2005; Harada and Gotoh, 2006)

Simulation of multiple particles is a computationally demanding task and the incorporation of inter-particle collision algorithms could make the problem intractable.

Several authors have simply disregarded the interaction between moving particles (van Haarlem and Boersma, 1998; Portela and Oliemans, 2003; Dorgan and Loth, 2004).

Inter-particle collisions based on hard-sphere models have been implemented and studied by Sommerfeld (1992, 2003), Lun and Liu (1996), Schmeckle et al. (2001), Yamamoto et al. (2001), Schmeckle and Nelson (2003), Kartushinsky and Michaelides (2004) and Ten Cate et al. (2004). However, there is little experimental information regarding the detailed process of collision, and only general parameters (such as the friction and restitution coefficients) are obtained from these experiments (Schmeckle et al., 2001).

When the number density of the particles and the collision frequency are sufficiently large, inter-particle collisions can be modeled as a stochastic model (Sommerfeld, 2001). These models usually rely on generating a fictitious collision partner and the calculation of the collision probability according to the gas kinetic theory.

On the other hand, when the density number of the particles is small, inter-particle collision can be computed directly by estimating the occurrence of collisions from particle trajectories and computing the velocity changes according to momentum and energy conservations laws. In the present study, the latter approach is considered by using an extension of the Tsuji et al. (1985) algorithm for particles colliding with a wall, presented by Crowe et al. (1998) for the case of two moving particles colliding.

2.6 Turbulence Effect on Bed Load Transport

In the most recent numerical simulations for particles moving in saltation mode (Lee and Hsu, 1994; Niño and García, 1994; Hu and Hui, 1996; Lee et al., 2006;

Lukerchenko et al., 2006; Lee et al., 2006) the velocity field has been represented using a logarithmic expression, which is a very good approximation of the average flow conditions in open channels. In addition, sedimentary structures such as bedload sheets, are produced by a spatially and temporally variable sediment transport resulting from the interaction of a turbulent flow with sediment grains in an erodible bed (Heathershaw and Thorne, 1985; Best, 1992; Nelson et al., 1995; Mazumder, 2000; Schmeeckle and Nelson, 2003). Therefore, to improve the prediction capability of numerical simulations of bed load transport it is necessary to provide a velocity field capable of reproducing the turbulence characteristic of natural open-channel flows.

The use of detailed 3-D flow models (obtained with LES, DNS) integrated to models for the simulation of suspended or bed load sediments transport in fluvial environments is relatively limited (Keylock et al., 2005). For suspended solids, a few 3-D models that incorporate flow unsteadiness (Yost and Katopodes, 1996; Zedler and Street, 2001) correspond to early attempts to study the effect of turbulence on sediment transport. For bed load transport, to the best of the writer's knowledge, no research resolving an unsteady velocity field has been presented. A first attempt to introduce variability in the velocity field around particles saltating close to a fixed bed was presented by Schmeeckle and Nelson (2003). In that study, the instantaneous velocity was obtained from near-bed measurements.

2.7 Experimental Data Available

Experiments providing detailed information of individual particle trajectories in bed load motion are scarce. Although several studies have offered information on particle

saltation velocity (\bar{u}_p) in the 70s and 80s (Gordon et al., 1972; Fernandez-Luque and van Beek, 1976; Arbulieva et al., 1987; Sutsepın, 1987) none of them has included other properties of the saltation phenomenon such as jump length (L) and height (H).

The most recent experiments containing detailed information on the trajectories of particles saltating in water are those by Niño et al. (1994), Lee (1993), Lee and Hsu (1994), Niño and García (1998a,b) and Lee et al. (2006). The main characteristics of these experiments are presented in Table 2-3. The experimental results of Niño and García (1998b) show somewhat unexpected values for the particle mean velocity in the stream-wise direction. It is expected that, as the value of τ_* increases, the value of the stream-wise particle velocity u_p also increases; however, the experimental results follow an almost constant line. The same quasi-constant trend is also observed in the measurements of the particle jump height, which is also expected to increase as the value of τ_* increases. In the case of the experiments by Lee (1993), Lee and Hsu (1994) and Lee et al. (2006) the values of H , L and \bar{u}_p increase as the value of τ_* increases, as expected.

For the particle rotation the experimental data is even more scarce. Data of the average particle spinning in the span-wise axis (ϖ_y) is presented in Niño and García (1998a) for particles of sand size, and in Lee and Hsu (1996) for larger particles (Table 2-3).

The values of the friction (f) and restitution (e) coefficients (see Chapter 3 for definition) are not always readily available from experiments. They highly depend on the material of the saltating particle and on the flow conditions; therefore, it is not possible to

provide general values. For natural sediments, Niño and García (1998a) showed that the value of the friction coefficient is relatively constant and equal to 0.89, but the value of the restitution coefficient varies between 0.5 and 0.2 depending of the flow conditions (Niño and García, 1998a). Schmeckle et al. (2001) used values of 0.1 and 0.65 for the friction and restitution coefficients, respectively, obtained also from experiments in a turbulent channel with natural sediments. Tsuji et al. (1987) considered polystyrene particles moving in a gas flow and used a value of 0.8 and 0.4 for both e and f , respectively. For non-spherical particles, Shen et al. (1989) employed values of 0.95 and 0.3 for the same coefficients.

Table 2-3: Characteristics of experimental results of saltating particles.

Author	Recording device	Particle size (mm)	Number of jumps considered	u_c (m/s)	Data available
Niño et al. 1994	Standard video camera	15-31	80	0.14-0.23	Average and standard deviation of H , L and \bar{u}_p
Lee and Hsu (1996)	High-speed photography	4.76-6.54	800	0.066-0.102	Average ϖ_y
Niño and García (1998a,b)	High-speed video camera	0.5-0.8	1-2 jumps for 100 particles	0.021-0.026	Average and standard deviation of H , L , \bar{u}_p and ϖ_y
Lee et al. (2000)	Standard video camera	6	No information available	0.038-0.054	Average of H , L and \bar{u}_p
Lee and Hsu (1994)	Standard video camera	1.36-2.47	No information available	0.036-0.105	Average of H , L and \bar{u}_p
Lee et al. (2006)	Standard video camera	0.6	No information available	0.039-0.068	Average of H , L and \bar{u}_p

2.8 Global Evaluation and Knowledge Gaps

Based on the above assessment, it is noticed that particle-laden flows have been studied using the two-phase flow theory mostly in gas flows, for isotropic and homogeneous turbulence, zero-volume particles, and dilute mixtures. They have not been employed much to address the motion of particles in rivers or open channels where the turbulence is neither isotropic nor homogeneous. Particles in gas flows present also a different response time.

Sediment transport under natural conditions presents characteristics that can be simulated by numerical means using the two-phase flow theory. Complex issues that need to be addressed in this problem are related to the interaction among particles and the turbulent nature of any natural flow.

Most of particle tracking and collision models presented before are two-dimensional in nature. More understanding of the flow can be gathered through DNS or LES velocity fields, linked to a 3-D particle tracking code and a collision algorithm (including a bed-representation sub-model).

For the 3-D particle tracking code, the relative importance of the forces on a particle is dependent upon the particle size. No study, to the best of the writer's knowledge, has addressed a wide range of particle sizes.

In the case of wall-collision algorithms, the disagreement among researchers is centered on how to include in an algorithm the shape of the bouncing particle and the intrinsic roughness of the wall in a physically-meaningful way, thereby providing a realistic approximation of what is observed in nature. In addition to collisions of particles

with beds of rivers, several applications coming from the mechanical engineering field have prompted the development of collision models for industrial purposes. To the best of the writer's knowledge, no systematic study has addressed the accuracy of proposed algorithms of collision with the wall, nor the ranges of validity of the diverse coefficients involved.

One problem associated with the papers related to natural sediment moving in open channels (Sekine and Kikkawa, 1992; Niño and García, 1994, 1996, 1998a,b; and Schmeeckle and Nelson, 2003) is that the fluid flow has been considered to be unaffected by the presence of the moving particles, and it has been assumed to be represented by the law of the wall, without spatial/temporal fluctuating components. The benefits of using this methodology are clear: it imposes the fluid velocity as a time independent known condition for the problem. However, in order to improve the approximations of the transport of sediment as bed load, it is necessary to model more accurately the fluid flow-sediment particle interaction. A numerical simulation involving both interacting phases (solid and liquid) appears as a requisite in order to obtain more accurate predictions.

A first step is to consider a one-way coupling simulation, where the carrier velocity field is simulated, including the turbulence dynamics. The effect of the turbulence on particles in bed load motion has not been studied in detail, to the best of the author's knowledge.

CHAPTER 3

TWO-DIMENSIONAL PARTICLE TRACKING MODEL

3.1 General Considerations

The Eulerian-Lagrangian representation of the two-phase flow under study requires a particle tracking model able to describe the position of each particle at each time step. Using the Newton's second law, and numerical approximations, the position, velocity, and acceleration of the particle are calculated using the forces acting on each particle. As a first step, a two-dimensional (2-D) theoretical particle model presented by Niño and García (1994) was used to develop a computational code to track the position of one particle. After its validation, several issues regarding the particle collision with the wall and the computation of the forces driving the particle motion were addressed.

Given some initial conditions, the expressions provided by Niño and García can be integrated over time to obtain the particle trajectory and velocities in the stream-wise and wall-normal directions. In spite of the detailed inclusion of all forces affecting the particle trajectory, there still remain some issues that need to be defined and discussed. For instance, the values of the coefficients of drag, lift, and virtual mass are assumed in the model to be known and well defined. However, there is disagreement among authors on how to estimate those values, and on the values themselves.

The Basset term involves additional difficulties for the numerical integration because the denominator in the integrand vanishes when the upper integration limit is enforced. Also, the Basset integral considers the integration of the particle acceleration from the beginning of its motion, which leads to important computational memory

requirements and large computing time. These factors could make the calculation inviable.

3.2 Two-Dimensional Particle Tracking Model

One of the most recent and complete forms of the governing equation for the motion of a small spherical particle in an unbounded fluid (within the Stokes drag range) was presented by Mei et al. (1991), based on the equations proposed earlier by Maxley and Riley (1983):

$$\begin{aligned} \rho_s \frac{d\vec{u}}{dt} = & (\rho_s - \rho)\vec{g} + \frac{9}{d_p} \left(\frac{\rho \mu}{\pi} \right)^{1/2} \int_0^t \frac{d}{d\tau} \left(\vec{v} - \vec{u} + \frac{1}{24} d_p^2 \nabla^2 \vec{v} \right) \frac{d\tau}{\sqrt{t-\tau}} + \rho \frac{D\vec{v}}{Dt} \\ & + \rho C_m \frac{d}{dt} \left(\vec{v} - \vec{u} + \frac{1}{24} d_p^2 \nabla^2 \vec{v} \right) - \frac{18}{d_p^2} \left(\vec{u} - \vec{v} - \frac{1}{24} d_p^2 \nabla^2 \vec{v} \right) \end{aligned} \quad (3-1)$$

where ρ and ρ_s denote fluid and particle density, respectively; \vec{u} and \vec{v} indicate the particle and fluid velocity vectors, respectively; μ refers to the dynamic viscosity; C_m denotes the added mass coefficient; d_p indicates particle diameter; \vec{g} is the vector of gravity acceleration; t denotes time, and τ is a dummy variable for integration. The terms on the right hand side of Eq. (3-1) correspond to the following forces per unit volume, respectively: buoyancy force, Basset history force, the forces due fluid acceleration, virtual mass, and Stokes drag. The operator $D(\cdot)/Dt$ indicates the material derivative using the fluid velocity in the advective term, as opposed to the material derivative using the particle velocity in $d(\cdot)/dt$.

Considering the Reynolds decomposition of the particle velocity ($\vec{u} = \vec{u}^a + \vec{u}'$, where \vec{u}^a and \vec{u}' represent the particle mean velocity vector (averaged over the turbulence) and particle velocity fluctuation vector respectively), and for the fluid velocity ($\vec{v} = \vec{v}^a + \vec{v}'$ where \vec{v}^a and \vec{v}' , represent the fluid mean velocity vector and fluid velocity fluctuation vector respectively) expression (3-1) is replaced by:

$$\begin{aligned} \rho_s \left(\frac{d\vec{u}^a}{dt} + \frac{d\vec{u}'}{dt} \right) &= (\rho_s - \rho) \vec{g} + \frac{9}{d_p} \left(\frac{\rho \mu}{\pi} \right)^{1/2} \int_0^t \frac{d}{d\tau} \left(\vec{v}^a + \vec{v}' - \vec{u}^a - \vec{u}' + \frac{1}{24} d_p^2 \nabla^2 \vec{v}^a + \frac{1}{24} d_p^2 \nabla^2 \vec{v}' \right) \frac{d\tau}{\sqrt{t-\tau}} \\ &+ \rho \frac{D\vec{v}^a}{Dt} + \rho \frac{D\vec{v}'}{Dt} + \rho C_m \frac{d}{dt} \left(\vec{v}^a + \vec{v}' - \vec{u}^a - \vec{u}' + \frac{1}{24} d_p^2 \nabla^2 \vec{v}^a + \frac{1}{24} d_p^2 \nabla^2 \vec{v}' \right) \\ &- \frac{18}{d_p^2} \left(\vec{u}^a + \vec{u}' - \vec{v}^a - \vec{v}' - \frac{1}{24} d_p^2 \nabla^2 \vec{v}^a - \frac{1}{24} d_p^2 \nabla^2 \vec{v}' \right) \end{aligned} \quad (3-2)$$

Averaging over the turbulence, neglecting second order terms (such as $\nabla^2 \vec{v}^a$) and fluid acceleration terms, the following expression is obtained:

$$\rho_s \frac{d\vec{u}^a}{dt} = (\rho_s - \rho) \vec{g}_i - \frac{18}{d_p^2} (\vec{u}^a - \vec{v}^a) + \frac{9}{d_p} \left(\frac{\rho \mu}{\pi} \right)^{1/2} \int_0^t \frac{d}{d\tau} (\vec{v}^a - \vec{u}^a) \frac{d\tau}{\sqrt{t-\tau}} + \rho C_m \frac{d}{dt} (\vec{v}^a - \vec{u}^a) \quad (3-3)$$

Second order terms are normally small compared with the other terms. Using dimensional analysis of the term $d_p^2/24 \nabla^2 \vec{v}^a$, it is possible to scale it as $\approx d_p^2 u/l^2$, where u is a velocity scale and l is an arbitrary length scale. Assuming l is equal to the average particle jump length, experimental data of Niño and García (1998 a,b) shows that the fraction $(dp/l)^2$ is of the order of 10^{-2} , indicating that the zero order term (\vec{u}^a) is a

hundred times larger than the corresponding second order term, validating the simplification of the latter term.

Because Eq. (3-1) considers only linear drag, it is only valid when the particle Reynold number ($Re_p = d_p \left| \vec{u}_r^a \right| / \nu$, where ν is the kinematic viscosity of the fluid and $\vec{u}_r^a = \vec{u}^a - \vec{v}^a$ is the particle relative velocity vector) is much smaller than one. To extend the governing equation in a way to make it applicable to larger particles and higher velocities (i.e., larger Reynolds numbers), the linear drag force needs to be replaced by the following non-linear form (Niño and García, 1994):

$$-\frac{3}{4} \frac{C_D}{d_p} \rho \left| \vec{u}^a - \vec{v}^a \right| \left(\vec{u}^a - \vec{v}^a \right) \quad (3-4)$$

where C_D is the drag coefficient.

Considering that the gradients of mean flow velocity in the vertical direction are large in the region close to the bed (as opposed to those in the horizontal direction) and because the particle sizes considered in this study cover a wide range, a lift force in the vertical component must be taken into account. Following Wiberg and Smith (1985), the lift term to be added in the right hand side of Eq. (3-3) is:

$$\frac{3}{4} \frac{C_L}{d_p} \rho \left(\left| \vec{u}_r^a \right|_T^2 - \left| \vec{u}_r^a \right|_B^2 \right) \quad (3-5)$$

where the sub indices T and B denote particle top and bottom, respectively, and C_L is the lift coefficient.

Eventually, the rotation of the saltating particle can introduce another force in the vertical direction, known as the Magnus force or rotational lift (as presented in Gupta and Pagalthivarthi, 2006, for instance). White and Schulz (1977) added a term in the right

hand side of Eq. (3-3), derived from the original expression of Rubinow and Keller (1961):

$$\frac{3}{4} \rho \left| \vec{u}_r^a \right| \left(\Omega - \frac{1}{2} \frac{\partial u^f}{\partial z} \right) \quad (3-6)$$

where Ω denotes the particle angular velocity in the transverse direction, u^f is the time-averaged stream-wise component of the fluid velocity and z represents the wall-normal direction.

Adding lift and Magnus terms to Eq. (3-3), the acceleration of a particle immersed in a velocity field can be calculated using the following expression:

$$\begin{aligned} \rho_s \frac{d\vec{u}^a}{dt} = & (\rho_s - \rho) \vec{g} - \frac{3}{4} \frac{C_D}{d_p} \rho (\vec{u}^a - \vec{v}^a) \left| \vec{u}_r^a \right| + \frac{9}{d_p} \left(\frac{\rho \mu}{\pi} \right)^{1/2} \int_0^t \frac{d}{d\tau} (\vec{v}^a - \vec{u}^a) \frac{d\tau}{\sqrt{t-\tau}} \\ & + \rho C_m \frac{d}{dt} (\vec{v}^a - \vec{u}^a) + \frac{3}{4} \frac{C_L}{d_p} \rho \left(\left| \vec{u}_r^a \right|_T^2 - \left| \vec{u}_r^a \right|_B^2 \right) \vec{z} + \frac{3}{4} \rho \left| \vec{u}_r^a \right| \left(\Omega - \frac{1}{2} \frac{\partial u^f}{\partial z} \right) \vec{z} \end{aligned} \quad (3-7)$$

where \vec{z} is the unit vector in the wall normal direction. After some arrangements, a new expression is obtained:

$$\begin{aligned} (\rho_s / \rho + C_m) \frac{d\vec{u}^a}{dt} = & (\rho_s / \rho - 1) \vec{g} - \frac{3}{4} \frac{C_D}{d_p} (\vec{u}^a - \vec{v}^a) \left| \vec{u}_r^a \right| + \frac{9}{d_p} \left(\frac{\nu}{\pi} \right)^{1/2} \int_0^t \frac{d}{d\tau} (\vec{v}^a - \vec{u}^a) \frac{d\tau}{\sqrt{t-\tau}} \\ & + C_m \frac{d}{dt} \vec{v}^a + \frac{3}{4} \frac{C_L}{d_p} \left(\left| \vec{u}_r^a \right|_T^2 - \left| \vec{u}_r^a \right|_B^2 \right) \vec{z} + \frac{3}{4} \left| \vec{u}_r^a \right| \left(\Omega - \frac{1}{2} \frac{\partial u^f}{\partial z} \right) \vec{z} \end{aligned} \quad (3-8)$$

The local mean flow velocity in the stream-wise direction can be specified by using the semi-logarithmic law for turbulent open-channel flows, considering all cases of fully-rough, smooth, and transitional flows, according to whether $k_s^+ = (k_s u_*) / \nu$ is larger than 70, smaller than 5, or in between those values, respectively (White, 1974; Niño and García, 1994, 1998). (In the equation above, k_s is a measure of the roughness

of the wall, and u_* the wall-friction velocity. Eq. (3-8) involves in fact 3 equations, one for each of the three directions (the stream-wise, wall-normal and span-wise directions). The proposed velocity field is 2-D, and it could be assumed that the particle motion in the span-wise direction is negligible. Defining the particle diameter d_p as the length scale and the wall-friction velocity u_* as the velocity scale, the above equation can be recast in dimensionless terms in the directions x and z as follows:

$$\frac{du_p}{dt} = \alpha \frac{\sin \theta}{\tau_*} - \frac{3}{4} \alpha C_D (u_p - u_f) \left| \vec{u}_r \right| + \alpha C_m w_p \frac{du_f}{dz^*} + \frac{9\alpha}{\sqrt{\pi R_p} \tau_*^{1/4}} \int_0^t \frac{d}{d\tau} (u_f - u_p) \frac{d\tau}{\sqrt{t-\tau}} \quad (3-9)$$

$$\begin{aligned} \frac{dw_p}{dt} = & -\alpha \frac{\cos \theta}{\tau_*} - \frac{3}{4} \alpha C_D w_p \left| \vec{u}_r \right| + \frac{3}{4} \alpha C_L \left(\left| \vec{u}_r \right|_T^2 - \left| \vec{u}_r \right|_B^2 \right) - \frac{9\alpha}{\sqrt{\pi R_p} \tau_*^{1/4}} \int_0^t \frac{d}{d\tau} w_p \frac{d\tau}{\sqrt{t-\tau}} \\ & + \frac{3}{4} \alpha \left| \vec{u}_r \right| \left(\varpi_y - \frac{1}{2} \frac{du_f}{dz} \right) \end{aligned} \quad (3-10)$$

In (3-9) and (3-10), $u_p = u^a/u_*$ and $w_p = w^a/u_*$ are the dimensionless particle velocity in the stream-wise and wall-normal directions (Fig. 3-1, where u^a and w^a are the stream-wise and wall normal components of \vec{u}^a); $u_f = u^f/u_*$ is the dimensionless, stream-wise component of the fluid velocity; \vec{u}_r is the dimensionless particle relative vector; τ_* is given by $u_*^2/(g R d_p)$; g is the acceleration of gravity; R_p equals $(R g d_p^3)^{0.5}/\nu$; ϖ_y equals $\Omega d_p/u_*$; and $\alpha = (1 + R + C_m)^{-1}$. Eqs. (3-9) and (3-10) were used by Niño and García (1994) in their model of gravel bed load (1994) and sand bed load (1998b) transport.

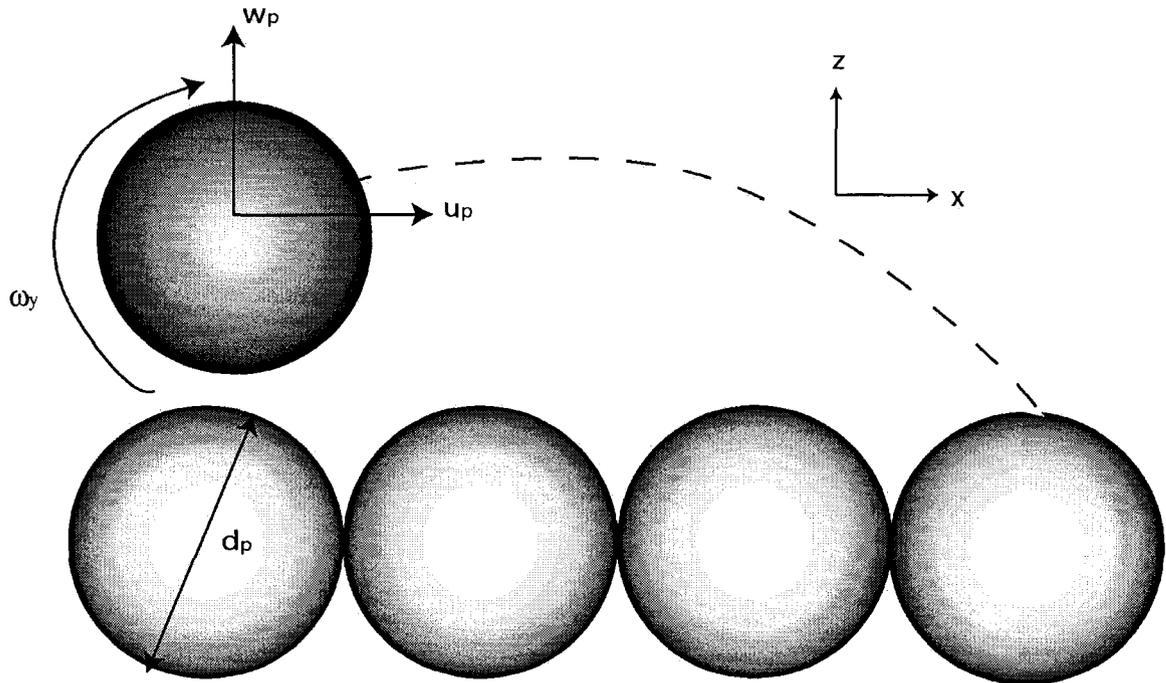


Figure 3-1: Definition sketch for saltating particle in 2-D.

Mathematically, this is an initial-value problem for which initial conditions are needed. After computing the velocities, the positions of the particles are determined by simple numerical integration. It can be noticed that the integral contained in the Basset term becomes improper when evaluated in the upper integration limit. Niño and García used the approximation proposed by Brush et al. (1964) to overcome that problem, dividing the integral in two terms.

$$\int_0^t \frac{du_{ri}}{\sqrt{t-\tau}} d\tau = \int_0^{t-\delta t} \frac{du_{ri}}{\sqrt{t-\tau}} d\tau + 2\sqrt{\delta t} \left(u_j \frac{du_{fi}}{dx_j} - \frac{du_{pi}}{dt} \right) \quad (3-11)$$

with δt denoting the time step employed in the numerical solution, and i indicating the component x or z . Summation on j is implied in (3-11).

The angular velocity is an external variable that must be estimated *a priori* in the Niño and García (1994) model. They used an expression obtained from laboratory experiments (Niño and García, 1998a) to estimate the value of ϖ_y as:

$$\varpi_y = 5.11 - 1.13 \frac{\tau_*}{\tau_{*c}} \quad (3-12)$$

where τ_{*c} denotes the critical, threshold value of the dimensionless bed shear stress for the initiation of motion, which was estimated using the Shield's curve (Shield, 1936).

3.2.1 Drag, Lift and Virtual Mass Coefficients

The value of the drag coefficient C_D for a sphere is a function of the flow Reynolds number. For creeping flow, the drag coefficient can be computed as $C_D = \frac{24}{\text{Re}_p}$ (Oseen, 1927), valid only in the Stokes range. Unfortunately, the behavior of C_D for unsteady particle motion is not completely known (Yen, 1992) and the only alternative is to estimate it as that of a single, free-falling sphere. To increase the range of applicability of this coefficient, several authors have proposed different expressions. Rubey (1933) proposed a simple approximation to the drag coefficient:

$$C_D = \frac{24}{\text{Re}_p} + 2 \quad (3-13)$$

The expression derived by Engelund and Hansen (1967) is best suited to natural sands and gravels:

$$C_D = \frac{24}{\text{Re}_p} + 1.5 \quad (3-14)$$

According to Karamanev (2001), one of the best correlations with experimental results for a freely settling or rising sphere was proposed by Turton and Levenspiel (1986):

$$C_D = \frac{24}{Re_p} \left(1 + 0.173 Re_p^{0.6257} \right) + \frac{0.413}{1 + 1.163 \times 10^4 Re_p^{-1.09}} \quad (3-15)$$

The approximation proposed by Yen (1992) for the drag coefficient has been used before for the motion of saltating particles (Niño and García, 1998b; Lukerchenko et al., 2006):

$$C_D = \frac{24}{Re_p} \left(1 + 0.15 \sqrt{Re_p} + 0.017 Re_p \right) - \frac{0.208}{1 + 10^4 Re_p^{-0.5}} \quad (3-16)$$

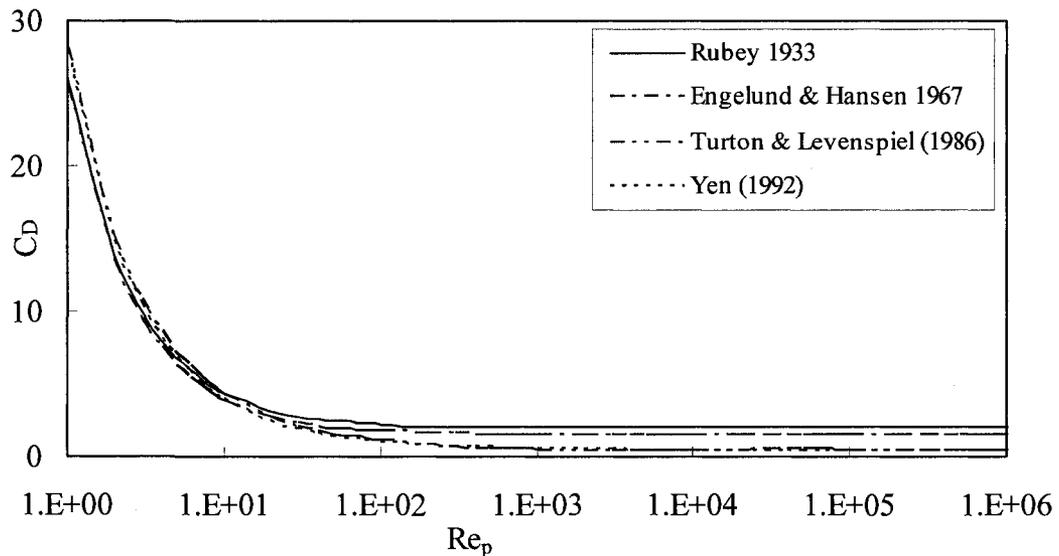


Figure 3-2: Comparison of different expressions for the drag coefficient.

Fig 3-2 shows a comparison of the different expressions for the drag coefficient. The largest differences among the diverse formulations occur when the value of Re_p is beyond, say, 10^2 , where the relative differences between the largest value of the drag coefficient (using Rubey's equation) and the smallest drag coefficient (using Yen's

equation) is about 30%. The expressions by Turton and Levenspiel (1986) and Yen (1992) produce very close results. Considering the wide use of Yen's predictor in the computation of the drag force in Lagrangian models for natural sediments, the present work adopts such formulation.

The effect of turbulence on the drag force acting on a particle was studied by Bagchi and Balachandar (2003), comparing time dependent expressions of the drag force obtained with DNS with predictions based on standard drag correlations. They concluded that no significant difference can be found between both approaches; therefore the use of the mean relative velocity to compute the drag coefficient, as it is used in this thesis, provide a reasonable accurate result for estimating the drag force.

Different expressions for the lift force and the lift coefficient can be found in literature. A comparative study of the effect of the different lift models was presented by Gupta and Pagalthivarthi (2006), where significant differences in particle trajectories (about 100% in terms of the mean particle jump length) due to the computation of lift force were identified. Wiberg and Smith's (1985) method, developed for saltating grains in water, provides a simple yet efficient expression to calculate this force with good agreement with experimental results (see below). In that study, the lift coefficient was considered to be equal to 0.2, the same value that is used in the present study and in several others studies (Wiberg and Smith, 1985; Niño and García, 1994, 1998b; Lee et al., 2000). Different authors have suggested that C_L is not constant (Gupta and Pagalthivarthi, 2006; Lee et al., 2006) but varies in narrow margins (usually between 0 and 0.5).

The value of the added mass coefficient for a single sphere moving in a still fluid at infinity is obtained in a now classic derivation in fluid mechanics; this value is $C_m = 1/2$ (see Panton, 1996, page 576). This value of C_m has been widely used to calculate the added mass force in Lagrangian models of particle motion in boundary layers (Maxley and Riley, 1993; Mei et al., 1991; Schmeeckle and Nelson, 2003), even though its range of applicability is limited to spheres that translate but do not rotate, and to flows in which the irrotational hypothesis is applicable.

Several studies have been developed to calculate the added mass coefficient under different velocity fields, mainly for spheres or sphere-like bodies. These studies have shown that an added mass coefficient equal to $C_m = 1/2$ is not adequate for all cases. Drew and Lahey (1987) derived an expression for the virtual mass force for a single sphere which is accelerating and undergoing a constant rotation in an inviscid, incompressible flow. In this case, they obtained an added mass coefficient equal to 0.5; however, they acknowledged previous works that suggested that the coefficient must depend on the particle rotation. Storti and D'Elia (2001) used a semi-numerical computation for the added mass coefficient of an oscillating hemi-sphere at different frequencies. The values for the added mass coefficient obtained varied between 0.5 and 0.14 (surge mode) and between 0.8 and 0.5 (heave mode). The first and last values represent the hemi-sphere oscillating at a very low and very high frequencies, respectively. Sankaranarayanan et al. (2002) obtained the virtual mass coefficient for rising isolated bubbles as a function of the aspect ratio of the bubbles. For spherical bubbles, the added mass coefficient varied between 0.5 and 1.5; for distorted bubbles, the variation of the coefficient was between 0.5 and 4.5, depending of the bubble volume or

area fraction. Kendoush (2005) used the velocity flow field induced by a rotating sphere to derive a value of C_m equal to 5 when a sphere rotates around its axis, but does not translate.

For simplicity, the value of the C_m was assumed to be equal to 0.5, as several particle tracking models have done previously (Niño and García, 1994, 1998b; Yeganeh et al., 2000; Schmeckle and Nelson, 2003; Lukerchenko et al., 2006)

3.2.2 Two-Dimensional Particle-Wall Collision Algorithm

García and Niño (1992) proposed a simple two-dimensional, wall-particle rebound model. The situation presented in Fig. 3-3 is considered, where a saltating particle approaching the bed at an angle θ_{in} strikes the surface that faces upstream with angle θ_b . The particle velocity is resolved into normal and tangential components with respect to the collision surface, $u_{n|in}$ and $u_{t|in}$, respectively, and it is assumed that these components are reduced after the collision, such that

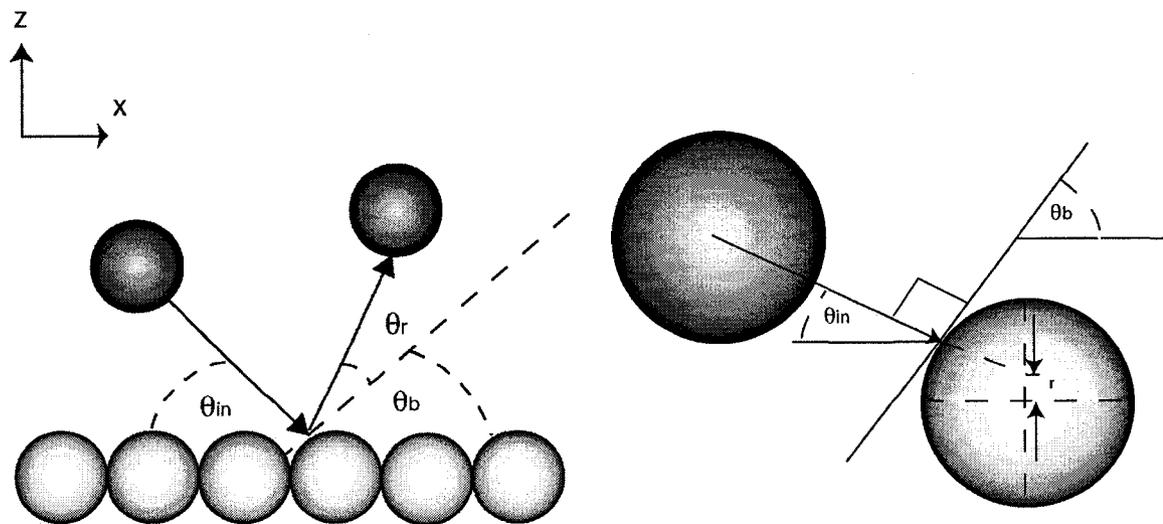


Figure 3-3: Wall collision parameters. Two-dimensional approach. Lateral view.

$$u_{n|out} = f u_{n|in} \quad (3-17)$$

$$u_{t|out} = e u_{t|in} \quad (3-18)$$

where e and f denote the restitution and friction coefficients, respectively, and $u_{n|out}$ and $u_{t|out}$ represent the velocity components of the particle immediately after the collision. In such a case, the particle rebounds with an angle θ_r , leading to:

$$\tan(\theta_r) = \frac{e}{f} \tan(\theta_{in} + \theta_b) \quad (3-19)$$

Using this equation, the particle dimensionless velocity components immediately after the collision (indicated with the superscript \wedge) can be expressed in terms of the particle dimensionless velocity components immediately before the collision (indicated with the superscript \sim) as follows:

$$\hat{u}_p = f (\tilde{u}_p^2 + \tilde{v}_p^2)^{0.5} \cos(\theta_{in} + \theta_b) \frac{\cos(\theta_r + \theta_b)}{\cos(\theta_r)} \quad (3-20)$$

$$\hat{v}_p = f (\tilde{u}_p^2 + \tilde{v}_p^2)^{0.5} \cos(\theta_{in} + \theta_b) \frac{\sin(\theta_r + \theta_b)}{\cos(\theta_r)} \quad (3-21)$$

To complete the particle-wall collision algorithm, a bed roughness representation is required to reproduce the randomness in the saltating process. García and Niño (1992) assumed that the bed is formed by uniformly pack spheres as is shown in Fig. 3-3, and considered a bed particle conditional probability density function $p(\theta_b | \theta_{in})$ of an angle θ_b for a given value of θ_{in} . Geometrical considerations dictate that the angle θ_b can vary from a maximum of 30° at the upstream side of the particle to a minimum of -30° at the downstream side of the particle. The set of angles within the range is mapped into a

set of values r_1 along a vertical line passing through the center of the particle (see Fig.3-3). The probability density of the r_1 value is assumed to be uniform, which is equivalent to assume that the considered particle has a uniform probability of being located anywhere in the bed. The following geometrical relationship among θ_b , θ_m and r_1 is established

$$r_1 = \frac{d_p}{2} (\cos(\theta_b) - \tan(\theta_m) \sin(\theta_b)) \quad (3-22)$$

3.3 Model Validation

Niño and García's (1994) model describing the particle trajectory as explained above, together with the sub-model for the rebound with a wall, and the bed roughness representation were implemented in a computational code as part of this thesis. The code can be found in Appendix A. The code provides the particle position at every time step depending of the flow conditions, given by the value of τ_* , and the particle characteristics, given by the value of R_p . The code was written using the Fortran language, in such a way that the user decides whether to include or not the effect of one specific force in the computation of the particle trajectory. The code is composed by a main program, which reads from an external file the characteristic of the flow, particle size and number of particles of the desired run, and calls several specific subroutines. Some of the subroutines were obtained from Press et al. (1992) (such as the random number generator and the Gamma function subroutines), but most of them were developed and tested by the writer.

The particle position is directly obtained by integrating the Eqs. (3-9) and (3-10) using a fourth-order Runge-Kutta method (Isaacson and Keller, 1993). The accuracy of the prediction depends only of time step used for the integration.

A simple test was designed to check the convergence of the model. A particle of 0.8 mm of diameter of natural sediment ($R_p=100$) moves along the bottom of a channel with a flow of τ_* equal to 0.056 and its trajectory is computed using a dimensionless time step equal to dt . The dimensionless height (H) and length (L) (by using the particle diameter as a length scale) of an individual jump is recorded and compared with the results obtained with a simulation using half of the original time step (dimensionless). The same procedure is applied repeatedly until the values of H and L are relatively close between one simulation and the next. The results are presented in Table 3-1.

The first results show a large difference between the values of H and L in two consecutives runs. These differences (ΔH and ΔL) decrease as the value of dt decreases, confirming the convergence of the simulation results. It is also noticed that the computational time required to run these simulations increases fast: the computational time consumed for the largest values of dt corresponds to fractions of a second; however, for smallest values of dt (10^{-4}) the simulation required around two minutes to solve the same problem (consider here that this time is for a single jump). Assuming that the “true” value of the jump length and height is given by 1.44338 and 5.76954 respectively (values obtained for dt equal to 10^{-4}), a value of dt equal to 10^{-3} is chosen in order to achieve less than 0.1% of error in the trajectory computations. This value of dt will be used in all the simulations presented in the following sections.

Table 3-1: Comparison of the dimensionless jump height (H) and length (L) due to the change in the model discretization time dt . $R_p=100$, $\tau_*=0.056$.

dt	Dimensionless Jump Height (H)	Dimensionless Jump Length (L)	Change in Height (ΔH)	Change in Length (ΔL)
0.2	1.887350	6.329901	-	-
0.1	1.630320	6.247470	-0.25703	-0.08243
0.05	1.521670	5.894048	-0.10865	-0.35342
0.025	1.476511	5.873639	-0.04516	-0.02041
0.0125	1.457386	5.785197	-0.01912	-0.08844
0.0063	1.449019	5.779334	-0.00837	-0.00586
0.0031	1.445418	5.783846	-0.00360	0.00451
0.0016	1.443925	5.774705	-0.00149	-0.00914
0.0008	1.443401	5.771622	-0.00052	-0.00308
0.0004	1.443277	5.770566	-0.00012	-0.00106
0.0002	1.443306	5.769560	0.00003	-0.00101
0.0001	1.443383	5.769542	0.00008	-0.00002

To validate the performance of the model during one jump, the experimental results provided by Niño and García (1998) were considered. In this case, natural sediment particles ($R=1.65$) of 0.6-0.8 mm of diameter ($R_p=50-80$) were placed in a horizontal channel, where water flows with shear velocities ranging from to 0.02 to 0.03 m/s ($\tau_* = 0.056-0.095$). The trajectories of the particles were recorded using a video camera placed in a specific location of the channel. Figs. 3-4 to 3-6 show the experimental results obtained from these experiments. Using the same flow and particle conditions, the model was run considering the action of: (1) buoyancy, drag, virtual mass and Magnus forces, (2) the previous forces plus lift and (3) the forces stated in (1) plus the Basset force and finally (4) all forces in (1) plus lift and Basset. The results obtained from the model show that it is necessary to include all forces to produce an accurate prediction of the particle position in the sand range of particle sizes.

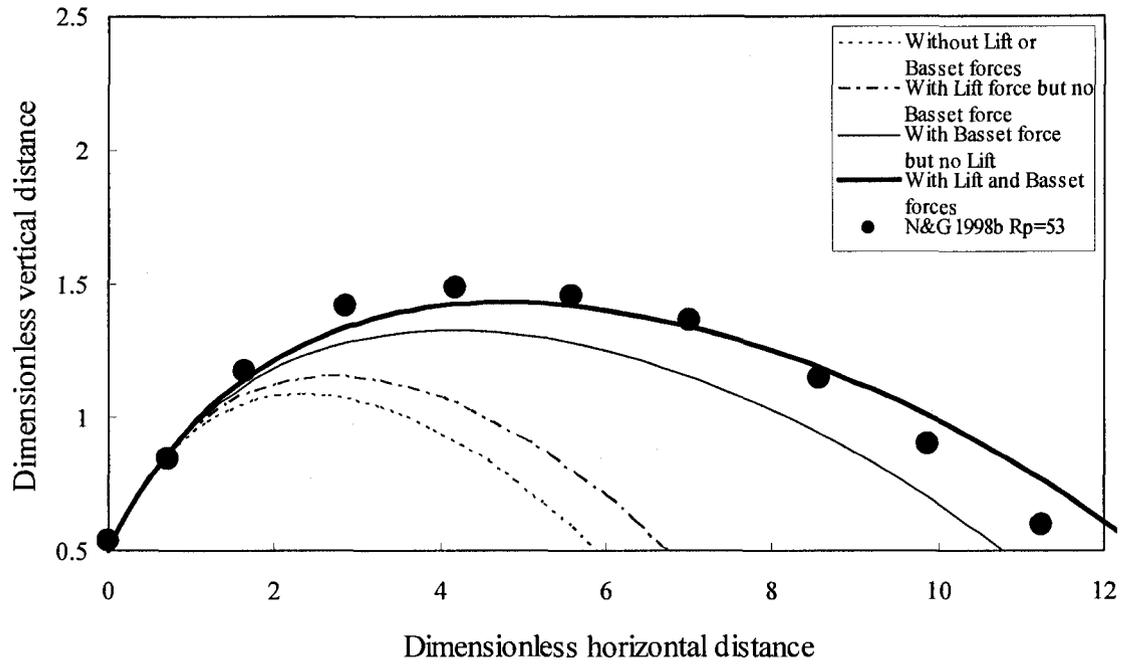


Figure 3-4: Comparison of predictions of the particle free-flight sub-model with data obtained by Niño and García (1998b) for sands. Distances are made non-dimensional by using the particle diameter. Single jump case. $d_p=0.6$ mm; $u_* = 0.025$ m/s.

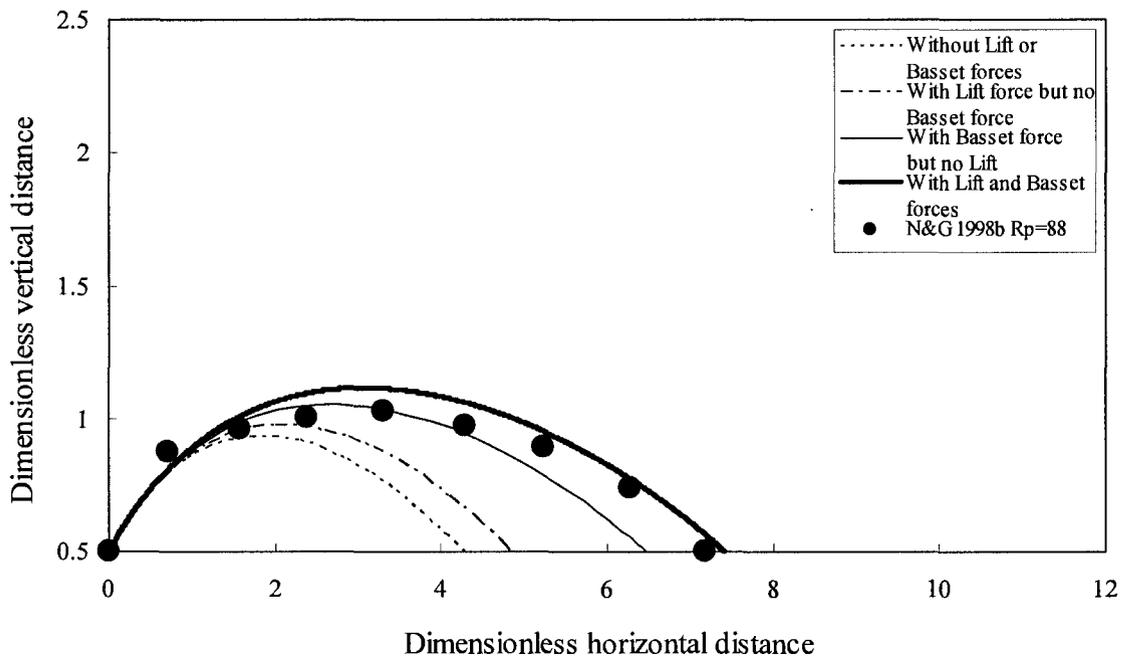


Figure 3-5: Comparison of predictions of the particle free-flight sub-model with data obtained by Niño and García (1998b) for sands. Distances are made non-dimensional by using the particle diameter. Single jump case. $d_p=0.8$ mm; $u_* = 0.027$ m/s.

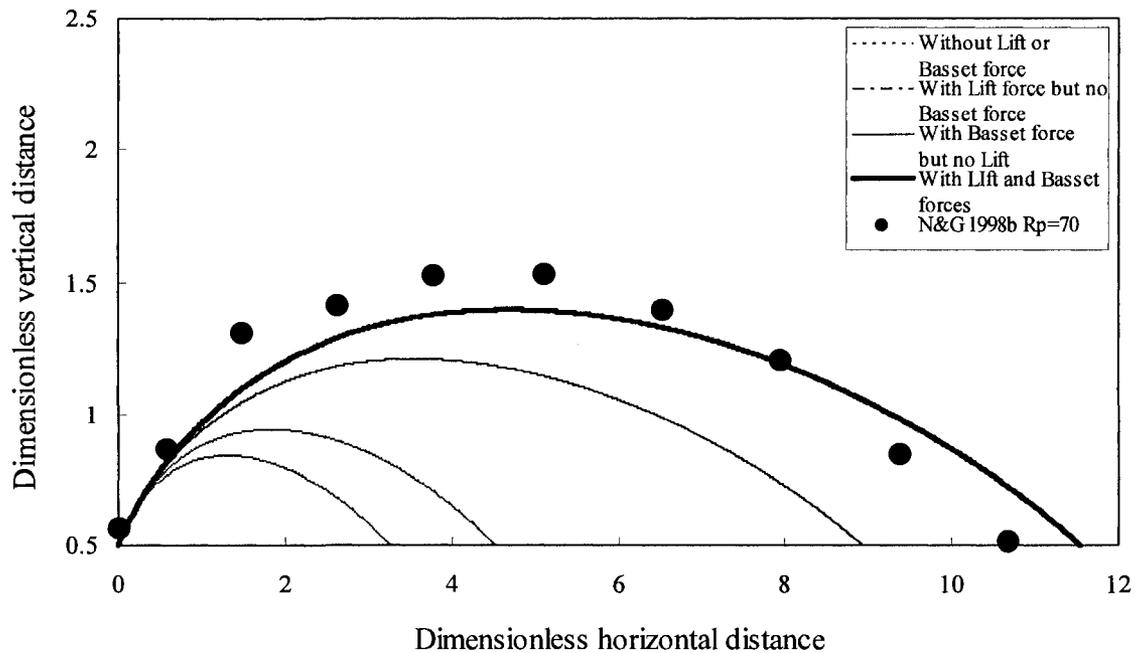


Figure 3-6: Comparison of predictions of the particle free-flight sub-model with data obtained by Niño and García (1998b) for sands. Distances are made non-dimensional by using the particle diameter. Single jump case. $d_p=0.7$ mm; $u_* =0.032$ m/s.

In a different set of experiments, Niño and García (1994) recorded the trajectories of particles of gravel size. In this case, the particle diameter considered was $d_p=30$ mm and a wall-friction velocity of 0.22 m/s. Fig. 3-7 shows the results obtained experimentally and the simulated particle trajectory when considering: (1) all forces except the Magnus force, (2) all forces but the Magnus and Basset forces and (3) all terms except the Magnus, Basset and lift forces. In this case, it is concluded that only the lift force plays a significant role in determining the particle trajectory. Therefore, the Basset and Magnus forces can be disregarded in the simulations. The previous findings agree with the conclusions obtained by Niño and García (1994) and they also confirm the fact that for larger particles the Basset force can be neglected, but not for small size particles.

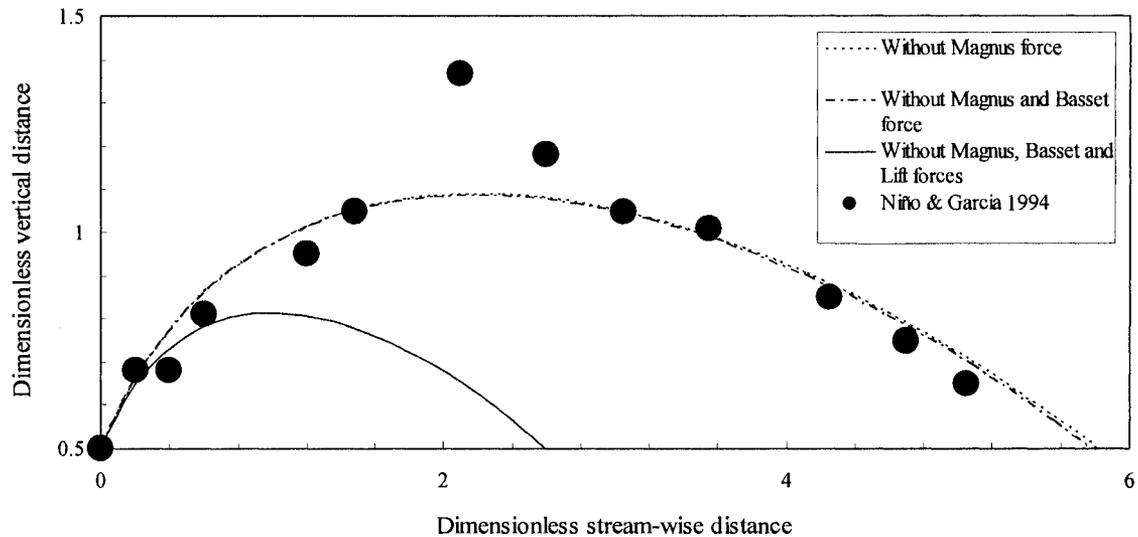


Figure 3-7: Comparison of predictions of the particle free-flight sub-model with data obtained by Niño et al. (1994) for gravels. Distances are made non-dimensional by using the particle diameter. Single jump case. $d_p=30$ mm; $u_* = 0.22$ m/s.

3.4 Relative Importance of the Forces on the Particle Trajectory

The relative importance of each intervening force in the particle trajectory is presented in Fig. 3-8, for a sand-size particle ($R_p=73$). The relative weight of each force (in absolute value) is expressed with respect to the total force exerted on the particle, as a function of the particle height relative to each jump height when it moves upwards and downwards. The figure shows the average value obtained when simulating the particle motion for a large simulation time (equivalent to more than 100 particle jumps). It is observed that the importance of each force depends on the particle position within the jump, as expected.

Drag and buoyancy are the most important forces driving the particle motion, also as expected, accounting for more than 50% of the total force acting on the particle. Despite being a force with a constant value along the particle trajectory, buoyancy changes its percentage with respect to the total force while the particle rises from the bed,

but remains relatively constant when the particle falls. The relative importance of the drag force is very similar when the particle moves upwards and downwards.

Basset and virtual mass forces become more important close to the bottom of the bed, where the particle acceleration is bigger. Basset force plays an important role when the particle starts a new jump and is very close to the bed and its importance decreases as the particle moves through the fluid. On the other hand, lift and Magnus forces play a relatively small role. Magnus force becomes more relevant when the particle moves relatively far from the wall.

When increasing the flow velocity parameters, the particle dimensionless jump height and length increase as expected (discussed in more detail in Chapter 4), but the relative importance of the different forces remains approximately the same. Fig. 3-8 also shows the relative importance of the Basset force for small particles, as it was addressed in the previous section. A study devoted to optimize the computation of this force is presented in the next section.

Fig. 3-9 shows the relative weight of each force with respect to the total force exerted on a particle with $R_p=250$. Comparison of Fig.3-8 and Fig.3-10 shows that the relative importance of the Basset force decreases as the particle becomes larger as is expected. The relative importance of the remaining forces are similar for both particle sizes, however, the Magnus forces percentage with respect to the total force presents an inflection point around z/H equal to 2, that only appears when the ratio τ_c/τ_c is greater than 3.

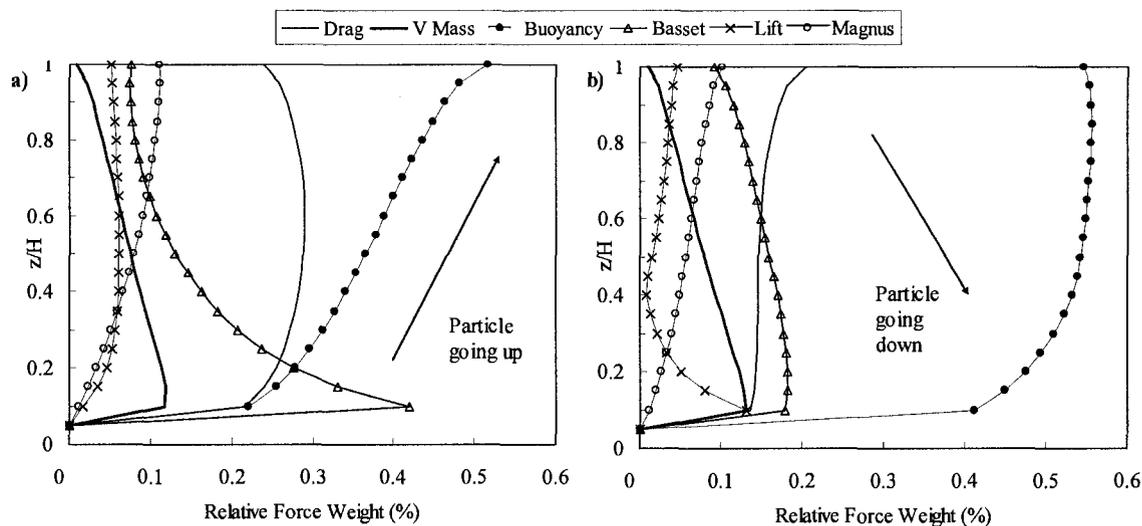


Figure 3-8: Force analysis. $R_p = 73$. $\tau_* / \tau_{*c} = 2$. a) Particle going upwards. b) Particle going downwards.

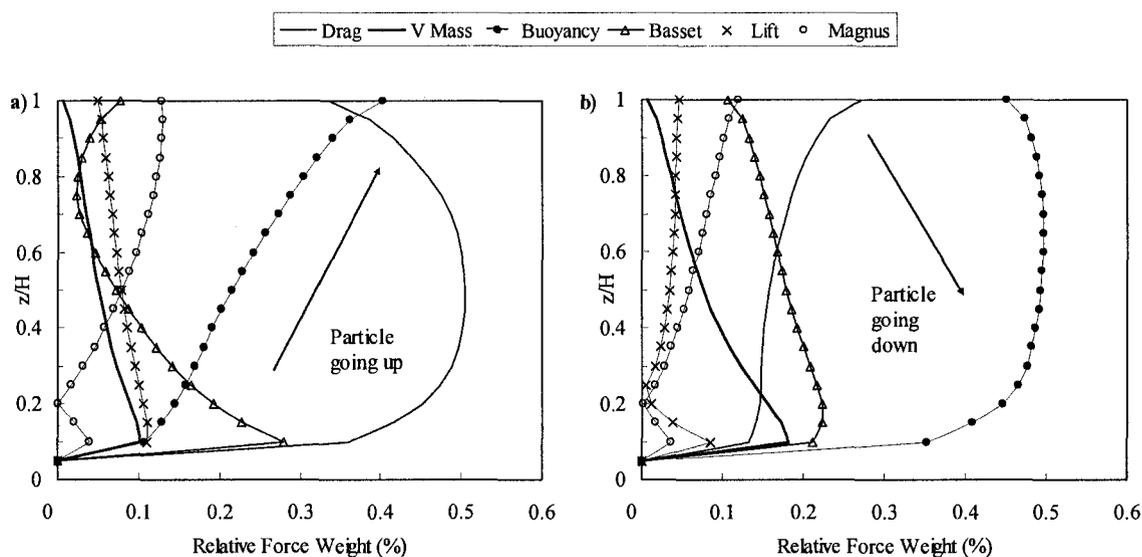


Figure 3-9: Force analysis. $R_p = 250$. $\tau_* / \tau_{*c} = 2$. a) Particle going upwards. b) Particle going downwards.

3.5 Analysis of the Magnus Force

The 2-D particle tracking model proposed by Niño and García (1994) uses an empirical expression to compute the particle rotation (see Eq. (3-12)) and this effect is embedded in the calculation of the Magnus force. Extending the use of this model beyond

its range of validity (even for particles sizes slightly smaller/larger than the ones used to obtain the empirical expression) may or may not provide accurate results. To evaluate this, the results obtained via numerical simulation can be compared with the experimental data available, in terms of the mean particle jump length (L) and height (H) and the average particle stream-wise velocity (\bar{u}_p).

In the case of smaller particles of $R_p=73$ (i.e., particles of the range of Niño and García's (1998a,b) experiments, where the empirical expression for the particle rotation was obtained), the results obtained from the numerical simulations and from the experiments presents very good agreement (see Section 4.4 for more detail).

For larger particles of $R_p=250$ (i.e., particles of the range of Lee and Hsu's (1994) experiments) the simulation results and the experimental data available present two different trends (see Fig. 3-10). The experimental data show the expected trend between the dimensionless parameters (H , L , and \bar{u}_p) and the friction velocity τ_* : as the value of τ_* increases, the flow velocity increases and the particle is capable of describing jumps that are higher and longer, and therefore the value of the dimensionless parameter studied increases. In turn, the numerical results show that there is a value of τ_* after which the dimensionless particle stream-wise velocity, jump length, and height no longer increase.

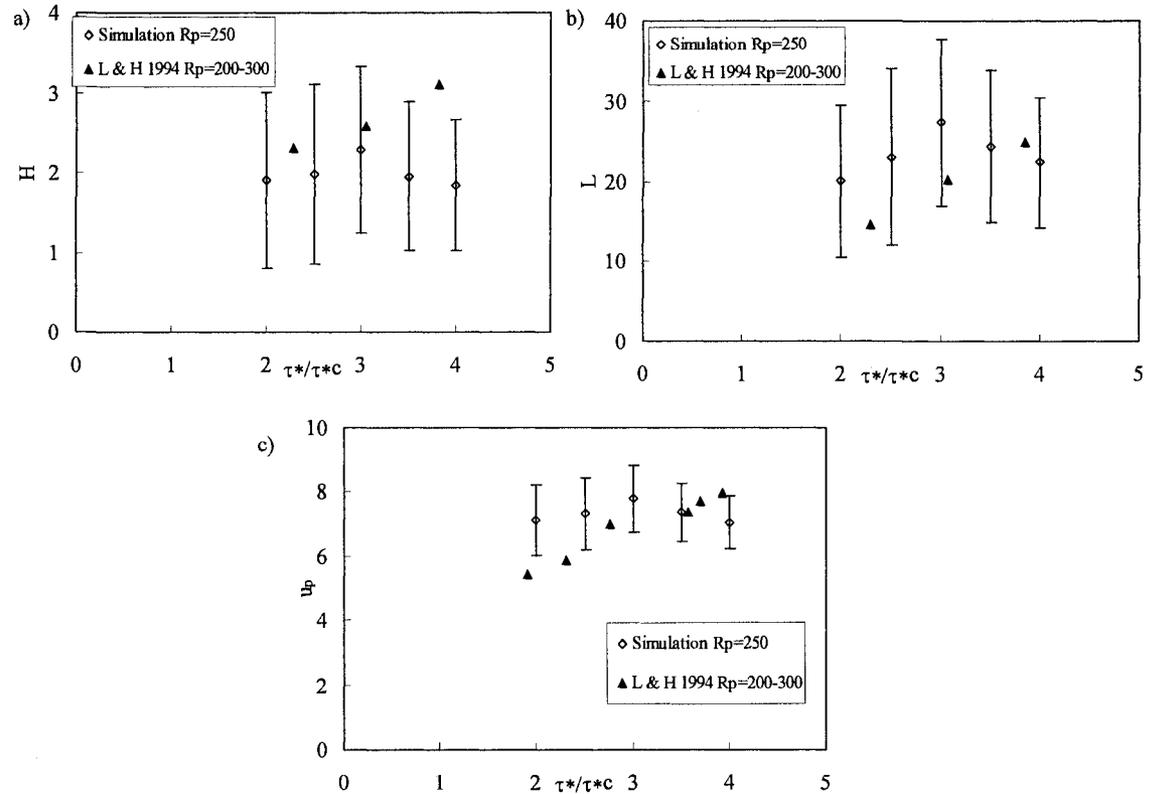


Figure 3-10: Comparison of simulations with experimental data for the case of a particle moving in a flume and rebounding with the wall (2-D model). The figure shows results associated with: a) the particle jump height (H), b) the particle jump length (L) and c) the particle stream-wise mean velocity. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. $R_p=250$.

The importance of each intervening force in the particle trajectory was studied, in an attempt to explain the model behavior. When switching on and off the lift, Basset, and virtual mass forces, a change in the magnitude of the dimensionless parameters (H , L and \bar{u}_p) is observed, but there is still a value of τ^* after which the dimensionless parameters no longer increase. However, when turning off the Magnus force, the following results are obtained: as the friction velocity increases, the value of each dimensionless parameter (H , L and \bar{u}_p) increases. Therefore, the Magnus force is responsible for the model behavior.

To study the effect of the shear stress on the Magnus force, Fig. 3-11 shows the relative weight of this force with respect to the total force for different friction velocity values, as a function of the particle position. From Eq. (3-6), the sign of the Magnus force depends on the difference between the particle rotation (obtained from the empirical expression) and the value of the derivative of the stream-wise fluid velocity with respect to the wall-normal direction, evaluated at the elevation where the particle is located.

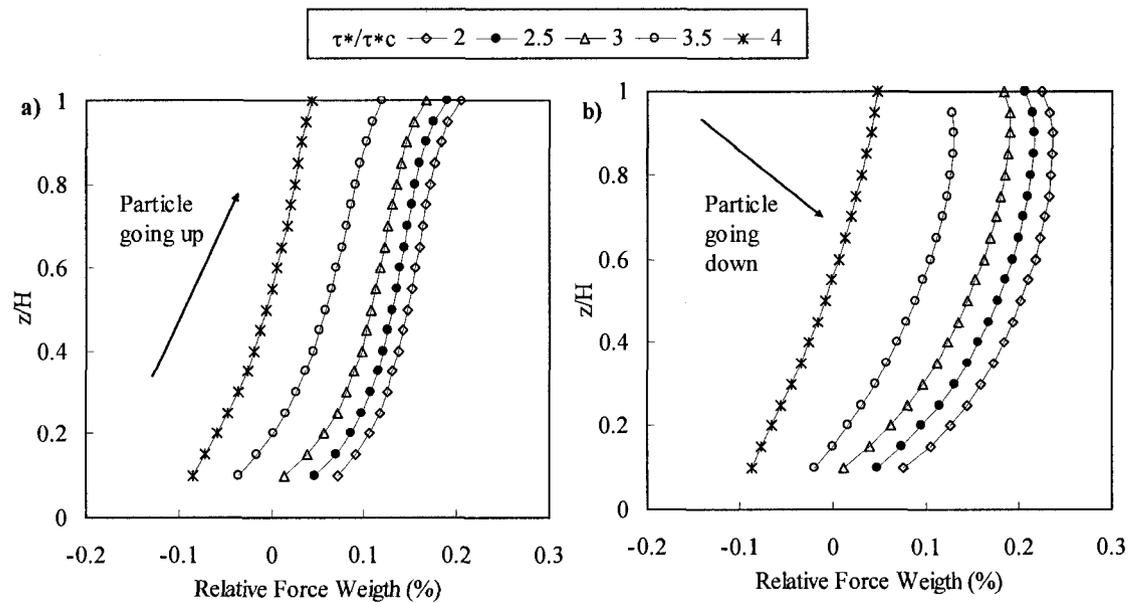


Figure 3-11: Variation of the Magnus force along the wall-normal direction, for different values of the relative shear stress. a) Particle going upwards. b) Particle going downwards.

For smaller values of the friction velocity (τ_*/τ_{c*} less than 3), the value of particle rotation is larger than the value of the fluid velocity derivative, at every point in the vertical. In this case, the sign of the Magnus force is positive and it is maintained across the wall-normal direction. As the value of the shear stress increases, the rotation of the particle decreases, as calculated by using Eq. (3-12) and eventually, the fluid velocity derivative close to the bed (where this value is larger) becomes larger than the particle rotation. Thus, the sign of the Magnus forces switches from positive to negative

acceleration when the particle is close to the bed. When the particle moves away from the wall, the value of the fluid velocity derivative becomes smaller and the Magnus force becomes positive again. This change of sign significantly affects the particle acceleration, and therefore the particle trajectory: if the Magnus force has a positive sign, it increases the particle acceleration making the particle travel further away. On the other hand, when the Magnus force becomes negative, the particle acceleration decreases and so does the jump length and height.

It is concluded that the use of the empirical relation given by Niño and García (1994) turns out to be valid only for particle with R_p numbers between 60 and 90. Therefore, in order to extend the 2-D model for a larger range of particle sizes, it is necessary to provide a theoretical expression of the particle rotation to compute more accurately the Magnus force. From the previous results, the computation of the rotation of the particle is a key factor to correctly estimate the value of the Magnus force. A model to calculate the angular velocity of the moving particle is included in the next chapter.

3.6 A New Algorithm to Compute the Basset Term

To circumvent the improper nature of the Basset integral, several authors have used the Brush et al. (1964) approach (Niño and García, 1998b, Lukerchenko et al., 2006) as seen in Eq. (3-11). Another standard approach to deal with the singularity of the Basset term is to employ specific quadratures such as the Second Euler-Maclaurin summation formula (Press et al., 1992), hereafter referred to as the SEML.

Tatom (1988) proposed to approximate this term using the Riemman-Liouville integral definition of a semi-derivative (SD). The Basset term is calculated as follows:

$$\int_a^t \frac{du_r}{\sqrt{t-\tau}} d\tau = \Gamma\left(\frac{1}{2}\right) \frac{d^{-0.5}\left(\frac{du_r}{dt}\right)}{d(t-a)^{-0.5}} \quad (3-23)$$

where $\Gamma(\cdot)$ represents the gamma function (Abramowitz and Stegun, 1970), and a denotes the lower limit of integration, equal to 0 in the classical definition of the Basset force. The semi-derivative term is calculated using an infinite series form, as found in Oldham and Spanier (1974):

$$\frac{d^b f}{[d(t-a)]^b} = \lim_{N \rightarrow \infty} \left\{ \left(\frac{t-a}{N} \right)^{-b} \frac{1}{\Gamma(-b)} \sum_{k=0}^{N-1} \frac{\Gamma(k-b)}{\Gamma(k+1)} f\left(t - \frac{k(t-a)}{N}\right) \right\} \quad (3-24)$$

where b is an arbitrary value. In this case the value of b is equal to -0.5. This methodology was adopted in this work with the objective of reducing the computational cost of the Basset term. To the best of the writer's knowledge, this way of computing the Basset term has not been incorporated into particle models so far.

In order to demonstrate the advantages of the proposed methodology, the convergence of the semi-derivative approach (SD) is compared with that of the SEML, taken verbatim from Press et al. (1992). To that end, two arbitrary functions were selected, $u_r(\tau) = \tau^3/3$ (or, $du_r(\tau)/d\tau = 2\tau$) and $du_r(\tau)/d\tau = \cos(\tau)$ to test the convergence of both techniques. Both cases have analytical solutions. The Basset integral was evaluated between $a = 0$ and $t=100$ for the first function, and between $a = 0$ and $t=100 \pi/2$ for the second function. The comparison of results obtained with the SD and the SEML are presented in Tables 3-2 and 3-3 for the first and second function, respectively, where it is possible to notice that the SEML provides consistently larger

values of the relative error than the SD method, for the same number of points/terms. Therefore, the SD converges faster than the SEML. Naturally, the faster convergence for a given relative error of the SD leads to savings in the computational cost of the Basset term.

Table 3-2: Relative error of the quadratures associated with the semi-derivative approach and the Second Euler-Maclaurin summation formula, when used to compute the Basset integral with $u_r(\tau) = \tau^3 / 3$ (for different numbers of points/terms). The analytical solution for this case is given by: $\frac{-2}{15} \sqrt{t-\tau} (8t^2 + 4t\tau + 3\tau^2)_0^t$, which was evaluated at $t=100$. The relative error was computed as the absolute value of the difference between the numerical and the analytical results of the Basset integral multiplied by 100 and divided by the analytical value, for both methodologies.

Points/ Terms	Semi- derivative method	Second Euler- Maclaurin summation formula	Percentage of error semi- derivative	Percentage of error Euler- Maclaurin
81	107,490.35	99,928.89	0.7722	6.3167
243	106,941.09	102,783.03	0.2573	3.6409
729	106,758.12	104,425.68	0.0857	2.1009
2,187	106,697.15	105,373.07	0.0286	1.2127
6,561	106,676.83	105,919.86	0.0095	0.7001
19,683	106,670.05	106,235.50	0.0032	0.4042
59,049	106,667.80	106,417.74	0.0011	0.2334
177,147	106,667.04	106,522.95	0.0004	0.1347
531,441	106,666.79	106,583.69	0.0001	0.0778
1,594,323	106,666.71	106,618.77	0.0000	0.0449

The convergence of the SD approach was also compared with that of the Brush et al.'s method (the two terms) using the Simpson quadrature for the first term (see Appendix B). An arbitrary quadratic function aimed at mimicking conditions of relative velocity for a saltating particle was selected, for which a simple analytical solution for the integral was available. For 5,000 points/terms for instance, the relative error in the SD computations as compared to the analytical solution is 0.015, while it is 0.054 for the Brush et al.'s method using Simpson quadrature. Those errors reduce to 0.002 and 0.020,

respectively, for 50,000 points/terms, showing that the SD methodology offers a faster convergence. This result is noteworthy since alternative techniques for improper integrals often employ standard methods such as Simpson quadrature to undertake the final integration after a change of variables (Press et al., 1992).

Table 3-3: Relative error of the quadratures associated with the semi-derivative approach and the Second Euler-Maclaurin summation formula, when used to compute the Basset integral $du_r(\tau)/d\tau = \cos(\tau)$ (for different numbers of points/terms). The analytical solution for this case is given by:

$$\int_0^t \frac{\cos \tau}{\sqrt{t-\tau}} d\tau = -\sqrt{2\pi} \left\{ \cos(\tau) C \left[\sqrt{\frac{2}{\pi}} \sqrt{t-\tau} \right] + \sin(\tau) S \left[\sqrt{\frac{2}{\pi}} \sqrt{t-\tau} \right] \right\}_0^t, \text{ with the}$$

Fresnel integrals given by $C(\tau) = \int_0^\tau \cos \left(\frac{\pi z^2}{2} \right) dz$ and $S(\tau) = \int_0^\tau \sin \left(\frac{\pi z^2}{2} \right) dz$; this integral

was evaluated at $t=100\pi/2$. The relative error was computed as the absolute value of the difference between the numerical and the analytical results of the Basset integral multiplied by 100 and divided by the analytical value, for both methodologies.

Points/ Terms	Semi- derivative method	Second Euler- Maclaurin summation formula	Percentage of error semi- derivative	Percentage of error Euler- Maclaurin
81	1.75815554	0.36399574	40.309	70.951
243	1.42522996	0.76393412	13.740	39.035
729	1.31140474	0.97209484	4.656	22.422
2,187	1.27263337	1.09093575	1.562	12.938
6,561	1.25959918	1.15946342	0.522	7.469
19,683	1.25524151	1.19902245	0.174	4.312
59,049	1.25378748	1.22186151	0.058	2.490
177,147	1.25330265	1.23504761	0.019	1.437
531,441	1.25314102	1.24266058	0.006	0.830
1,594,323	1.25308714	1.24705588	0.002	0.479

Finally, the SD approach is compared with the approximation proposed by Brush et al. (1964) including the two terms for a single particle jump. The objective with this comparison is to show that both methodologies produce similar results for the Basset force at a given point in a jump. Therefore, it presents the results of the Basset force

evaluated after the particle has traveled a jump and is about to hit the bed. In this test, the Basset force is computed for every time step until the jump is completed. Given the intrinsic differences between results from two quadratures in general, and the accumulation of differences every time the Basset force is computed in each time step during the jump, the results were not expected to be identical. In this test case, all forces except lift and Magnus were included, and involved a particle with $R_p=53$ moving in a flow with $\tau_*=0.056$. Then, for 10,000 terms in the SD methodology, the differences between the values of the Basset force for both techniques are 2.61% for the stream-wise component of the force, and 1.21% for the wall-normal component; these differences remain almost the same for larger numbers of terms/points. Notwithstanding these small differences in the values of the forces, the differences in the values of jump length and height between both methodologies are smaller than 0.4% and 0.17%, respectively. This indicates that both methodologies provide similar results.

Overall, the SD methodology reduces the computational cost by approximately 20% as compared with the alternative methods mentioned above.

3.6.1 Memory Time Concept

The integral associated with the Basset force must be performed, by definition, between the beginning of times and the current time of computation. However, in most previous works on particle motion close to walls, the computation of the Basset force was re-set to zero after each rebound against the wall, the justification of which does not seem to be based on clear-cut evidence. Between these two limiting procedures, it seems plausible that collisions of the particle with the wall may likely eliminate correlation in

the development of the boundary layer for jumps separated by “long” distances. Therefore, it seems reasonable to assume that only the most recent events influence the current particle velocity (see also Mei et al., 1991). Considering the previous argument, a new practical definition of the Basset integral is proposed:

$$\int_0^t \frac{du_{ri}}{\sqrt{t-\tau}} d\tau \approx \int_{t-T_{back}}^t \frac{du_{ri}}{\sqrt{t-\tau}} d\tau \quad (3-25)$$

where T_{back} represents the time interval during which the history of the particle affects the current particle velocity. This definition of the T_{back} is compatible with the use of the SD approach given by Eq. (3-25). (Incidentally, a similar concept was recently and independently published by Dorgan and Loth (2007), where a “window model” redefines the integration time of the Basset term.)

The “memory time period” should be related to relevant particle time scales and/or flow characteristic times. The behavior of particles moving in fluids is controlled by the particle time scale τ_p – particle relaxation time – which is defined as $\tau_p = \frac{\rho_s d^2}{18\nu}$ (Crowe et al., 1998). In order to define a flow characteristic time, the flow coherent structures that may affect the particle trajectory close to a boundary are considered. Experiments with sediment particles by, for example, Grass (1970), Kaftori et al. (1995), and Niño and Garcia (1996) have indicated that for smooth walls these structures maintain their identity for as long as 60 to 80 wall time units (defined by using d_p^2/ν as a time scale), and extend vertically a distance of about 100 wall units (defined by using

ν/u_* as a length scale). A flow characteristic time: $\tau_f = 80 \frac{\nu}{u_*^2}$, can therefore be assumed.

A third time scale can be obtained by manipulating the expression of the Basset force in the same way used to obtain the particle time scale τ_p (see Crowe et al., 1998 page 23; Groszmann and Rogers, 2004). (Those studies employ τ_p as a measure of the particle adaptation time regardless of whether the particle moves in the Stokes range or not. This is also the way it is used in this work, representing a time scale.) The particle response time is directly related to the drag force. The acceleration of the particle $\frac{d\vec{u}^p}{dt}$

in the Stokes range is given by:

$$\frac{d\vec{u}^p}{dt} = \frac{18\mu}{d_p^2 \rho_s} (\vec{u}^p - \vec{v}^p) \quad (3-26)$$

The leading factor in of the relative particle velocity $(\vec{u}^p - \vec{v}^p)$ of the right-hand side has units of time^{-1} , and corresponds to the inverse of the particle response time. Analogously, the particle acceleration due to the effect of the Basset force is given by:

$$\frac{d\vec{u}^p}{dt} = \frac{9}{2d_p} \frac{\rho}{\rho_s} \sqrt{\frac{\nu}{\pi}} \int_0^t \frac{d\vec{u}_r^p}{d\tau} \frac{d\tau}{\sqrt{t-\tau}} \quad (3-27)$$

Using the same velocity and length scale as before (u_* and d_p , respectively) Eq.

(3-27) turns into:

$$\frac{du_p^*}{dt^*} = \frac{9}{2u_*} \frac{\rho}{\rho_s} \sqrt{\frac{\nu}{\pi}} \int_0^{t^*} \frac{du_r^*}{d\tau} \frac{d\tau}{\sqrt{t^* - \tau}} \quad (3-28)$$

The term in front of the integral has units of time^{0.5}; therefore the representative time scale for this force can be computed as the square of this term: $\tau_B = \frac{8l}{\pi(R+1)^2} \frac{\nu}{u_*^2}$.

Adopting usual values for R , it is possible to obtain $\tau_B \approx 4 \frac{\nu}{u_*^2}$, which is about 20 times smaller than the fluid time scale derived before. It is noteworthy that the structure of both flow time scales is the same; after all, coherent structures are also involved in the development of the boundary layer due to changes in the relative velocity.

Two methods are proposed herein to define more precisely the value of T_{back} devised on Eq. (3-25); while one of the methods can be applied to the motion of particles at any location, the second method uses the fact that particles “saltate” close to a solid boundary.

Model 1: The memory time T_{back} is obtained by choosing the biggest value between the particle relaxation time, the flow characteristic time, and the Basset time scale, and then multiplying it by a factor C .

Model 2: An alternative definition for T_{back} can be obtained by considering the memory time in terms of a number of previous jumps. If N_{back} is the number of jumps considered in the computation of the Basset term, the value of T_{back} is defined from the beginning of the N_{back}^{th} jump. The value of T_{back} can be estimated by multiplying N_{back} by the mean time duration of the jumps, therefore an equivalent value of C can be calculated.

In principle, these methods could provide different values of T_{back} , because they are related to different integration times. In order to compare the results obtained with

both methods, the code was run using the SD approximation for the Basset force along a simulation time equal to 1,000 non-dimensional units. (This time provides a significantly large number of jumps to study, independently of the initial conditions of the particle.) Values for $R_p=100$ and $\tau_*=0.056$ were used. Initial conditions for both methods were identical. Since smaller particles “fly” for longer periods, these 1,000 time units are equivalent for these flow and particle conditions to approximately 500 particle jumps. The stochastic collision model causes the height and length of each jump to have different values. Figs. 3-12 present the results obtained in terms of jump length and height, respectively. For these simulations, a value of C equal to 100 represents the computation of the Basset term from the beginning of the particle movement ($t - T_{back}$ equal to 0). Using just a number of previous steps (which does not possess any specific physical meaning) is equally efficient to fixing a number of particle or fluid time scales (whichever is larger), clearly indicating that there is a loss of correlation between current and previous jumps.

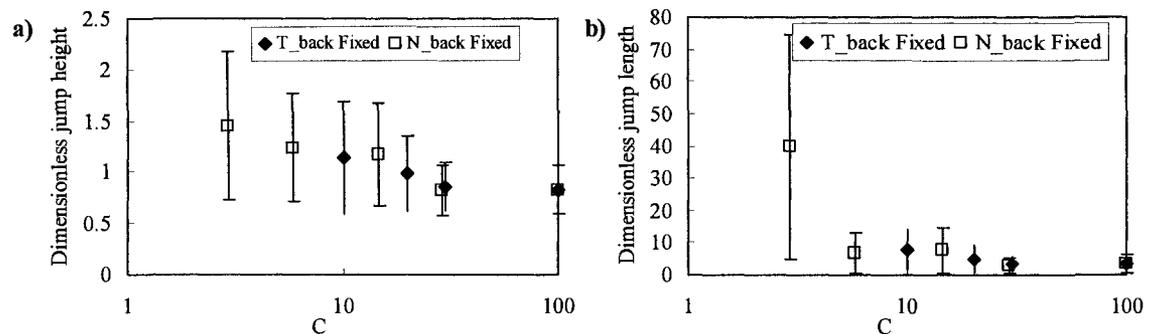


Figure 3-12: Comparison of the average jump height (a) and average jump length (b) computed using the semi-derivative approximation for the Basset force. Multiple jumps (stochastic collisions). $R_p=100$. $\tau_*=0.056$.

Three additional sets of simulations were performed covering a wide range of values of τ_* within the range of small particles and a long simulation time (1,000 dimensionless time units). The flow velocity was varied between 0.07 and 0.5 m/s in the simulations. A reference simulation was defined by calculating the Basset integral from the beginning of the particle motion, represented with a value of $N_{back} = 500$. The reference simulation provides the value of the Basset integral classically defined, as the integration is performed on the entire particle history. Each variable of study (particle velocity component and jump length and height), was specified an interval consisting of one standard deviation with respect to the average value of the reference simulation (drawn as dashed horizontal lines in each figure). This interval helps determining the degree of departure of the variables when the Basset integral is computed with a limited number of previous jumps.

The first set of runs used a value of R_p equal to 100 and a value of τ_* equal to 0.056. Results are presented in Figs. 3-13. Fig. 3-13a shows that for a value of N_{back} equal to 25 the average dimensionless stream-wise velocity is outside of the reference interval. However, all the remaining plots show that for N_{back} equal to 25 the average variable value is inside of this interval, and the range is close to the reference simulation. Consequently, values of N_{back} equal to 25 or 50 particle jumps seem to be “appropriate” under these conditions. For these numbers of previous jumps, the computational time is reduced to only 70% of the time of the reference simulation.

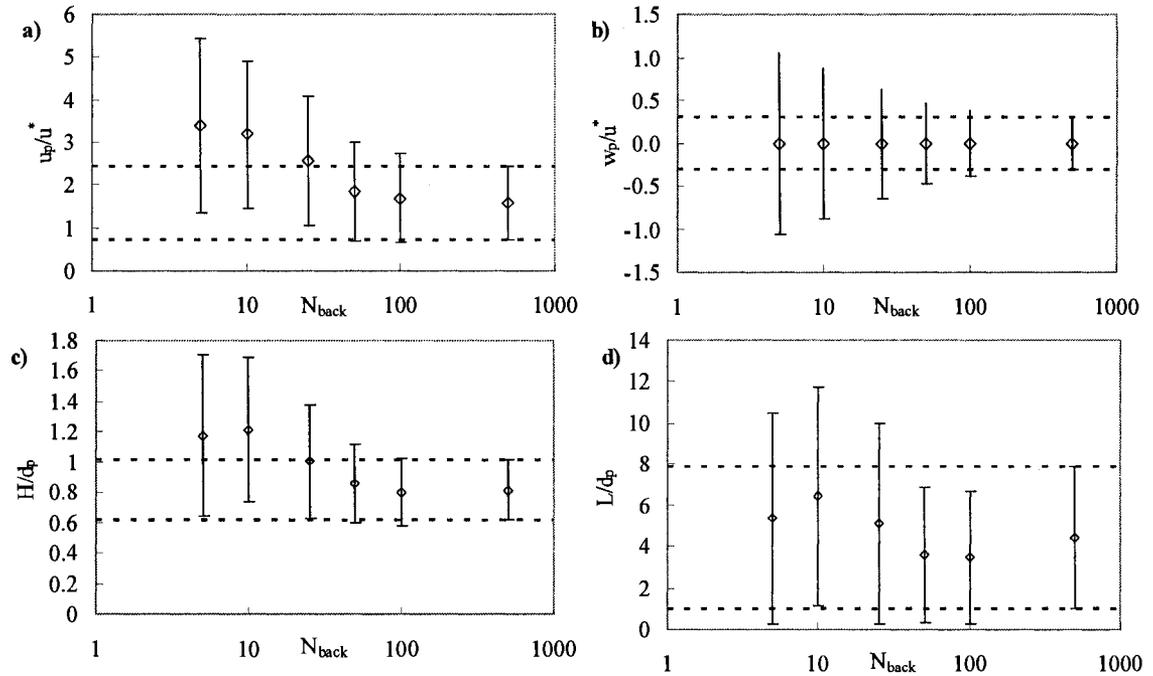


Figure 3-13: Dimensionless jump variables versus number of jumps in the past (N_{back}) included in the Basset force computation. Multiple-jump simulation using stochastic collision model. $R_p=100$. $\tau_*=0.056$. Dotted lines represent the reference range. a) Dimensionless stream-wise component of the velocity. b) Dimensionless wall-normal component of the velocity. c) Dimensionless jump height. d) Dimensionless jump length.

The second set used a value of R_p equal to 100 and a value of τ_* equal to 0.11.

The results are presented in Figs. 3-14. Therein, for all the parameters studied, the reference range is large enough to contain all the values of N_{back} evaluated. However, selecting a number of jumps between 10 and 25, it is possible to argue that a satisfactory compromise between computing time reduction and representation of the integral has been achieved.

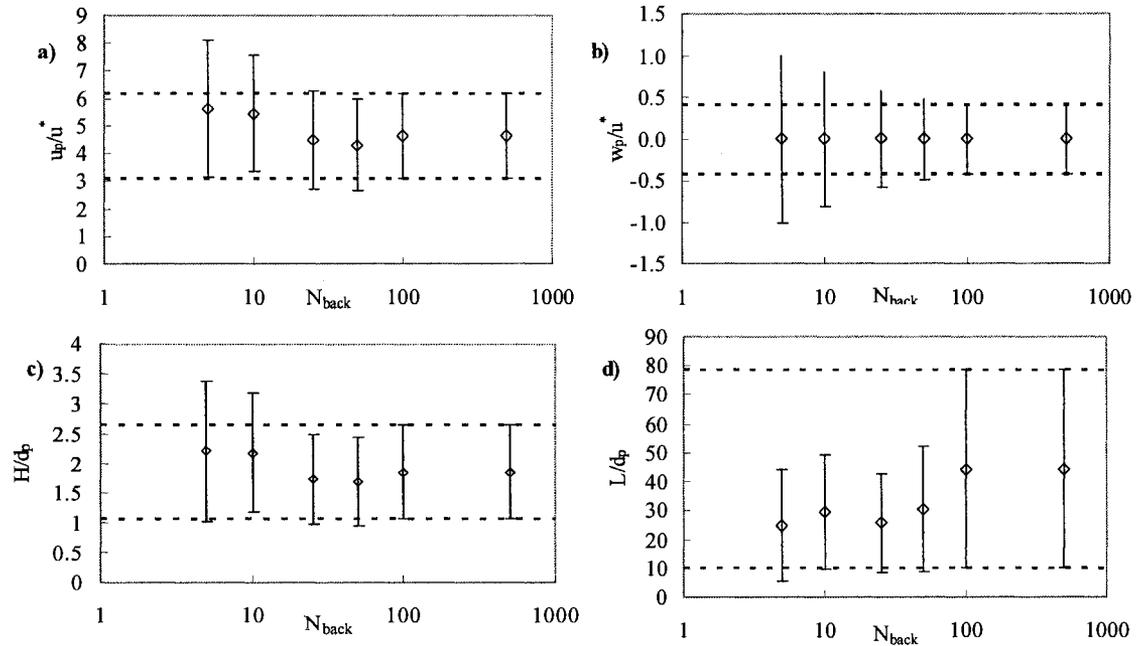


Figure 3-14: Dimensionless jump variables versus number of jumps in the past (N_{back}) included in the Basset force computation. Multiple-jump simulation using stochastic collision model. $R_p=100$. $\tau_*=0.11$. Dotted lines represent the reference range. a) Dimensionless stream-wise component of the velocity. b) Dimensionless wall-normal component of the velocity. c) Dimensionless jump height. d) Dimensionless jump length.

The third and last set of runs used a value of $R_p=100$ and a value of $\tau_*=0.28$, as shown in Fig. 3-15. The total number of jumps obtained for the reference simulation is only 22, because the velocity of the flow is large. It is found that the average jump length, height and velocity components lie within of the reference interval for relatively low values of N_{back} (about 10 jumps).

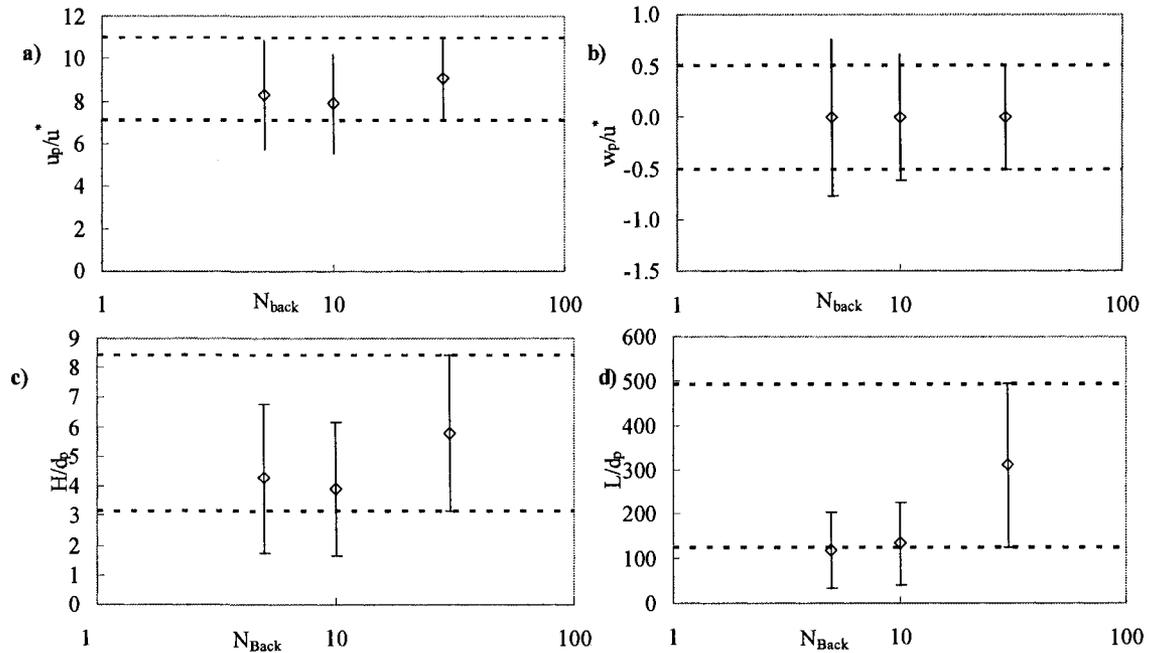


Figure 3-15: Dimensionless jump variables versus number of jumps in the past (N_{back}) included in the Basset force computation. Multiple-jump simulation using stochastic collision model. $R_p=100$. $\tau_* = 0.28$. Dotted lines represent the reference range. a) Dimensionless stream-wise component of the velocity. b) Dimensionless wall-normal component of the velocity. c) Dimensionless jump height. d) Dimensionless jump length.

3.6.2 Time Reduction

The simulation time directly depends on the value of N_{back} selected. Table 3-4 shows the reduction in time for a simulation of 500 particle jumps using different values of N_{back} , as compared with the computational time used in the reference simulation. Reductions of 10-20% can be achieved employing about 25 to 50 previous jumps.

Table 3-4: Decrease in simulation time (in percentage) versus N_{back} . Run considers 500 particle jumps.

N_{back} value	Relative total time	Time savings (%)
5	0.61	38.5
10	0.64	36.3
25	0.80	19.7
50	0.92	8.1
100	1.00	0.0

3.6.3 Global Efficiency of the New Basset Force Approximation

The use of a semi-derivative (SD) approximation to calculate the Basset integral was found to reduce the computational time by 20-30% as compared with Brush *et al.*'s method. In addition, this method is simple to implement. In addition, it was found that the approximation of the Basset force using the T_{back} concept reduces the computational time of the run to about 70 to 90 percent of the original simulation time (time reductions of 30 to 10%). These time reductions may save days or weeks in large simulations, such as those shown in Chapter 5.

3.7 Assessment of Values for the Friction and Restitution Coefficients

Due to the big variability on the values of the friction and restitution coefficients found in the literature (see Section 2.7 of this thesis) a sensitivity analysis of both parameters was performed. The effect of these two coefficients on the particle trajectory for a rough bed was studied herein by comparing the length and height of particle jumps after hitting the wall (see Figs. 3-16 and 3-17). A particle of $R_p = 73$ was used and the García and Niño (1992) bed representation roughness model. The simulation model was run long enough to have statistical meaningful values of particle jump length and height. From Eq. (3-17), increasing the value of the restitution coefficient produces a relative increment in the normal component of the particle velocity leading to higher jumps. On the other hand, increasing the friction coefficient leads to a relative increment of the tangential component of the particle velocity after the collision.

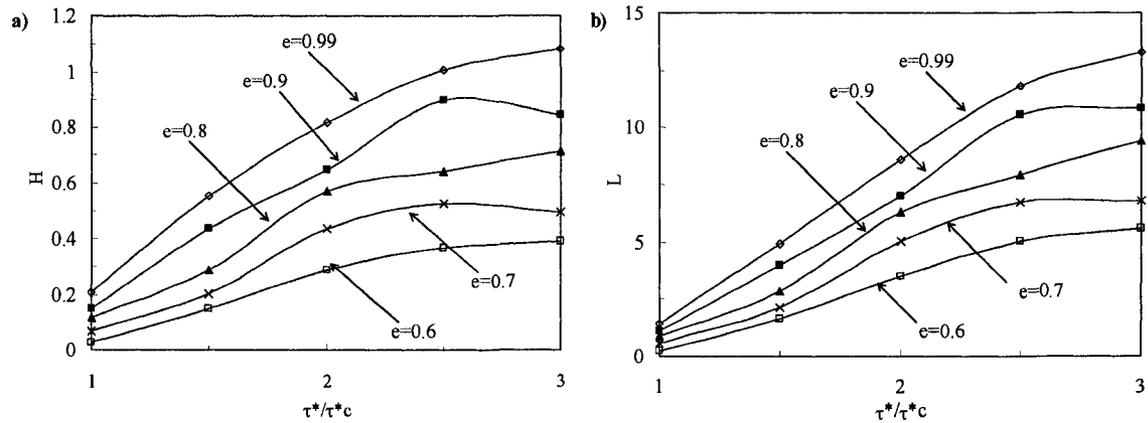


Figure 3-16: Effect of the restitution coefficient e on the particle trajectory after colliding with the wall. The value of the friction coefficient f is considered constant and equal to 0.4. $R_p = 73$. a) Dimensionless particle jump height, averaged over 100 jumps. b) Dimensionless particle jump length, averaged over 100 jumps.

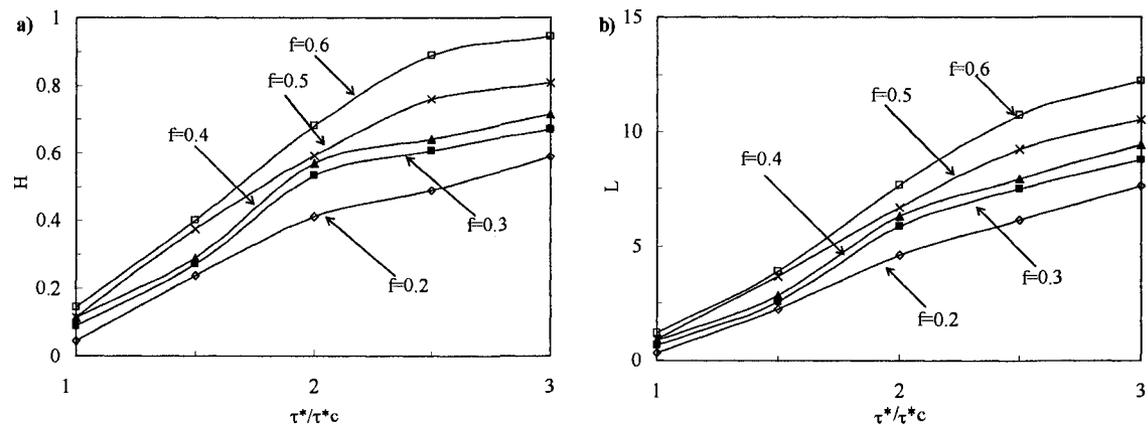


Figure 3-17: Effect of the friction coefficient f on the particle trajectory after colliding with the wall. The value of the restitution coefficient e is considered constant and equal to 0.8. $R_p = 73$. a) Dimensionless particle jump height, averaged over 100 jumps. b) Dimensionless particle jump length, averaged over 100 jumps.

As expected, increasing the restitution coefficient (e) produce an increase in the wall-normal particle velocity after the collision, leading to jumps which are greater in height (Fig. 3-16a) and in length (Fig. 3-16b). In this case, the bed roughness redefines the normal and tangential velocity of the particle after the rebound, affecting both jumps length and height at the same time. The same effect is seen in Fig. 3-17, where an increase in the friction coefficient not only leads to longer jumps, but also higher particle

jumps. The change in the average jump length and height values by changing the friction coefficient is slightly smaller than the change obtained by adjusting the restitution coefficient.

As shown in Figs. 3-16 and 3-17, by tuning the values of the friction and restitution coefficients it is possible to adjust the simulation values of the particle jump length and height, and eventually match the numerical results with any experimental data set. However, by changing arbitrarily these coefficients, any physical consideration of the particle shape or material that those values contain is completely lost.

CHAPTER 4

THREE-DIMENSIONAL PARTICLE TRACKING MODEL

A new 3-D particle trajectory model is presented in this chapter. This model, which includes the description of the particle translational and angular velocity in the 3-D space, and it was implemented computationally by extending the previously tested 2-D code.

The collision of the particles with the wall in 3-D is addressed in this chapter. A comparison between existing collision sub-models is included in this chapter as well. A new 3-D bed roughness representation is also presented here, and it is compared with other sub-models.

The code also includes the simultaneous motion of several particles at the same time, and inter-particle collisions. Results obtained from this code are presented at the end of this chapter.

4.1 Three-Dimensional Particle Trajectory Model

Using all the information provided in Chapter 3, it is relatively easy to create a 3-D particle trajectory model. From the Mei et al.'s expression, Eq. (3-3) was obtained by neglecting the second order terms, averaging over the turbulence and increasing the range of applicability of the drag force (by the use of a non linear expression). Dividing that expression by the coefficient $\frac{d_p}{u_*'^2 \rho}$, a dimensionless vector differential equation to

calculate the particle velocity \overline{u}_p under the effects of buoyancy, drag, Basset, virtual mass and fluid acceleration is established.

$$\begin{aligned} \frac{1}{\alpha} \frac{d\overline{u}_p}{dt} = & R \frac{d_p \overline{g}}{u^{*2}} - \frac{3}{4} C_D \left(\overline{u}_p - \overline{u}_f \right) \left| \overline{u}_p - \overline{u}_f \right| + \frac{9}{\tau_*^{0.25} \sqrt{R_p} \pi} \int_0^t \frac{d}{d\tau} \left(\overline{u}_f - \overline{u}_p \right) \frac{d\tau}{\sqrt{t^* - \tau}} \\ & + C_m \frac{d\overline{u}_f}{dt} + \frac{D\overline{u}_f}{Dt} \end{aligned} \quad (4-1)$$

A 3-D vector expression for the Magnus force presented by Crowe et al. (1998) and based on the original work of Rubinow and Keller (1961) is as follows:

$$\frac{3}{4} \left(\frac{1}{2} \overline{\omega}_f - \overline{\omega} \right) \times \left(\overline{u}_f - \overline{u}_p \right) \quad (4-2)$$

where $\overline{\omega}_f$ is the dimensionless fluid vorticity vector and \times indicates the vector cross product. For the lift force, Lee et al. (2006) extended the expression proposed by Wiberg and Smith (1985) to a 3-D scenario by projecting the resulting force in the particle coordinate system as follows:

$$\overline{F}_{lift} = \frac{3}{4} C_L \left(\left| \overline{u}_r \right|_T^2 - \left| \overline{u}_r \right|_B^2 \right) \left[\frac{w_p}{\left| \overline{u}_p \right|} \frac{u_p}{\sqrt{u_p^2 + v_p^2}}, \frac{w_p}{\left| \overline{u}_p \right|} \frac{v_p}{\sqrt{u_p^2 + v_p^2}}, \frac{\sqrt{u_p^2 + v_p^2}}{\left| \overline{u}_p \right|} \right] = \frac{3}{4} C_L \left(\left| \overline{u}_r \right|_T^2 - \left| \overline{u}_r \right|_B^2 \right) \overline{e} \quad (4-3)$$

where \overline{e} is a unit vector representing the direction of the lift force. Adding both Magnus and lift expressions to Eq. (4-1), the 3-D model describing the particle motion is presented as:

$$\frac{1}{\alpha} \frac{d\overline{u}_p}{dt^*} = R \frac{d_p \overline{g}}{u^{*2}} - \frac{3}{4} C_D \left(\overline{u}_p - \overline{u}_f \right) \left| \overline{u}_p - \overline{u}_f \right| + \frac{9}{\tau_*^{0.25} \sqrt{R_p} \pi} \int_0^t \frac{d}{d\tau} \left(\overline{u}_f - \overline{u}_p \right) \frac{d\tau}{\sqrt{t^* - \tau}} + C_m \frac{d\overline{u}_f}{dt}$$

$$+\frac{D\vec{u}_f}{Dt} + \frac{3}{4}\left(\frac{1}{2}\vec{\omega}_f - \vec{\omega}\right) \times (\vec{u}_f - \vec{u}_p) + \frac{3}{4}C_L\left(\left|\vec{u}_r\right|_T^2 - \left|\vec{u}_r\right|_B^2\right)\vec{e} \quad (4-4)$$

The 3-D model tracks the particle translational velocity given any flow field (Fig. 4-1). The particle rotation is calculated as presented in Section 4.2 below, to completely describe the particle motion in the space. These equations can be integrated in order to obtain the particle trajectory using the same procedure presented for the two dimensional model. The results obtained for the Basset term approximation is also applied in this case.

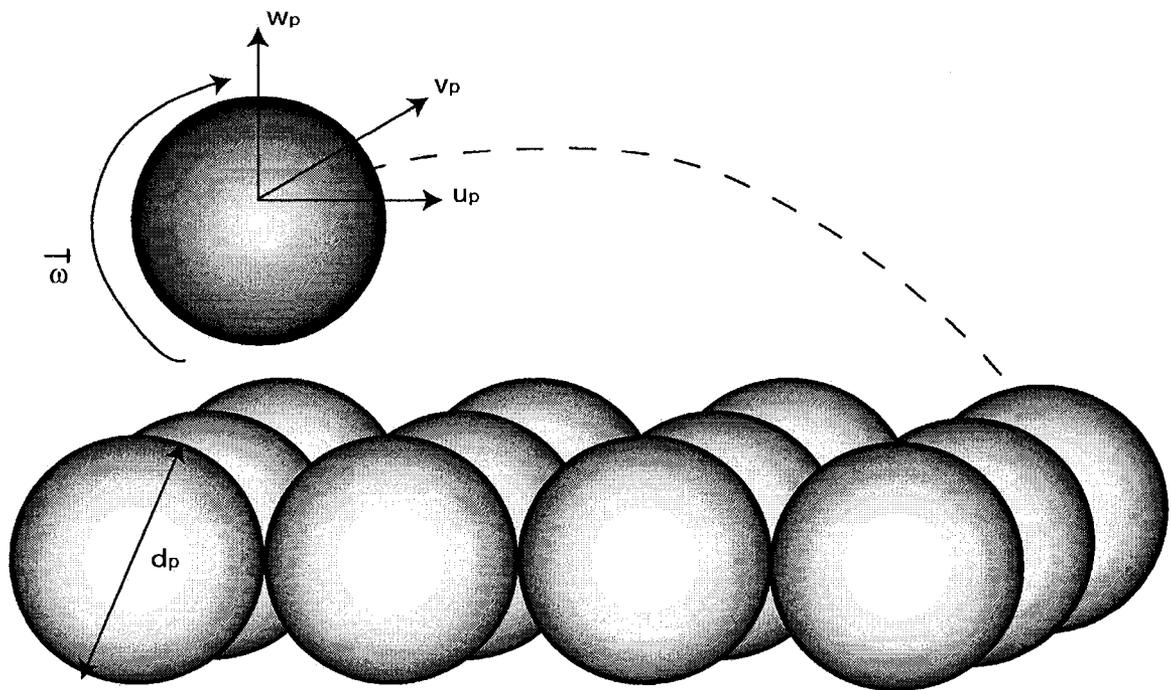


Figure 4-1: Definition sketch for saltating particle in 3-D.

In previous papers (Schmeecklee et al. 2003, Lee et al., 2006), similar equations to (4-4) were used by neglecting some terms and simplifying others via convenient hypothesis concerning the uniformity of the velocity profile in the transverse direction. No turbulence was considered in those papers.

In a turbulent velocity field, it is not possible to simplify any term from Eq. (4-4) and the complete expression must be used. For simplicity, in this chapter it will be assumed that the velocity field is provided by the law of the wall expression, to study the main characteristics of the particle motion. Therefore, the following expression describing the dimensionless span-wise component of the particle velocity (v_p) is added to Eqs. (3-9) and (3-10):

$$\frac{dv_p}{dt} = -\frac{3}{4}\alpha C_D (v_p) \left| \vec{u}_r \right| - \frac{9\alpha}{\sqrt{\pi} R_p \tau_*^{1/4}} \int_0^t \frac{d}{d\tau} v_p \frac{d\tau}{\sqrt{t-\tau}} \quad (4-5)$$

The following facts were considered to obtain Eq. (4-5): a) the components of the lift and virtual-mass forces in the transverse direction are identically zero because the velocity profile was assumed to be uniform in the transverse direction (see Chapter 3); b) the components of the Magnus and lift forces in the transverse direction were neglected on grounds that the particle relative rotation vector has small components other than the $\vec{\omega}_y$ (see Section 4.2). These facts were corroborated by simulating a particle jump under the same flow conditions with both the complete and the simplified expressions for both lift and Magnus forces. The complete expression is given by Eq. (4-6):

$$\begin{aligned} \frac{1}{\alpha} \frac{d\vec{u}_p}{dt^*} = & R \frac{d_p \vec{g}}{u^{*2}} - \frac{3}{4} C_D \left(\vec{u}_p - \vec{u}_f \right) \left| \vec{u}_p - \vec{u}_f \right| + \frac{9}{\tau_*^{0.25} \sqrt{R_p \pi}} \int_0^t \frac{d}{d\tau} \left(\vec{u}_f - \vec{u}_p \right) \frac{d\tau}{\sqrt{t^* - \tau}} + C_m \frac{d\vec{u}_f}{dt} \\ & + \frac{3}{4} \left(\frac{1}{2} \vec{\omega}_f - \vec{\omega} \right) \times \left(\vec{u}_f - \vec{u}_p \right) + \frac{3}{4} C_L \left(\left| \vec{u}_r \right|_T^2 - \left| \vec{u}_r \right|_B^2 \right) \vec{e} \end{aligned} \quad (4-6)$$

Tables 4-1 and 4-2 presents the results obtained for two particle sizes of interest and for 5 flow conditions each, showing that in the case of lift force the differences between both methods are negligible. In the case of Magnus force, the differences in terms

of dimensionless particle jump height and length are reasonable small (less than 3%) between the complete and simplified expressions.

Table 4-1: Comparison of results of 3-D particle-tracking model in terms of change in the particle jump characteristics when complete (Eq. 4-6) and simplified (Eq. 4-5) lift force expressions are used.

Δ represent s the percentage of change of the simplified expression relative to the complete expression.

		Complete Magnus Exp.		Simplified Magnus Exp.			
R_p	τ_*	H	L	H	L	ΔH (%)	ΔL (%)
73	1	0.179	2.224	0.175	2.175	2.4	2.2
	1.5	0.273	3.776	0.268	3.696	1.9	2.1
	2	0.374	5.690	0.369	5.630	1.3	1.0
	2.5	0.490	8.167	0.478	8.228	2.3	0.7
	3	0.602	10.776	0.592	11.031	1.6	2.4
250	1	0.209	2.644	0.209	2.612	0.3	1.2
	1.5	0.347	4.689	0.339	4.715	2.3	0.5
	2	0.486	7.361	0.474	7.367	2.5	0.1
	2.5	1.084	16.218	1.064	15.799	1.9	2.6
	3	1.315	21.296	1.278	20.898	2.8	1.9

Table 4-2: Comparison of results of 3-D particle-tracking model in terms of change in the particle jump characteristics when complete (Eq. 4-6) and simplified lift (Eq. 4-5) force expressions are used.

Δ represent s the percentage of change of the simplified expression relative to the complete expression.

		Complete Lift Exp.		Simplified Lift Exp.			
R_p	τ_*	H	L	H	L	ΔH (%)	ΔL (%)
73	1	0.320	4.136	0.319	4.131	0.23	0.13
	1.5	0.498	6.987	0.497	6.983	0.10	0.06
	2	0.675	10.282	0.675	10.277	0.01	0.04
	2.5	0.861	14.167	0.862	14.162	0.11	0.03
	3	1.026	17.898	1.028	17.895	0.19	0.02
250	1	0.335	3.954	0.335	3.949	0.02	0.14
	1.5	0.521	6.667	0.523	6.668	0.25	0.03
	2	0.691	9.461	0.693	9.466	0.42	0.05
	2.5	0.883	13.019	0.888	13.040	0.63	0.16
	3	1.070	16.714	1.078	16.756	0.74	0.25

4.2 Particle Rotation

Due to the combined effects of bed roughness and velocity gradients, particles rotate during the saltation process (Lee and Hsu, 1996). Hui and Hu (1991) conducted flume experiments and found the spinning rate to be about 40 revolutions/s in water. The particle spin can be decomposed depending on the axis where the particle rotates: screw-spin (if rotating along the stream-wise axis), top-spin (if rotating along the span-wise axis) and side-spin motions (if rotating along the wall-normal axis) respectively. Under uniform flow conditions, the saltating process are dominated by top-spin motions. Side-spin and screw-spin exists only at an early stage of the rising limb process (Lee and Hsu, 1996).

Several authors (Yeganeh et al., 2000; Yamamoto et al., 2001; Lukerchenko et al., 2006; Harada and Gotoh, 2006) have used the following expression to describe the rotation of a particle around an i axis.

$$I \frac{d\overline{\omega}^p}{dt} = -C_i \frac{\rho}{2} \left(\frac{d_p}{2} \right)^5 \left| \overline{\omega}_r^p \right| \overline{\omega}_r^p \quad (4-7)$$

where $\overline{\omega}^p$ is the particle rotation vector, $\overline{\omega}_r^p$ is the particle rotational velocity relative to the fluid vorticity and I is the particle's moment of inertia. Assuming that particles are spheres, I can be calculated as $2/5 m (d_p/2)^2$, where m is the particle mass. The right-hand side of the equation represents the viscous torque against the particle rotation, which was theoretically obtained by Dennis et al. (1980) and Takagi (1977). C_i is a non-dimensional coefficient which is a function of the Reynolds number for the rotational motion $Re_r = d_p^2 \left| \overline{\omega}_r^p \right| / 4\nu$, given by:

$$C_t = \frac{C_1}{\sqrt{Re_r}} + \frac{C_2}{Re_r} + C_3 Re_r \quad (4-8)$$

where the coefficients C_1 , C_2 and C_3 are presented in Table 4-3.

Table 4-3: Values of coefficient C1, C2 and C3. Computation of C_t coefficient.

Re_r	0-1	1-10	10-20	20-50	50+
C_1	0	0	5.32	6.44	6.45
C_2	50.27	50.27	37.2	32.2	32.1
C_3	0	0.0418	5.32	6.44	6.45

Using the particle diameter and the wall-friction velocity as the length and velocity scales, respectively, the non-dimensional particle rotation vector $\vec{\omega}$ as a function of the dimensionless particle rotational velocity relative to the fluid vorticity $\vec{\omega}_r$, is given by the following vector equation:

$$\frac{d\vec{\omega}}{dt} = \frac{-16C_t}{15\pi(R+1)} \left| \vec{\omega}_r \right| \vec{\omega}_r \quad (4-9)$$

In a 2-D case, only the top-spin component (ω_y) is calculated. from Eq. (4-9). It is interesting to note that Niño and García (1998a) employed an empirical equation to compute the top-spin component of the rotation vector.

The algorithm to calculate particle rotation was implemented computationally in the same 3-D code used to calculate particle translational velocities. Validation of the particle rotation sub-routines is discussed in the next chapter.

4.3 Algorithm for Particle-Wall Collisions

To describe the motion of a saltating particle, it is necessary to incorporate a collision algorithm to compute the particle velocity after the rebound event. A complete collision model usually can be divided in two sub-models: a) a series of equations describing the particle velocity after the rebound and b) a representation of the bed roughness.

4.3.1 Particle Rebound Sub-Models

Two widely-used sub-models for particle rebound were considered in this study. The first model considered is the García and Niño (1992). 2-D model which was based on the ideas of Tsujimoto and Nakagawa (1983), considering a saltating particle that approaches the bed at an angle θ_{in} and strikes the surface of the bed that faces upstream with an angle θ_b (see Section 3.1.2). The model has been used by Niño and García (1998a,b), Lee et al. (2000), and Lee et al. (2006) with very good agreement with experimental data for natural sediments saltating in a turbulent channel. This model has been extended to three dimensions by Lee et al. (2006). A limitation of this model is in that it does not describe the change on the particle rotation after the collision with the bed.

The second rebound sub-model, derived originally by Matsumoto (1970b) in 2-D and later extended to 3-D by Tsuji et al. (1985), considers the conservation of linear and angular momentum before and after the rebound; it also accounts for the particle rotation and the possibility of the particle sliding on the bed. Considering that the coefficients of restitution and friction are known, particle motions before and after the collision can be

estimated by solving three impulsive equations. The post-collision velocities (see Fig. 4-2), denoted with the superscript \wedge , as a function of the particle velocity immediately before the collision, denoted with the superscript \sim , are calculated depending on whether the particle slides or not on the bed. A particle slides if expression (4-10) below is satisfied:

$$\frac{\tilde{w}_p}{|\tilde{U}|_{in}} < \frac{-2}{7f(e+1)} \quad (4-10)$$

where $|\tilde{U}|_{in}$ is the modulus of the particle velocity vector before the collision with the wall. This criterion says that if the particle hits the bed at a shallow angle, the chances of sliding are high. If the criterion is met, i.e., if the particle slides, the post-collision velocities for the stream-wise, the span-wise and wall-normal directions are calculated as follows:

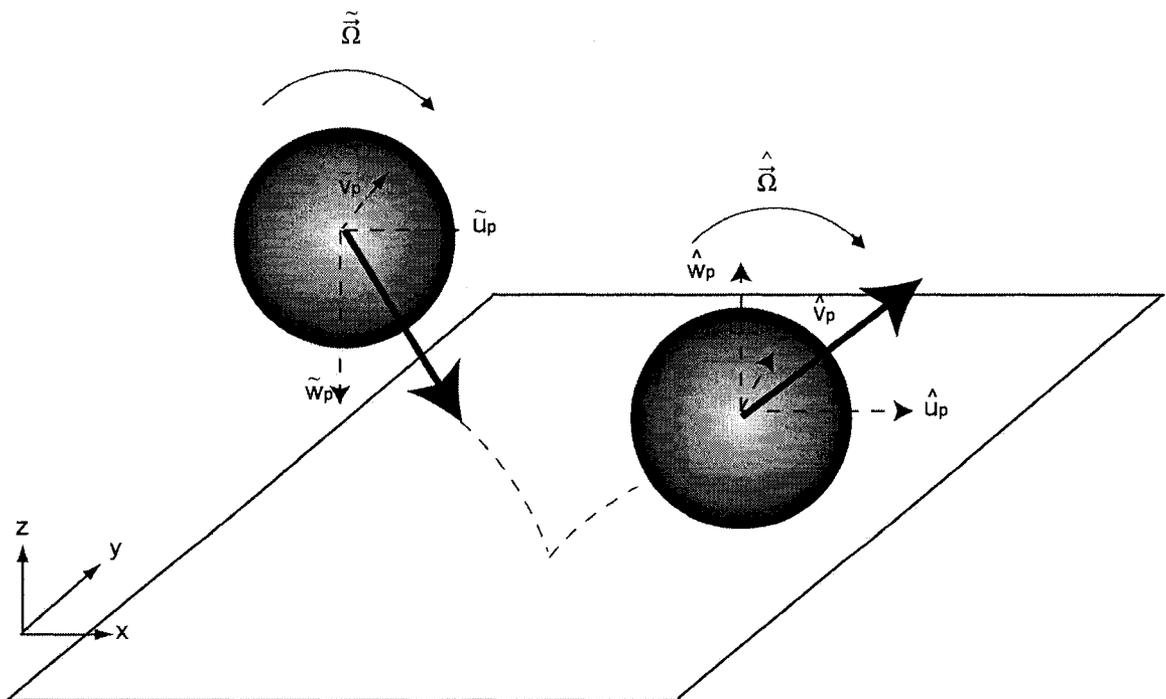


Figure 4-2: Parameter definition for a 3-D particle-wall collision. Tsuji et al. (1985)

$$\hat{u}_p = \left(\frac{5}{7}\right) \left(\tilde{u}_p - \frac{2a}{5} \tilde{\omega}_y \right); \quad \hat{v}_p = \left(\frac{5}{7}\right) \left(\tilde{v}_p + \frac{2a}{5} \tilde{\omega}_x \right); \quad \hat{w}_p = -e \tilde{w}_p \quad (4-11)$$

$$\hat{\omega}_x = \frac{2 \tilde{v}_p}{d_p}; \quad \hat{\omega}_y = -\frac{2 \tilde{u}_p}{d_p}; \quad \hat{\omega}_z = \tilde{\omega}_z \quad (4-12)$$

where ω_x , and ω_z indicate the components of the particle rotation vector along the stream-wise, and wall-normal directions, respectively, before the collision. If the particle does not slide, i.e., if the criterion in (4-10) is not met, the post-collision velocities are calculated as:

$$\hat{u}_p = \tilde{u}_p + \varepsilon_x f(e+1) \tilde{w}_p \quad \hat{v}_p = \tilde{v}_p + \varepsilon_y f(e+1) \tilde{w}_p \quad \hat{w}_p = -e w_p \quad (4-13)$$

$$\hat{\omega}_x = \tilde{\omega}_x - \frac{5}{d_p} \varepsilon_y f(e+1) \tilde{w}_p \quad \hat{\omega}_y = \tilde{\omega}_y + \frac{5}{d_p} \varepsilon_x f(e+1) \tilde{w}_p \quad \hat{\omega}_z = \tilde{\omega}_z \quad (4-14)$$

where the coefficients ε_x and ε_y are defined as $\varepsilon_x = (\tilde{u}_p + a \tilde{\omega}_y) / |\tilde{U}_{in}|$ and $\varepsilon_y = (\tilde{v}_p - a \tilde{\omega}_x) / |\tilde{U}_{in}|$. They satisfy the relation $\varepsilon_x^2 + \varepsilon_y^2 = 1$. Eqs. (4-11) to (4-14) are developed for a horizontal contact plane between the flying particle and the bed and their derivation is fully described in Crowe et al. (1998). The beauty of this model is that it can be easily extended to describe the inter-particle collision, feature that was also added to the present 3-D model for multiple moving particles (see Section 4.6).

Tsuji et al.'s model has been extensively used to describe the transport of particles in pneumatic conveying systems (Tsuji et al. 1987; Sommerfeld 1992; Sommerfeld and Huber 1999; Kartushinsky and Michaelides 2004) but only once, to the best of our

knowledge, to address the motion of natural sediment particles in a turbulent channel (Lukerchenko et al. 2006).

4.3.2 Previous model for the Treatment of the Surface: a Roughness (Bed-Representation) Model.

To avoid the attenuation of the vertical velocity after tens of rebounds, the irregularity of the collision process must be considered (Gordon et al. 1972, Crowe et al. 1998). Several authors have incorporated the effect of the bed roughness to the collision model through an angle between the channel surface and the real point of contact between the flying particle and the bed. In those models the wall has been replaced by a virtual wall where its inclination has been assumed to be either *uniformly distributed* in the range $(-4^\circ, +4^\circ)$ (Tsuji et al. 1987), or *normally distributed* between $(-4, 4^\circ)$ (Sommerfeld 1992).

García and Niño (1992) in turn assumed that the bed is formed by uniformly packed spheres as shown in the previous chapter.

4.3.3 A New Bed-Representation Sub-Model

This study introduces a new 3-D algorithm to represent wall roughness, extending the García and Niño (1992) treatment of the bed. The roughness pattern was defined as an array of spheres of diameter d_p placed one next to the other, as Niño and García did (See Fig. 3-1). The point of contact between the flying particle and the bed defines an inclination plane, which is now defined by using two angles, θ_b (Fig. 3-3) and α_b (Fig. 4-3). The values of θ_b and α_b can only vary between -30° and 30° (Niño and

García 1992), given the configuration of the particles lying in the bed. An auxiliary variable r_2 is defined as shown in Fig. 4-3. Analogously to the 2-D case, a geometric relation between α_b , α_{in} and r_2 is given by:

$$\frac{r_2}{d_p} = \frac{1}{2} (\cos(\alpha_b) - \tan(\alpha_{in}) \sin(\alpha_b)) \quad (4-15)$$

With this configuration, we defined three sub-models. Uniformly Distributed Angles (*UDA*) sub-model assumes that both angles θ_b and α_b are uniformly distributed between -30° and 30° and these angles are determined through random number generators. Independent Bed Angle (*IBA*) and Dependent Bed Angle (*DBA*) sub-models are based on relations using Eq. (4-15) which define θ_b with r_1 and θ_{in} and α_b with r_2 and θ_{in} .

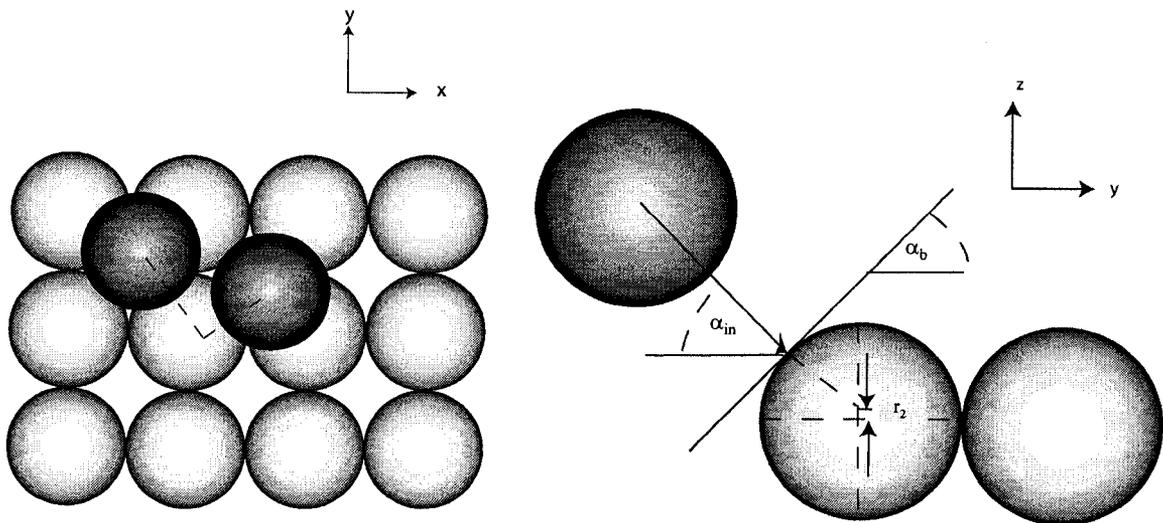


Figure 4-3: Three dimensional collision of a particle with the bed. The bed is composed by uniformly packed spheres placed one next to the other. The particle diameter d_p is equal for the moving and the resting spheres. Side view. Definition of α_{in} and α_b .

In the *IBA* sub-model the values of r_1/d_p and r_2/d_p are obtained as a random number uniformly distributed between 0 and 0.5, independently of the incidence angles

θ_{in} and α_{in} , respectively. In the *DBA* sub-model, the values of r_1/d_p and r_2/d_p are obtained by reducing this range depending on the particle conditions just before the collision, defined as r_{min1}/d_p and r_{max1}/d_p , and r_{min2}/d_p and r_{max2}/d_p , respectively. These ranges are defined as follows: the minimum values (r_{min1}/d_p and r_{min2}/d_p) are equal to 0 and the maximum values (r_{max1}/d_p and r_{max2}/d_p) are obtained by replacing the incidence angle (θ_{in} and α_{in} , respectively) in Eq. (4-15) and assuming that the bed contact angle (θ_b and α_b , respectively) is equal to 30° . Then, the values of r_1/d_p and r_2/d_p are obtained with Eq. (4-15) and random number generators between r_{min1}/d_p and r_{max1}/d_p , and r_{min2}/d_p and r_{max2}/d_p , respectively.

4.4 Assessments of Particle-Wall Collision Models

Six different cases were investigated. The results of these simulations were compared with two experimental data sets for validation of the large-scale characteristics of the model: the observations by Niño and García (1998b) for natural sediment particles saltating along a horizontal channel, and the experiments by Lee and Hsu (1994). Four sets of values for the friction and restitution coefficients were tested, as presented in the literature review and summarized in Table 4-4. Set Schmeckle represents the values provided by Schmeckle et al. (2001); Set Shen includes the values proposed by Shen et al. (1989); Set Niño considers the Niño and García's (1994) values, and Set Tsuji includes the values proposed by Tsuji et al. (1987).

Table 4-4: Set of values of friction and restitution coefficients used in the particle-wall collision model comparison.

Set	Author	Restitution coefficient e	Friction coefficient f
Schmeecklee	Schmeeckle et al. (2001)	0.65	0.1
Shen	Shen et al. (1989)	0.95	0.3
Niño	Niño and García (1994)	$e = 0.75 - 0.25 \tau_* / \tau_{*c}$	0.89
Tsuji	Tsuji et al. (1987)	0.8	0.4

4.4.1 Characteristics of the Numerical Tests

All four sets were tested for the six simulation runs defined in Table 4-5.

The numerical models were run for a simulation time long enough to have meaningful statistics. In order to remove the effect of the initial conditions, the first jumps were not considered in the statistical analysis.

Table 4-5: Simulation summary.

Run #	Flight model (Sections 3.2 and 4.1)	Rebound model (Section 3.2.2 and 4.3.1)	Roughness model (Sections 3.2.2, 4.3.2 and 4.3.3)
1	2D Niño and García (1994) (Section 3.2)	2D García and Niño (1992) (Section 3.2.2)	García and Niño (1992) (Section 3.2.2)
2	3D (Section 4.1)	3D Tsuji et al. (1985) (Section 4.3.1)	Tsuji et al. (1987) (Section 4.3.2)
3	3D (Section 4.1)	3D Tsuji et al. (1985) (Section 4.3.1)	(Sommerfeld 1992).
4	3D (Section 4.1)	3D Tsuji et al. (1985) (Section 4.3.1)	<i>UDA</i> sub-model (Section 4.3.3)
5	3D (Section 4.1)	3D Tsuji et al. (1985) (Section 4.3.1)	<i>IBA</i> sub-model B (Section 4.3.3)
6	3D (Section 4.1)	3D Tsuji et al. (1985) (Section 4.3.1)	<i>DBA</i> sub-model (Section 4.3.3)

4.4.2 Simulation Results of the Numerical Tests

Appendix C includes tables with the results of the runs for $R_p = 73$. Figure 4-4 present the results for Run 6, Set Tsuji, the one with the best agreement with the data.

In an attempt to provide an accessible means for interpretation, Table 4-6 below presents the results of the transformation of the statistical data into an overlap area index. An area for each of the three parameters is defined by creating regions bounded by using the average value of each parameter plus/less standard deviation. These areas are defined for both the measured data and the numerically obtained results. The overlap area is determined by calculating the percentage of the area of overlap with respect to the area of the experimental results. The overlap area index represents the average overlap area determined for each of the three parameters. Therefore, the bigger the overlap area index, the better the performance of the collision model (See Fig. 4-5 as example).

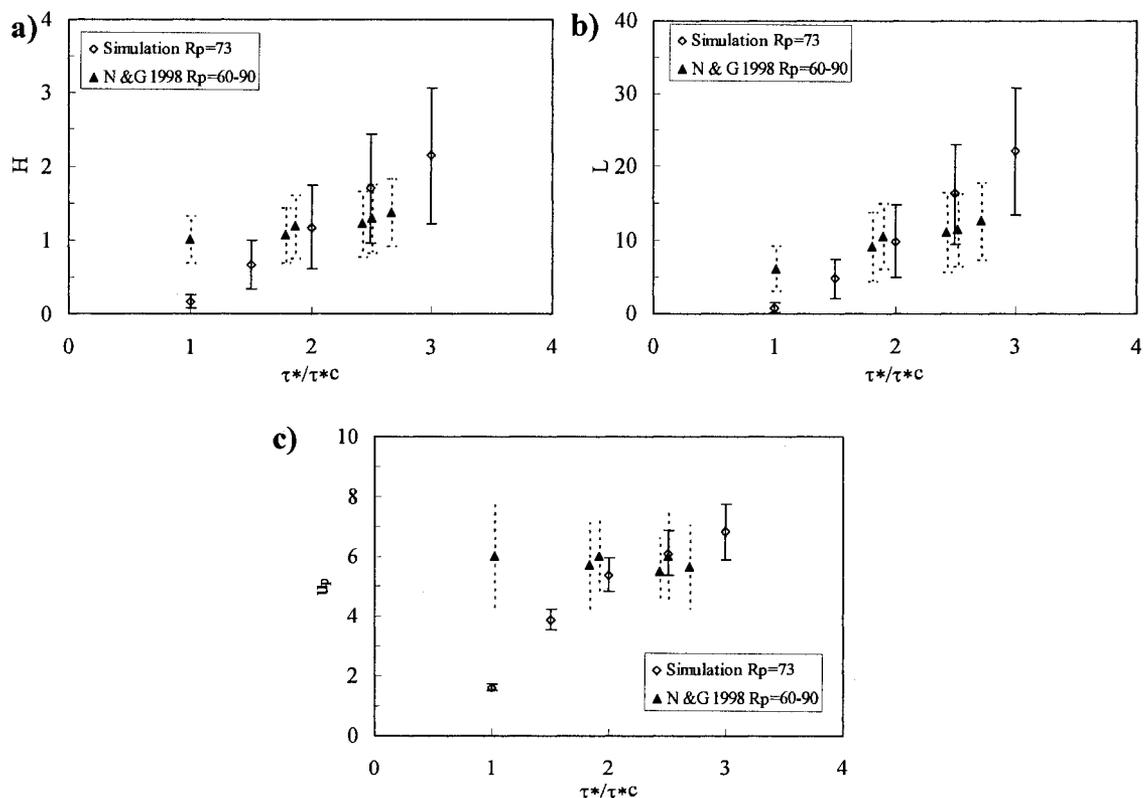


Figure 4-4: Comparison of simulations with experimental data for the case of a particle moving in a flume and rebounding with the wall (Run 6, Set Niño). The figure shows results associated with: a) the particle jump height (H), b) the particle jump length (L) and c) the particle stream-wise mean velocity. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. $R_p=73$.

As depicted in Fig. 4-5 and presented in Table 4-6, the results obtained for Run 6 Set Niño presents a 42.41% of overlap area with respect to the experimental results of particle jump height by Niño and García (1998b).

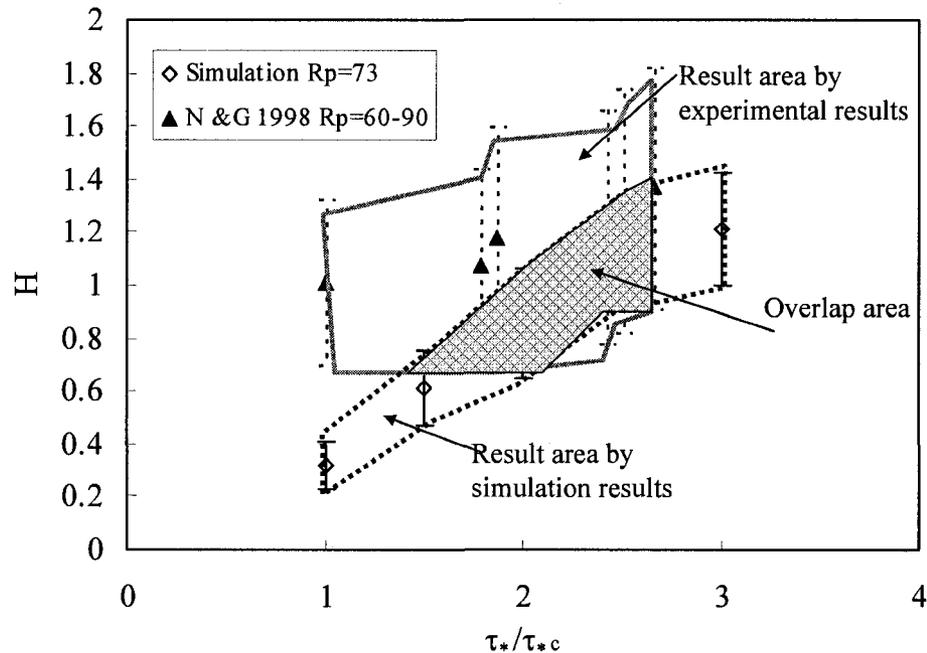


Figure 4-5: Overlap Area Determination: $R_p = 73$, Simulation Run 6. Set Niño. The overlap area is defined is depicted as the grey area in the figure.

Appendix D shows the results obtained for $R_p = 250$ using the different collision models. An example of simulation results for Run 5 and $R_p = 250$, Set Shen is presented in Fig. 4-6. Unfortunately, the experimental data by Lee et al. (1994) provide no information of the dispersion of them. Therefore, the overlap area index used for $R_p = 73$ can not be used in this case. A new statistical index the results of the root mean square error (RMSE) in introduces. Table 4-7 presents the results RMSE. The experimental results obtained by Lee and Hsu (1994) were considered as the observed value. If there is no experimental data available for the corresponding value of τ_* , an equivalent

experimental value is obtained by using a linear fit approximation of the existing experimental values. The error index for each simulation/set is defined as the average between the values of RMSE observed for the particle jump height, length and mean particle velocity.

Table 4-6: Statistical analysis using the overlap area indices. $R_p = 73$.

Run		Schmeeckle	Shen	Niño	Tsuji
1	Overlap Area H (%)	13.37	37.18	39.80	29.54
	Overlap Area L (%)	7.82	44.13	49.74	42.76
	Overlap Area U (%)	22.37	78.63	72.40	73.85
	Overlap Index	14.52	53.31	53.98	48.72
2	Overlap Area H (%)	0.00	0.00	0.00	0.00
	Overlap Area L (%)	0.00	0.00	0.00	0.00
	Overlap Area U (%)	12.86	13.96	0.00	4.57
	Overlap Index	4.29	4.65	0.00	1.52
3	Overlap Area H (%)	0.00	0.00	0.00	0.00
	Overlap Area L (%)	0.00	0.00	0.00	0.00
	Overlap Area U (%)	0.00	0.00	0.00	0.00
	Overlap Index	0.00	0.00	0.00	0.00
4	Overlap Area H (%)	0.00	23.45	12.91	66.17
	Overlap Area L (%)	1.70	43.79	48.67	76.55
	Overlap Area U (%)	0.00	0.00	58.37	52.87
	Overlap Index	0.57	22.41	39.98	65.20
5	Overlap Area H (%)	40.63	67.63	41.63	71.21
	Overlap Area L (%)	51.69	57.03	51.30	56.86
	Overlap Area U (%)	65.29	76.11	59.16	83.33
	Overlap Index	52.54	66.92	50.70	70.47
6	Overlap Area H (%)	55.32	61.62	45.56	75.75
	Overlap Area L (%)	45.33	57.50	55.33	56.96
	Overlap Area U (%)	65.31	87.63	51.35	82.49
	Overlap Index	55.32	68.92	50.75	71.74

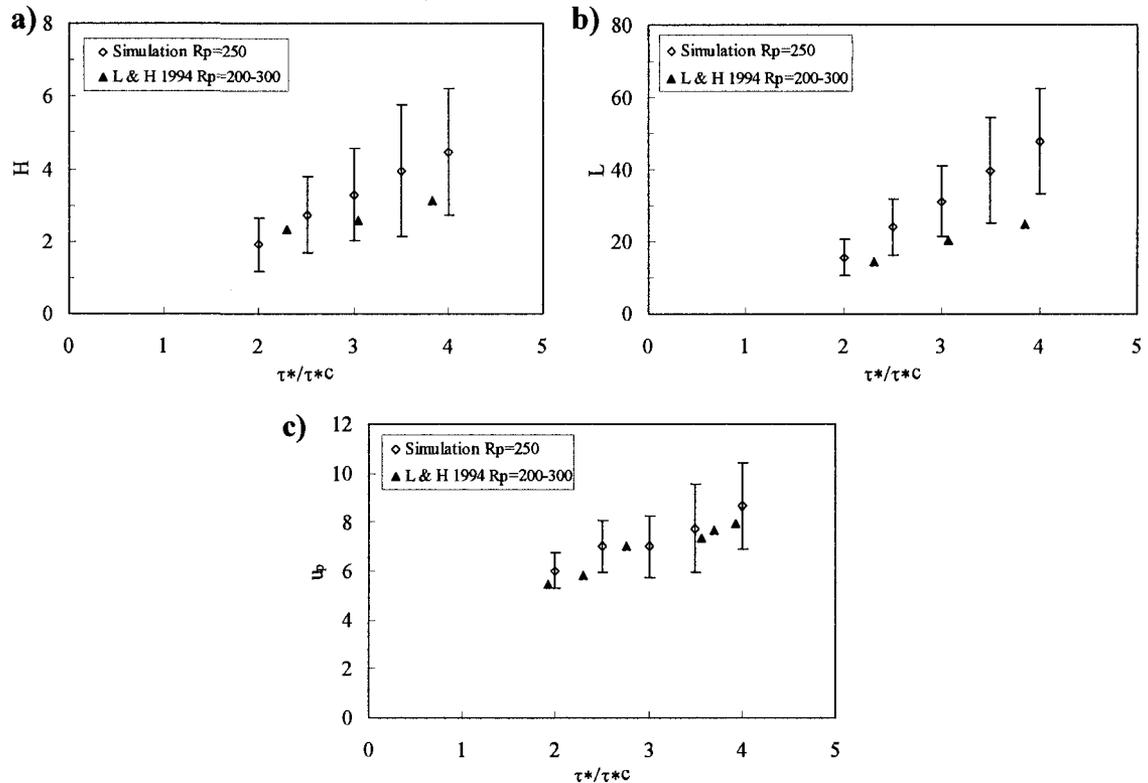


Figure 4-6: Comparison of simulations with experimental data for the case of a particle moving in a flume and rebounding with the wall (Run 6, Set Niño). The figure shows results associated with: a) the particle jump height (H), b) the particle jump length (L) and c) the particle stream-wise mean velocity. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. $R_p=250$.

For the García and Niño (1992) particle-wall collision and roughness models considered in Run 1 and $R_p=73$ the best agreement with the experimental results was obtained using the Set Niño of friction and restitution coefficients (which are the proposed by Niño and García (1994) for this particle size). On average, the results obtained for the model have a very good agreement with the average experimental value. However they present a slightly smaller standard deviation than the experimental data. In spite of the good agreement of this model, we are more inclined to use the post-collision model by Tsuji et al. (1985) due to its capability to be extended for inter-particle collision, feature that would be addressed in a future work.

Table 4-7: Statistical analysis using the root mean squared error (RMSE). $R_p=250$.

Run		Schmeeckle	Shen	Niño	Tsuji
1	RMSE H (%)	21.93	5.31	3.55	7.51
	RMSE L (%)	1.30	5.28	4.40	3.82
	RMSE U (%)	2.44	4.18	4.41	3.29
	Error Index	8.56	4.92	4.12	4.88
2	RMSE H (%)	37.66	37.31	37.78	37.61
	RMSE L (%)	5.00	4.82	5.10	4.99
	RMSE U (%)	1.46	2.78	6.00	4.60
	Error Index	14.71	14.97	16.29	15.73
3	RMSE H (%)	37.68	37.09	37.79	37.58
	RMSE L (%)	5.03	4.75	5.11	4.98
	RMSE U (%)	0.45	1.17	4.99	3.05
	Error Index	14.38	14.34	15.96	15.20
4	RMSE H (%)	25.26	21.96	19.84	18.39
	RMSE L (%)	1.74	6.99	2.03	5.83
	RMSE U (%)	5.78	11.29	7.76	10.06
	Error Index	10.93	13.41	9.88	11.43
5	RMSE H (%)	26.65	78.99	44.23	67.37
	RMSE L (%)	8.90	17.18	12.12	15.46
	RMSE U (%)	6.25	10.22	8.19	9.57
	Error Index	13.93	35.46	21.51	30.80
6	RMSE H (%)	32.83	59.97	26.05	47.80
	RMSE L (%)	9.86	13.66	8.88	12.04
	RMSE U (%)	6.00	7.72	4.50	7.20
	Error Index	16.23	27.11	13.14	22.35

As said, Tsuji et al.'s (1985) model in Run 2 assumes a normal probability distribution of the inclination angle of the bed between -4° and $+4^\circ$ and the roughness pattern as defined by Tsuji et al. (1985, 1987), simulation results in Table 4-6 indicate that Run 2 does not represent the experimental results. (The overlap areas for jump height and length—for all restitution and friction coefficients—are equal to zero). The same agreement with data if the narrow range is also observed for Run 3 simulations (Sommerfeld (1992) roughness pattern).

The new roughness model presented in Run 4 for $R_p=73$, which basically increases the range of the inclination angles with respect to the range proposed by Tsuji et al. (1987) and Sommerfeld (1992) generally increases the overlap index. The simulation results present a very good agreement with the experimental data with an overlap area index ranging from 22% (Set Shen) to 65% (Set Tsuji). However a closer look at the results shows that the roughness model used in Run 4 produces both very large or extremely small jumps, producing a jump distribution extremely non-uniform with a very large standard deviation (See Appendix C). Analyzing this model in detail, the assumption that the contact plane is defined with angles varying between 30° and -30° is unrealistic. Positive values of θ_b indicate that the particle hits the upstream face of the lying particle, and negative values of θ_b indicate a collision occurring at the downstream face of the bed particle. Considering the characteristic of the particle velocity just before the collision (positive stream-wise velocity and negative wall-normal velocity), there is a bigger probability of hitting the upstream than the downstream side of the roughness (see Sommerfeld and Huber 1999). Therefore, by assuming a uniform probability for θ_b (as the *UDA* sub-model does) we are providing the model with particle collision angles with no real physical meaning. This causes some of the jumps to have values of length and height almost equal to zero, which is unrealistic.

Simulation results for Runs 5 and 6 produce the strongest agreement with data for $R_p=73$. The results of Table 4-6 suggest that Run 5 improved the prediction made by the *IBA* sub-model, due to the fact that the definition of r_1/d_p and r_2/d_p reduce the range of variability of the angle θ_b . However, given some incidence angle θ_{in} , there are

some values of r_1/d_p between 0 and 0.5 that are unfeasible. Run 6 considers this geometrical constraint and the results obtained produce the best agreement with the experimental data available, by using the Set Tsuji of friction and restitution coefficient with an average overlap area of 72%. Therefore, the proposed roughness method (DBA sub-model), which assumes uniform values of r_1/d_p and r_2/d_p between a maximum and a minimum value is able to reproduce with a high degree of accuracy the experimental data provided by Niño and García (1998b).

For $R_p=250$, the numerical simulations were compared with the experimental data provided by Lee and Hsu (1994). In this case, the simulation results with the RMSE index are obtained with those Run 1 model, using the restitution and friction coefficient given by Niño and García (Set Niño). However, the simulation results for this model and the experimental data available presents two different trends, as it was discussed in Section 3.3.

This unrealistic trend shown by the numerical simulation using the two dimensional model by Niño and García (1994) is corrected by explicitly estimating the particle rotation at every time step using the Eq. (4-8), as done in Run 2 to Run 6.

Using the RMSE index as the parameter to select the best collision model for $R_p=250$, the results obtained using Run 4 Set Niño provided the best results. However a closer look at the results shows that the roughness model used in Run 4 produces both very large or extremely small jumps, producing a jump distribution extremely non-uniform with a very large standard deviation (See Appendix D). Therefore, these results

are dismissed. Therefore, using the RMSE, the best results for the collision model for $R_p=250$, are obtained by using Run 6 Set Niño.

Thus, considering the performance of the six models analyzed, for both sets of experimental data the simulation results obtained for Run 6 produce the best fit. Thus roughness *DBA* sub-model, provides a good representation of bed load particle motion in the range of particles corresponding to sand. The values of the restitution and friction coefficients depend on the particle size simulated: for the smaller size, values of 0.8 and 0.4 (Set Tsuji) respectively are selected, and for the larger size, values of 0.5 and 0.89 (Set Niño) respectively more accurately represent the particle motion close to the bed. If the friction and restitution coefficients represent the amount of energy conserved after the collision (as discussed in Section 3.5) and considering that the energy is proportional to the mass, it is reasonable to think that the larger the particle the larger the amount of energy lost in the collision, and thus, the smaller the value of both coefficients.

4.4.3 Validation of the Selected Collision Model

The angular velocity of the particle resulting from the model with the best performance (Run 6, Set Tsuji) is compared with the experimental results obtained by Niño and García (1998a). The experimental setup by Lee and Hsu (1996) obtained valuable information of particle angular velocity for larger particles ($R_p=2126$) and it was also simulated. Both simulations present a very good agreement with the experimental measurements, as depicted in Fig. 4-7.

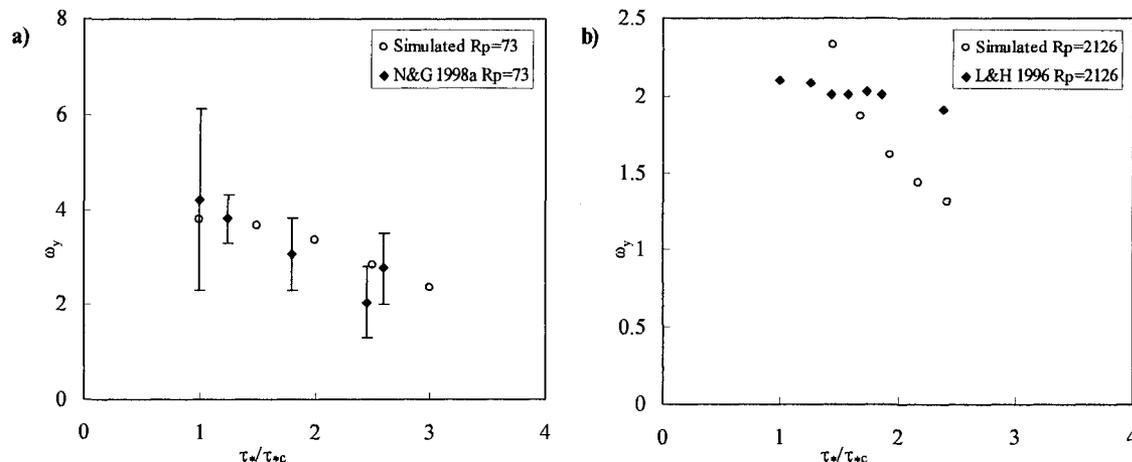


Figure 4-7: Particle angular velocity simulations. a) Niño and García (1998a) for $R_p=73$. b) Lee and Hsu (1996) for $R_p=2126$.

To validate the results of the selected collision model, the writer studied the characteristics of the collisions (small-scale level of validation). Niño and García (1998a) provided experimental information of the take-off angles of the particles of $R_p=60$ to 90. In Fig. 4-8 the experimental data obtained from this study is compared with the simulated results for the collision model with the best performance (Run 6, Set Tsuji), with an acceptable agreement.

The model corresponding to Run 6, Set Tsuji, was used to analyze the particle lateral dispersion defined as the absolute value of the angle of deviation of the particle in the x-y plane. A comparison with observations undertaken by Niño and García 1998a was made (See Fig. 4-9). In general terms, good agreement was found between the numerical simulation and the experimental results. The bed roughness model selected produced slightly bigger lateral dispersion angles than the values found experimentally. The results obtained numerically also had a larger deviation from the mean value.

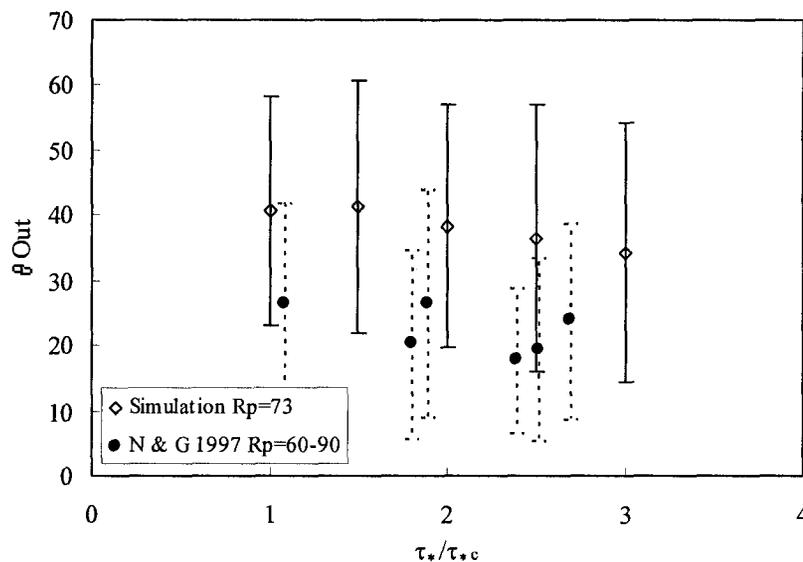


Figure 4-8: Take off angle comparison. Simulation and experimental results. $R_p=73$. Run 6 Set Tsuji.

The average value of the lateral deviation angle (α_d) resulting from the numerical simulation shows dependency with respect to the flow conditions, contrary to experimental results presented by Niño and García (1998a). The authors stated that the lateral deviation of particle trajectories may be produced by two different mechanisms: the first associated with the initial conditions given by the rebound with the wall and the second associated with cross-flow turbulent events. Both processes occur simultaneously and cancel each other as flow velocity increases, resulting in a constant lateral deviation. In our simulation, as the flow velocity increases, the particle stream-wise velocity increases (See Fig. 4-4c and 4-6c) but the span-wise component remains relatively constant, decreasing the average lateral deviation angle. There is no other effect (like the presence of secondary currents or turbulence effect in this section) that may counteract the shear flow effect on the decrease of the deviation angle.

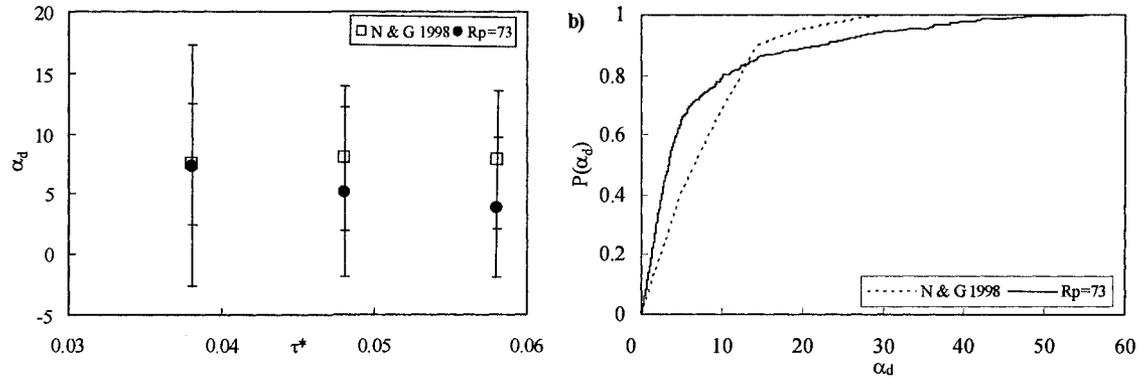


Figure 4-9: Comparison of the absolute value of the deviation angle $|\alpha_d|$ obtained in the numerical simulation and experimentally by Nifio and García (1998a). a) Absolute value of the deviation angle α_d of particle trajectories as a function of the flow conditions. b) Cumulative probability distribution of the absolute value of the deviation angle.

4.5 Analysis of Particle Velocity Under a Non-Turbulent Velocity Field.

Using the selected rebound sub-model (Tsuji et al. 1985) and bed roughness sub-model (*DBA* sub-model) Fig. 4-10 shows the temporal evolution of the particle velocity in the stream-wise, span-wise and wall-normal components in a non-turbulent velocity field.

A discontinuity pattern is observed in each of the parameters plotted in Fig. 4-10. This characteristic is given by the collision of the particle with the wall, which produces a sudden change in the particle trajectory. To understand the change in the velocity magnitude over the time it is necessary to understand what how the particle interacts with the flow field, and therefore what is the particle position with respect to the wall along the time.

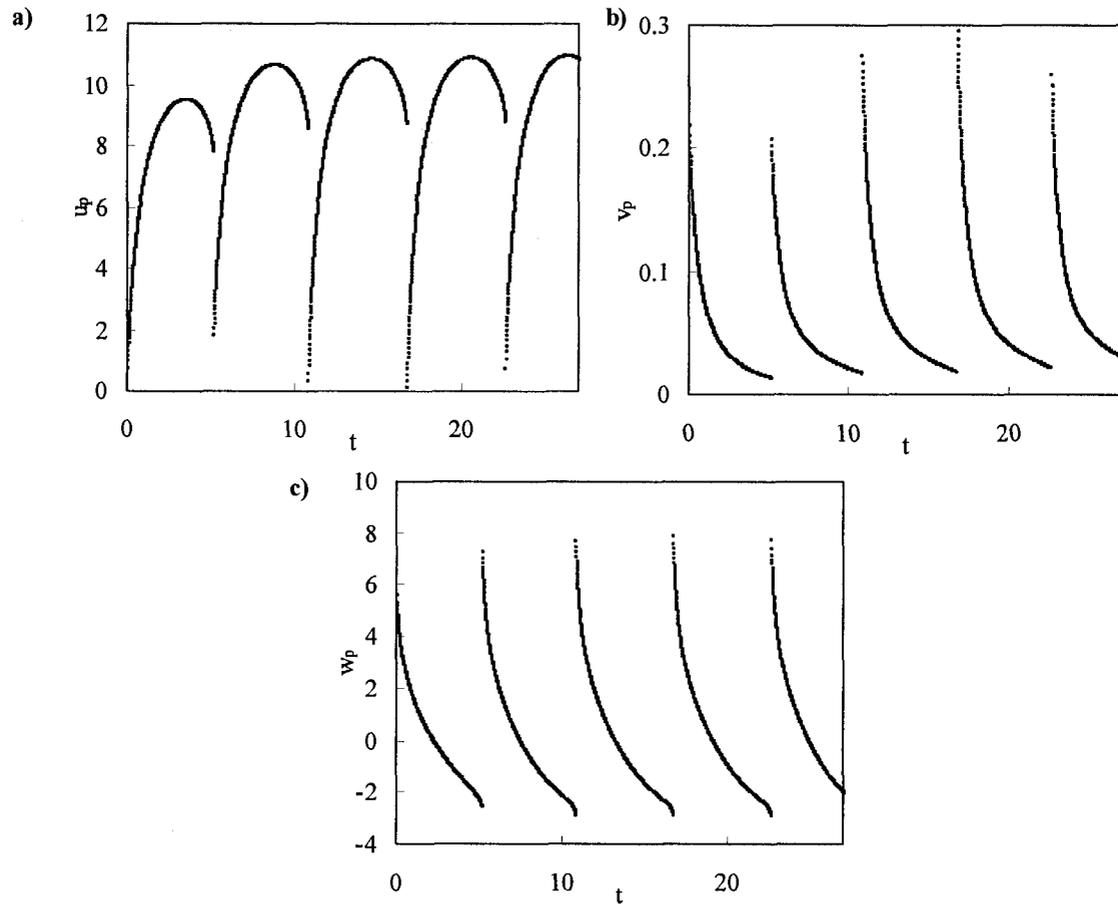


Figure 4-10: Single particle velocity time series in a velocity field defined by a semi-logarithmic law. $R_p = 73$, $\tau_* / \tau_{*c} = 2.0$. a) stream-wise component. b) span-wise component. c) wall-normal component.

Fig. 4-11 presents a zoom view of the data presented in Fig. 4-10 for one specific jump and the particle position with respect to the wall in the same time. As it is expected, the wall-normal component of the velocity presents a maximum value just after the particle collides with the wall, and it decreases due to the action of the gravity until the particle hits the wall again. The particle velocity in the wall normal component is equal to zero when the particle reaches its maximum altitude.

The span-wise velocity component decreases after each jump due to the friction effects given by the drag and Basset forces. The magnitude of this velocity is relatively small compared with the other two velocity components.

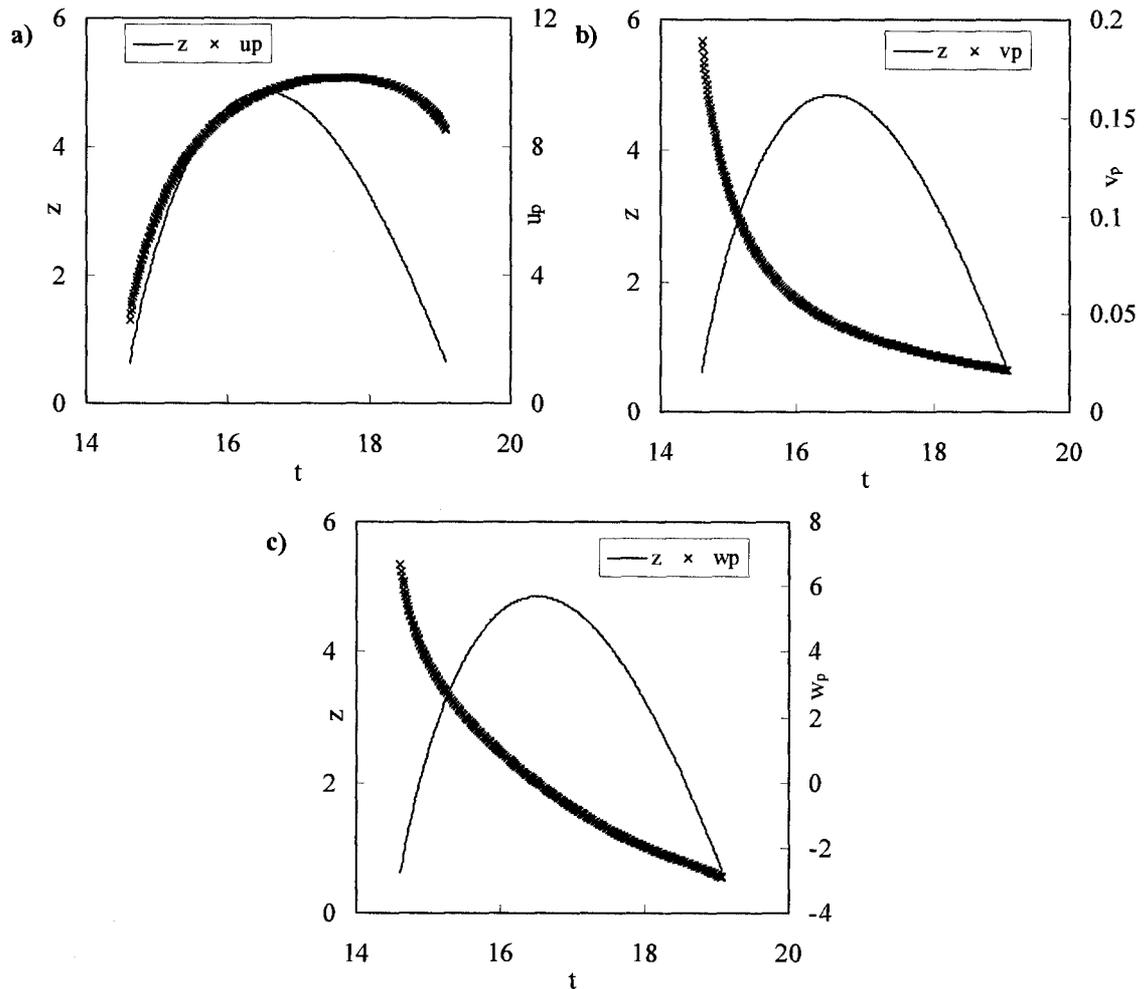


Figure 4-11: Velocity variation over the time for one particle $R_p = 73$, $\tau_p / \tau_c = 2$ with the corresponding particle elevation. a) Stream-wise component. b) Span-wise component. c) Wall-normal component.

The magnitude of the particle velocity at the stream-wise component is the largest of the three components, due to advection by the flow. After colliding with the wall, the particle gains enough momentum reach higher elevations where fluid velocity is higher.

Therefore, the magnitude of the stream-wise velocity increases until it reaches a maximum value, and then it decreases before the particle encounters the wall again. The maximum stream-wise velocity is not reached at the particle highest elevation point with respect to the wall, but slightly after. In spite of the fluid velocity decreasing below the maximum particle height, the inertia of the particle plays a fundamental role in maintaining the particle acceleration and allowing the particle velocity to increase for a longer period of time.

Fig. 4-12 shows the results obtained for a particle angular velocity along each axis over the time under the velocity field provided by the law of the wall. Two interesting features are observed in the plot: first, the decaying nature of the modulus of the angular velocity, due to the presence of the rotation drag force that decreases the particle acceleration every time step. Second, several peaks are observed in the plot, which correspond to the time step where the particle hits the wall (similar to the particle translation velocity). At that moment, the wall provides new angular momentum to the particle. It is also observed that the magnitude of the angular velocity along the span-wise axis (ω_y) is significantly larger than the other two components of the angular velocity, as it was observed by Lee and Hsu (1996).

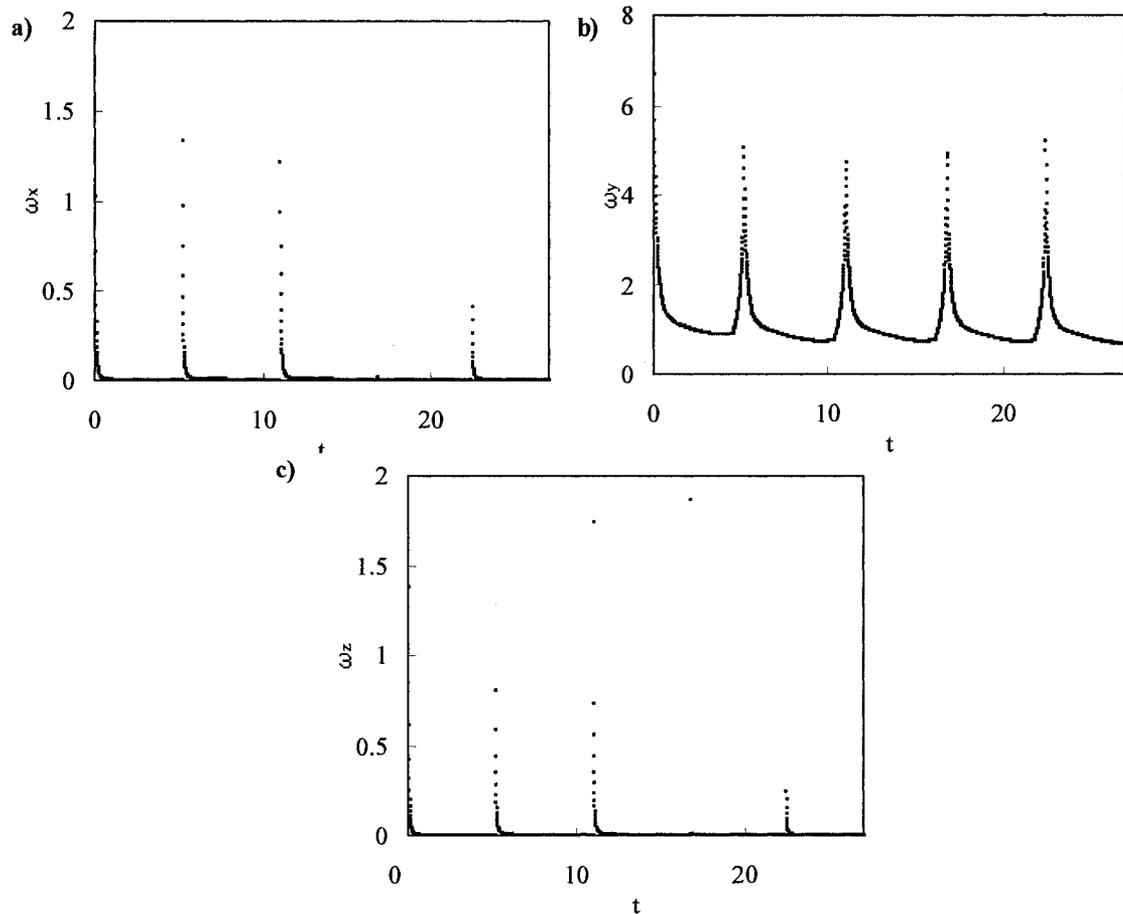


Figure 4-12: Dimensionless particle rotation along the different axis. $R_p = 73$, $\tau_* / \tau_{*c} = 2$. a) Stream-wise component. b) Span-wise component. c) Wall-normal component.

Considering all the conclusions obtained in this chapter, the three dimensional particle tracking model has been validated and represents a realistic approximation of the linear and angular velocities of particles close to a fixed wall.

4.6 Motion of Multiple Particles

A more realistic scenario to simulate the bed load transport requires the simulation of multiple particles moving at the same time, colliding with the wall and among them. The numerical code developed was modified to represent this condition. A

simple simulation was run to evaluate its performance. Fig. 4-13 shows the trajectory of 4 particles of $R_p=73$ and under flow conditions given by $\tau_* / \tau_{*c}=1.5$, without considering particle-particle collision. At first glance the trajectory of each particle seems realistic, compared with the trajectory observed for natural sediments along a channel. The dispersion of the particles in the span-wise component is small and is basically given by the rebound of the particle at the wall.

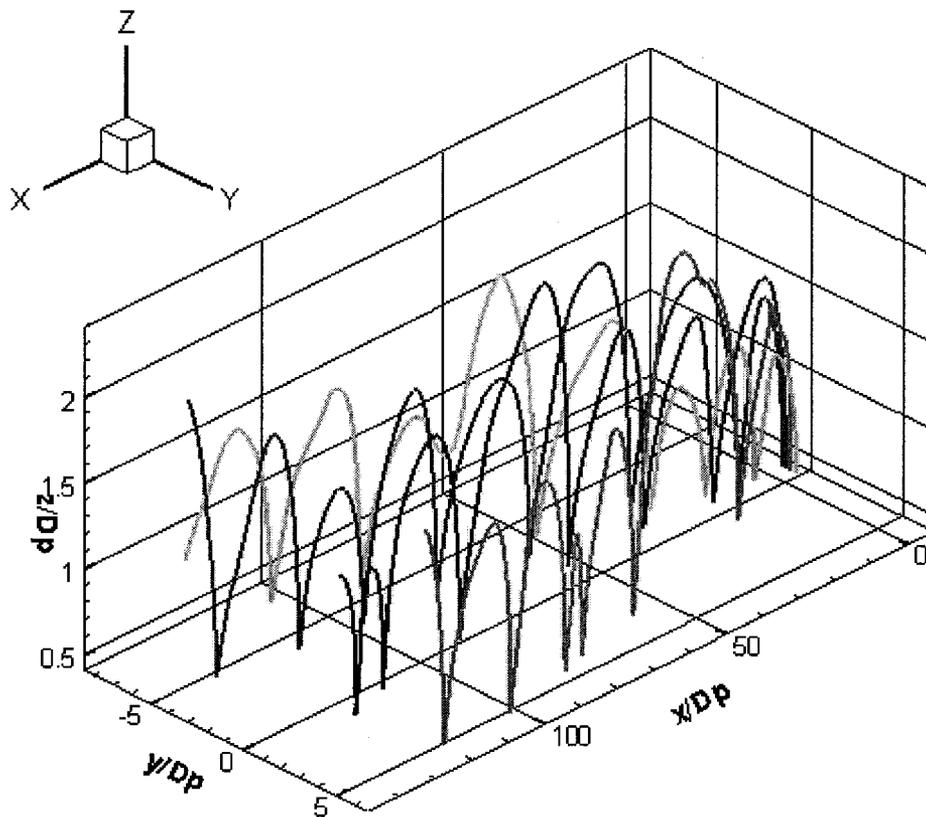


Figure 4-13: Three dimensional particle trajectory. No particle-particle collision $R_p=73$, $\tau_* / \tau_{*c}=1.5$.

4.7 Inter-Particle Collision Model

The next step requires adding an inter-particle collision algorithm to the numerical code. In order to simplify the computation of particle-particle collisions, only

binary collisions are taken into account. This assumption is valid only when the particle concentration is sufficiently low (less than 1%, Crowe et al 1998).

Post collision velocities for particles i and j (denoted with the superscript \wedge) are calculated using the following expressions based on the particle velocities just before the collision (denoted with the superscript \sim), presented by Crowe et al (1989) and Yamamoto et al (2001), based on the same assumptions as the conservation of momentum of Tsuji et al (1985) model.

$$\vec{\hat{u}}_{pi} = \vec{\tilde{u}}_{pi} + \vec{J} \quad \vec{\hat{u}}_{pj} = \vec{\tilde{u}}_{pj} - \vec{J} \quad (4-16)$$

$$\vec{\hat{\omega}}_i = \vec{\tilde{\omega}}_i + 5\vec{n} \times \vec{J} \quad \vec{\hat{\omega}}_j = \vec{\tilde{\omega}}_j + 5\vec{n} \times \vec{J} \quad (4-17)$$

where \vec{J} is the impulsive force exerted on particle i and \vec{n} is the normal unit vector directed from the center of the particle i to the contact point with the particle j (Fig. 4-14). Assuming spherical particles, known values of the restitution (e) and friction coefficient (f), \vec{J} can be calculated from (Tanaka and Tsuji (1991)):

$$\vec{J} = J_n \vec{n} + J_t \vec{t} \quad (4-18)$$

$$J_n = (1 + e) \vec{c} \cdot \vec{n} \quad (4-19)$$

$$J_t = \min \left[-f J_n, \frac{1}{7} |\vec{c}_{fc}| \right] \quad (4-20)$$

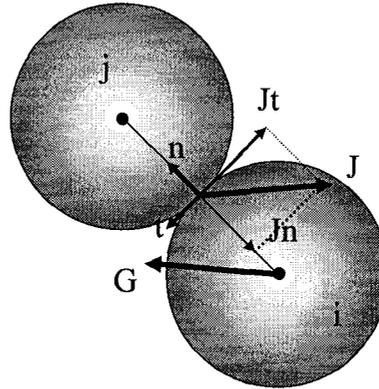


Figure 4-14: Inter-particle collision sketch. Definition of parameters.

In the above equations, \hat{t} is the tangential unit vector in the direction of the slip velocity of the particle j to i , given by $t = \vec{c}_{fc} / |\vec{c}_{fc}|$. \vec{c} is the relative velocity of the mass center $\vec{c} = \vec{u}_j - \vec{u}_i$ and \vec{c}_{fc} is the slip velocity between particle surfaces, defined as:

$$\vec{c}_{fc} = \vec{c} - (\vec{c} \cdot \vec{n})\vec{n} - \frac{1}{2}\vec{\omega}_i \times \vec{n} - \frac{1}{2}\vec{\omega}_j \times \vec{n} \quad (4-21)$$

The particle-particle collision model was implemented in the computational code previously tested. When the distance between the centers of the two particles is less than one diameter length, a particle-particle collision occurs. When the center of the particle is one-half a diameter above the wall, a wall-collision occurs. An example of the result obtained is presented in Fig. 4-15, where the trajectory of the center of two particles of diameter d_p . Particle 1 and Particle 2 are shown. The plot shows the change in the trajectory of both particles after hitting the wall (indicated with a filled arrow) and after colliding one against each other (blank arrow). Particle 2 hits the wall, come up, and collide with particle 1 above three times. The same results were animated using video tools (http://mpfg.engr.ucdavis.edu/Student/Andrea/two_phase.html) and the trajectories of both particles seem realistic. A snapshot of this video is shown in Fig. 4-16.

In the case of inter-particle collision, which lies in the small-scale level of validation, the verification of the algorithm is more complicated. Data for the small-scale level are generally scarce compared with the large-scale level, due to the difficulty of obtaining them (Heald et al. 2004). To the best of the writers knowledge, there are no experimental/numerical data available to validate the change of velocity of a particle moving in bed load after a collision.

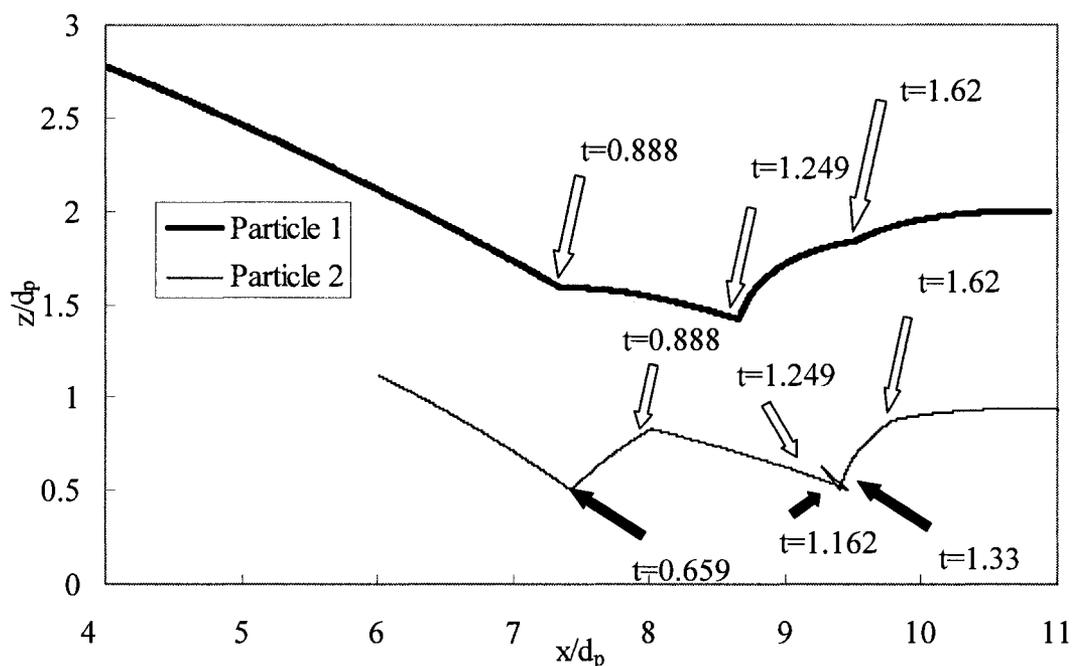


Figure 4-15: Particle 1 (dark line) and Particle 2 (light line) trajectories. Both lines represent the position of the center of each sphere. Blank arrows represent the dimensionless time where the both particles collide. Filled arrows represent the dimensionless time where Particle 2 hits the wall.
 $R_p = 73, \tau_* = 0.048.$

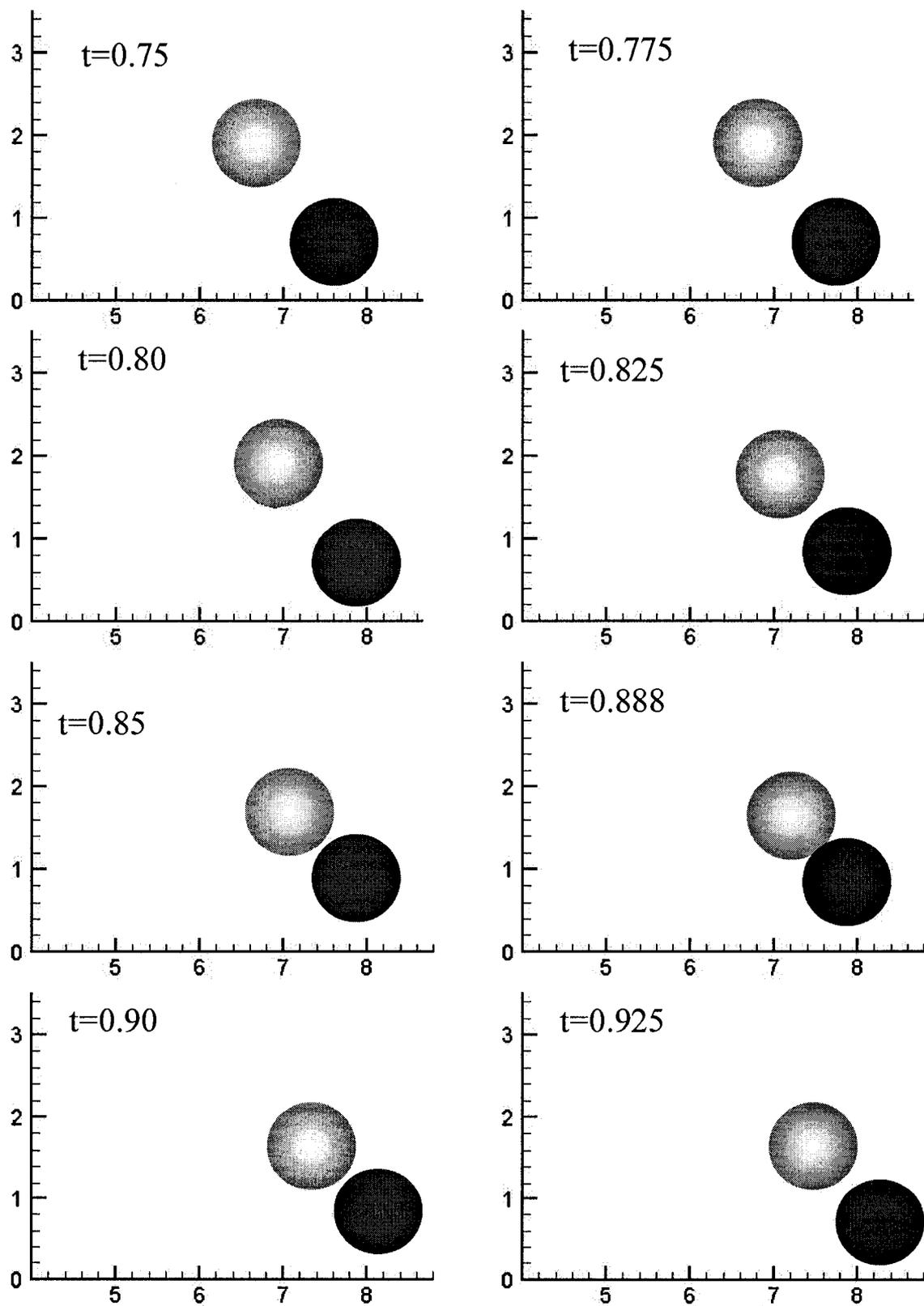


Figure 4-16: Snapshot of particles trajectories, showing a inter-particle collision at time $t=0.888$.

CHAPTER 5

PARTICLE MOTION UNDER A THREE-DIMENSIONAL TURBULENT VELOCITY FIELD

5.1 General Considerations

In Chapter 4, a new 3-D particle-tracking model including a new representation of the bed roughness was developed and validated. The numerical code was extended to address the multiple-particle problem for which an inter-particle collision algorithm was incorporated.

Up to this point, the velocity field has been represented using a semi-logarithmic expression; this velocity distribution is a very good approximation of the average flow conditions in boundary layer flows, but does not include turbulent fluctuations. The interaction of sediments with turbulence produce sedimentary structures such as bed load sheets (Schmeeckle and Nelson, 2003). To improve predictions of bed load transport, it is necessary to provide a more realistic velocity field capable of reproducing the turbulent characteristics of natural flows.

In this chapter, the results of a highly-resolved 3-D turbulent velocity field (HR3D) provided by Calo (2004) is coupled with the 3-D code developed in the previous chapters. By using this velocity field, the effect of turbulence on the particle motion close to the bed is studied in this chapter. One-way coupling between the moving particles and the flow field is assumed.

5.2 High-Resolution Three-Dimensional Velocity Field (HR3D). Simulation Description.

The simulation provided by Calo (2004) reproduces the ERCOFTAC T3A test case (Roach and Brierley, 1992) which involves a by-pass transition to turbulence on a flat plate due to free-stream turbulence passing above the plate (Fig. 5-1). The velocity field used in this work considers only a sub-section of the simulation, where the turbulence has already developed for some distance.

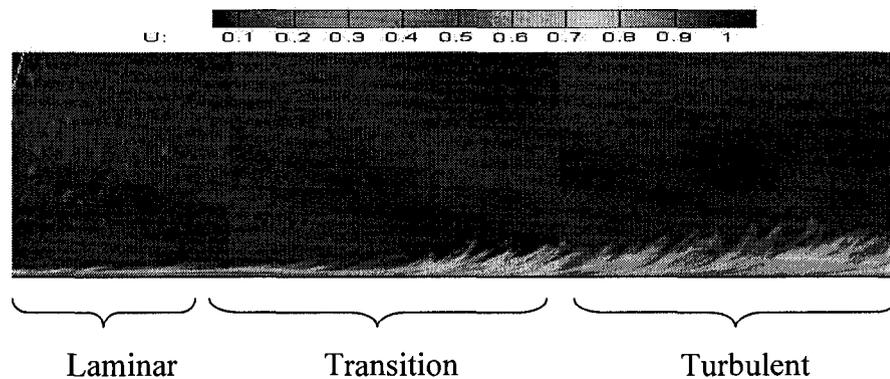


Figure 5-1: Computed stream-wise velocity component in the simulation domain. The extensions of the different zones are only approximate.

The HR3D simulation considers a total of 2048, 180 and 192 nodes in the stream-wise, wall-normal and span-wise directions, respectively. All lengths are made non-dimensional using the boundary layer thickness δ_0 at the inlet of the domain. The domain box is $620 \delta_0$, $40 \delta_0$ and $30 \delta_0$ in the longitudinal, vertical and transverse directions, respectively. The velocity scale selected was U_m , which represents the unperturbed stream-wise velocity of the fluid far away from the wall.

The grid points in the wall-normal direction are clustered close to the wall through a hyperbolic tangent stretching. The ratio between the largest and the smallest grid spacing is 60.

The boundary conditions are similar to those used by Jacobs and Durbin (2001). In the span-wise direction, periodic boundary conditions are applied to both velocity and pressure. For the pressure, a zero normal gradient is prescribed at the inflow and outflow boundaries, and at the wall a no-slip condition is enforced. At the top boundary, a Blasius outflow condition is applied to the wall-normal velocity component. The turbulent inflow was obtained using Rogallo's (1981) synthetic isotropic turbulence generation procedure.

The HR3D velocity field was obtained with in a second order, finite-volume code developed by Pierce and Moin (2001) and Jacobs and Durbin (2001) at the Center for Turbulence Research, Stanford University. In the code, the Navier-Stokes equations are solved on a staggered grid. All fluxes in the wall-normal direction are integrated implicitly using a Crank-Nicholson scheme (convective terms are linearized). The pressure is integrated fully implicitly. No closure for turbulence was used, which could converge to Direct Numerical Simulation (DNS) with appropriate spatial mesh sizes. (For the spatial steps used in Calo's simulation, the HR3D is close to be DNS but it can not be rigorously considered DNS).

The HR3D simulation used in this chapter considers a value of the Reynolds number, defined as $Re_\delta = U_m \delta_0 / \nu$, equal to 795. Given the inherent differences between the boundary layer in a flat plate and in an open-channel flow, it is necessary to discuss in detail the meaning of the coupling of the particle tracking code with the HR3D turbulent velocity field.

5.2.1 Boundary-Layer Flows in a Flat Plate and in Channels

It is generally accepted that a high degree of similarity exists between boundary layer flows over flat plates (as the one describe above) and channel flows (especially open-channel flows, which are the interest of this work, see Gad-El-Hak, 2000). Obviously, the most important difference is the spatially developing character of the former. In order to assess the feasibility of using Calo's velocity field as a surrogate of the velocity field in a channel, the features of both types of flows are discussed below. A short but very useful study of similarities and differences between turbulent boundary-layer, flat-plate flows, and channel flows is presented by Nieuwstadt and Bradshaw (1997) and Ashrafian (2004). The main differences and similarities are as follows:

- 1) The turbulent/non-turbulent interface, an important feature of the boundary-layer, flat-plate flows, is absent in channel flows.

- 2) The free surface (present only in open-channel flows) suppresses the vertical movement of eddies, as opposed to the outer layer of a flat-plate. (Nezu and Nakagawa, 1993). Experiments have shown that the turbulence intensity in the wall-normal direction decreases rapidly very close to the free surface ($z/H_c > 0.9$, where H_c represents the open-channel depth).

- 3) In channel flows, excess energy is transported by turbulent diffusion to the free-surface region where it compensates the dissipation (Nezu and Nakagawa, 1993), whereas in boundary-layer flows, the extra energy is used to sustain the thickening of the layer (Jiménez, 2004).

4) The wake strength for channel flows is much smaller than the counterpart in boundary-layer flows, due to the higher intermittency of turbulence in the outer region (Johnson, 1998).

Despite all the differences previously listed between boundary-layer flows over a flat plate and open-channel flows, the logarithmic profile which describes the mean stream-wise velocity is applicable to all wall-bounded flows (Gad-El-Hak, 2000). Wei et al. (2005) present "compelling evidence" of the logarithmic character of the mean profile in a large section of both channel and boundary-layer, flat-plate flows. Therefore, the use of the turbulent velocity field calculated in the HR3D simulation is capable of reproducing the mean velocity characteristics of an open-channel flow. Further, because saltating motion occurs close to the bed ($z/\text{depth} < 0.05$) the use of the HR3D velocity field yields accurate results. In order to verify whether the turbulence parameters defined for an open-channel flow are well reproduced in the HR3D simulation, it is necessary first to analyze both flows through a dynamic similarity analysis.

5.2.2 Dynamic Similarity

Two problems are dynamically similar if the variables of one problem can be related to the variables in the other. The simplest type of similarity occurs when two situations have different dimensional variables but the same non-dimensional variables (Panton, 2000). In numerical simulations, two problems that are dynamically similar will share the same non-dimensional solution.

Consider an incompressible flow in which both viscous and gravity effects are important (Kundu and Cohen, 2004). The governing equation for the conservation of momentum is:

$$\rho \left(\frac{\partial \vec{v}}{\partial t} + \vec{v} \cdot \nabla \vec{v} \right) = -\nabla p - g + \mu \nabla^2 \vec{v} \quad (5-1)$$

The equations can be made dimensionless by defining a characteristic length scale l and a characteristic velocity scale u . Eq. (5-1) is expressed in dimensionless terms as:

$$\rho \left(\frac{\partial \vec{v}_f}{\partial t} + \vec{v}_f \cdot \nabla \vec{v}_f \right) = -\nabla p - \frac{1}{Fr^2} + \frac{1}{Re} \nabla^2 \vec{v}_f \quad (5-2)$$

where \vec{v}_f is the dimensionless fluid velocity vector, $Fr = \frac{u}{\sqrt{gl}}$ is the Froude number and

$Re = \frac{ul\rho}{\mu}$ is the Reynolds number.

In order to use the HR3D simulation for a boundary-layer flow as the velocity field for saltating particles in open channels, it is necessary to check if the conditions for dynamic similarity are met. To achieve dynamic similarity between two flows, both Fr and Re have to be equal (when both viscous and gravitational effects are important). For a flow past an object in a homogeneous fluid (such as particles saltating at the bottom of a channel), gravity is only important if surface waves are generated (Kundu and Cohen, 2004). In boundary-layer flows, the Reynolds number is the only dimensionless number that needs to be equal in the two flows. Thus, the HR3D velocity field represents accurately the flow field interacting with salting particles, provided that the same Reynolds number is preserved.

The HR3D simulation of a flow on a flat plate has been compared with the experimental data obtained from a wind-tunnel experiment (Roach and Brierly, 1992); on the other hand, the focus of this thesis is on the flow of water in a open-channel. Comparing experimental data and numerical simulation results with two different fluids is customary. Rouse et al. (1958) used this procedure to study hydraulic jumps in water using wind-tunnel results, stating that as long the mean flow patterns are geometrically similar and the changes in energy are comparable, then it is not only the geometrical similarity, but also the dynamic similarity is imposed. The following section is devoted to corroborate these assumptions.

5.2.3 Simulated Flow Characteristics

Time-averaged velocities obtained from the HR3D simulation should be close to the analytical expressions for the mean velocity in a turbulent open channel. (i.e. the law of the wall). To check this, it is necessary to relate the velocity and time scales used in the HR3D simulation. The particle model considers the wall-friction (shear) velocity u_* and the particle diameter (d_p), as the velocity and length scales, with those of the law of the wall, U_m and δ_0 , employed in the HR3D simulation. Using the time-averaged values of the simulated velocity field (expressed in dimensionless terms as u^f/U_m), a relationship between velocity scales is found:

$$\frac{u^f}{U_m} = \frac{u^f}{u_*} \frac{u_*}{U_m} \quad (5-3)$$

From Eq. (5-3), u_f/u_* can be obtained for comparison with the law of the wall.

Defining $A = u_*/U_m$, the length scales can be related as follows:

$$\frac{z u_*}{\nu} = z^+ = \frac{z}{\delta_0} \text{Re}_\delta A \quad (5-4)$$

where the location of the nodes in the HR3D simulations are expressed in terms of z/δ_0 .

From Eqs. (5-3) and (5-4), the values are u^f/U_m , z/δ_0 and Re_δ are known from the HR3D velocity field. The value of A was determined by adopting the value that produces the best fit between the HR3D numerical results and the semi-logarithmic expression for the velocity profile in the case of a smooth channel. The value of A was determined to be equal to 0.045, as shown in Fig. 5.2.

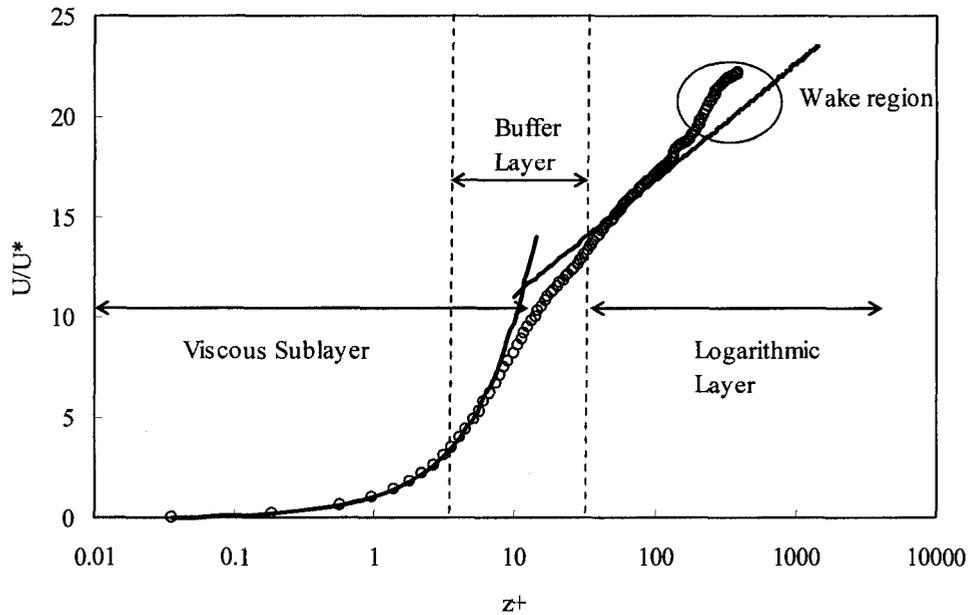


Figure 5-2: Comparison between the mean stream-wise fluid velocity obtained from the HR3D simulation and known expressions for the velocity profile in a turbulent open-channel. Solid lines represent analytical expressions, circles represent data obtained from numerical simulation data.

Because the Reynolds number must be preserved between the open-channel flow and the HR3D flow, another relationship between non-dimensional numbers can be developed:

$$\text{Re}_\delta = \frac{U_m \delta_0}{\nu} = \frac{U_m}{u_*} \frac{\delta_0}{d_p} \frac{u_* d_p}{\nu} = R_p \sqrt{\tau_*} \frac{1}{A} \frac{\delta_0}{d_p} \quad (5-5)$$

It becomes clear from Eq. (5-5), that δ_0 / d_p can be obtained for a given particle size and flow condition (u_*).

Nezu and Nakagawa (1993) presented results of extensive experimental research on turbulent open-channel flows. They obtained universal expressions for turbulence intensities, denoted u_f' , v_f' , and w_f' which represent the stream-wise, span-wise and wall normal components of the fluid turbulence intensity, respectively (defined as $u_f' = \sqrt{u_f^{f2}}$, $v_f' = \sqrt{v_f^{f2}}$, and $w_f' = \sqrt{w_f^{f2}}$, where u_f^f , v_f^f , w_f^f are the turbulent fluctuations of the fluid velocity in the stream-wise, span-wise and wall-normal directions, respectively) and turbulent kinetic energy (TKE_f), normalized with the friction velocity and the friction velocity squared, respectively. The relations are:

$$\frac{u_f'}{u_*} = 2.3 \exp\left(-\frac{z}{H_c}\right) \quad \frac{v_f'}{u_*} = 1.27 \exp\left(-\frac{z}{H_c}\right) \quad \frac{w_f'}{u_*} = 1.63 \exp\left(-\frac{z}{H_c}\right)$$

$$\frac{TKE_f}{u_*^2} = 4.78 \exp\left(-\frac{2z}{H_c}\right) \quad (5-6)$$

In the case of a boundary-layer flow over a flat-plate, there is no clear definition for H_c . In order to compare the numerical results of turbulence intensities and turbulent kinetic energy between the flat-plate and boundary layer flows, H_c was assumed to be equal to the depth of the simulation space.

Distributions for the flow turbulence intensity of the flow in each direction and for the turbulent kinetic energy (TKE_f) computed from the HR3D simulation are compared with the experimental regressions developed by Nezu and Nakagawa in Fig. 5.3. In general, good agreement is noticed between the different vertical profiles obtained from the numerical simulation and the empirical expressions.

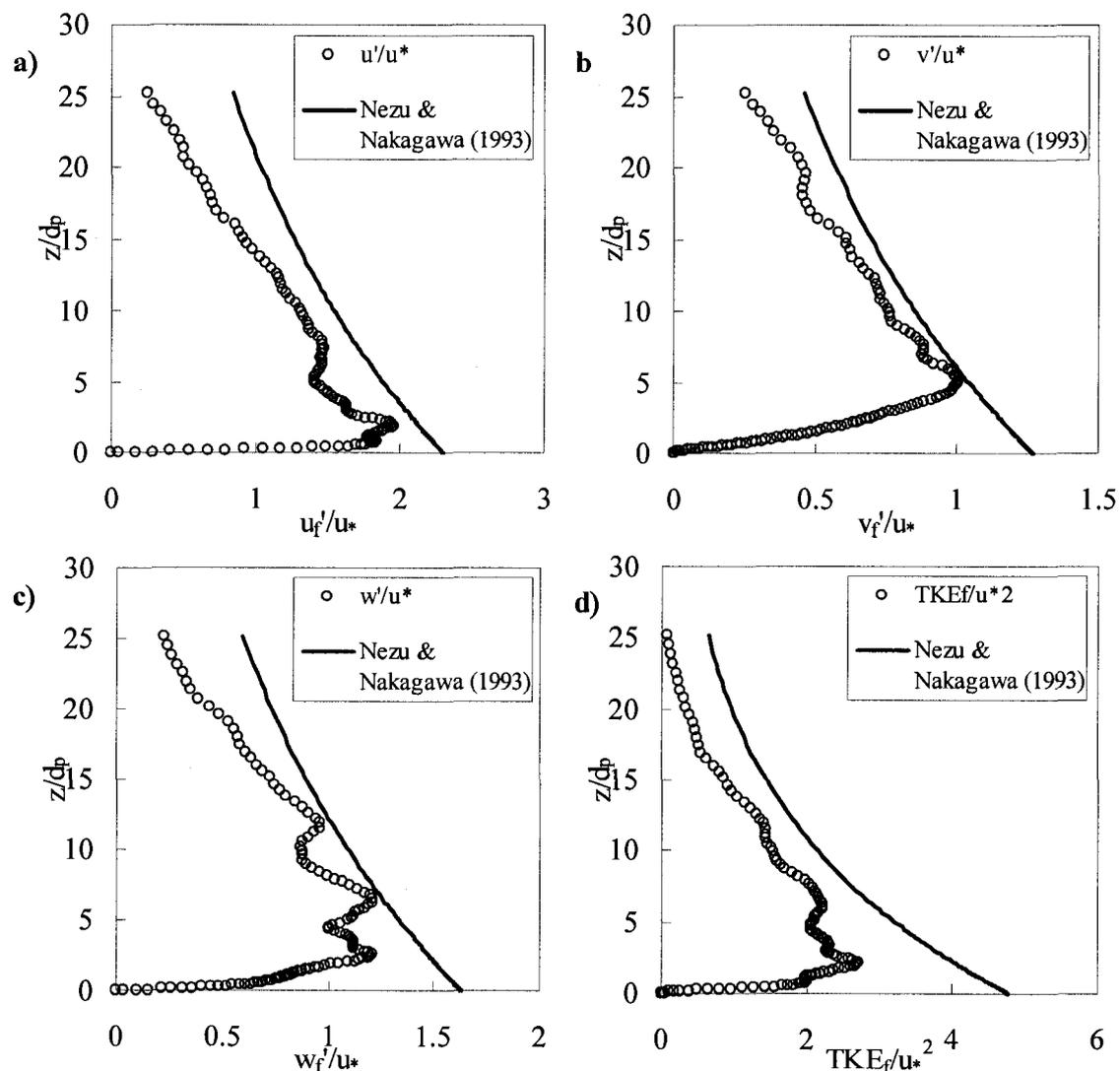


Figure 5-3: Turbulence intensities and turbulent kinetic energy from the numerical simulation (circles) compared with the experimental regressions suggested by Nezu and Nakagawa (1993) (solid line). a) Stream-wise component. b) Span-wise component c) Wall-normal component d) Turbulent kinetic energy of the flow.

Results from the HR3D simulation present the expected exponential decrease with the distance from the wall, in all four variables.

The turbulence intensities in the span-wise and wall-normal directions showed a better agreement with the regressions than those in the stream-wise direction. This effect is produced by the arbitrary definition of H_c . By changing the value of the open-channel depth, the values of the turbulence intensity in the stream-wise direction can be better adjusted to the corresponding empirical expressions.

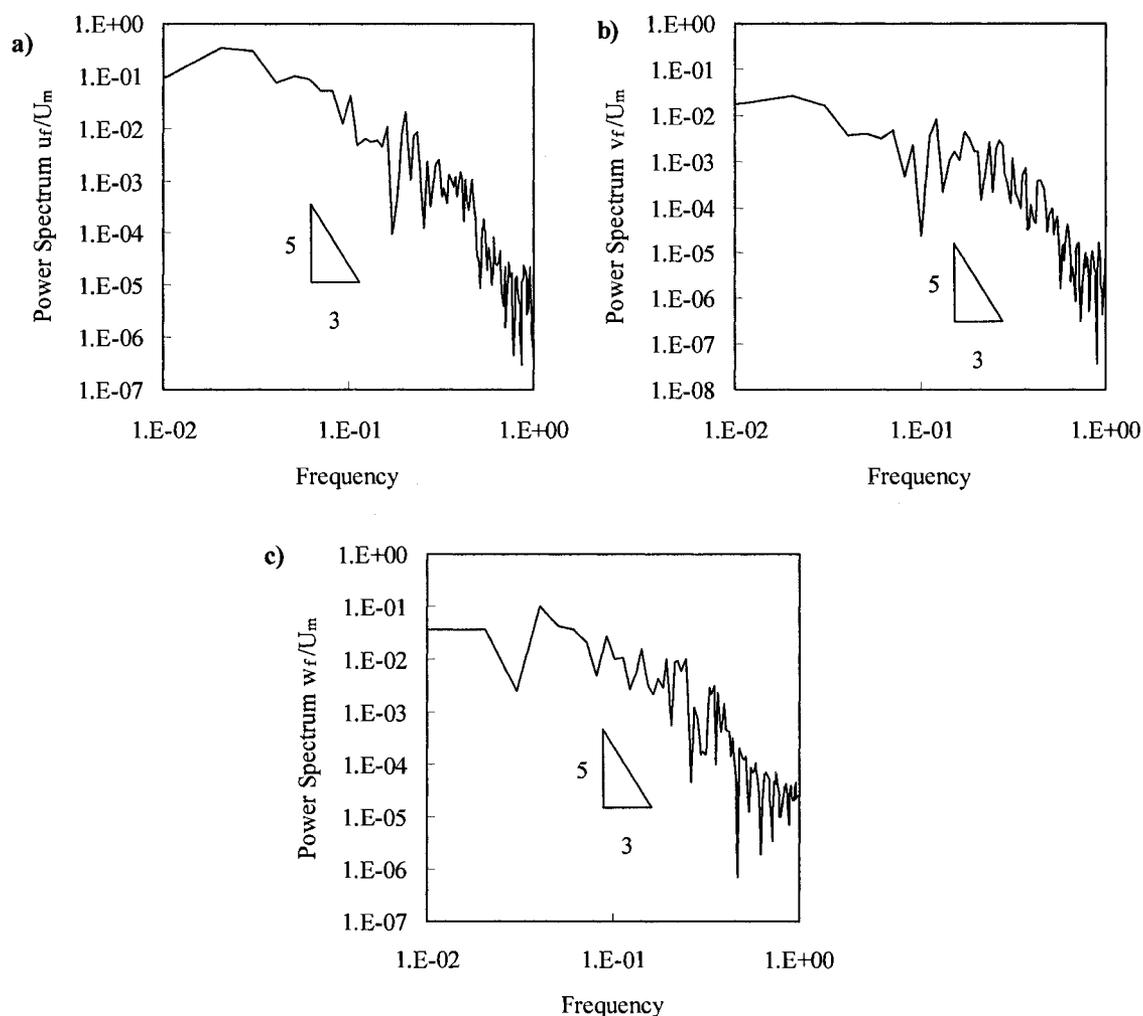


Figure 5-4: Wave-number spectrum of the velocity component at the turbulent region. a) stream-wise, b) wall normal and c) span-wise direction.

Finally it is necessary to corroborate that the wave-number spectrum obtained from the simulation follows the $-5/3$ Kolmogorov law in the inertial sub-range. Fig. 5-4 shows that for every velocity component this law is valid.

It is concluded that the HR3D velocity field represents satisfactorily the flow in an open channel. Thus, the HR3D was used to track particles moving close to the bed in the following sections. It is assumed that the effect of the particle volume on the surrounding fluid is negligible.

5.3 Three-Dimensional Simulation of the Motion of a Single Particle Under a Simulated Velocity Field.

The 3-D particle tracking model presented in Chapter 4 was coupled with the information provided by the HR3D velocity field. In this case, the motion of the particles was computed using Eq. (4-4), where the expression for the forces considered the fluctuating and non-uniform nature of the 3-D velocity field.

Collision *DBA* sub-model (Section 4.4.1) was used, and the best-fit values for the friction and restitution coefficients for each particle size were considered in the simulations presented below.

5.3.1 Validation of the Model for a Single Particle

Validation of the one-way coupling model for a single particle is presented in this section. Simulations of the experimental conditions presented by Niño and Garcia (1998a,b) and Lee and Hsu (1994) were performed. The numerical model was run for a

simulation time long enough to have meaningful statistics. In order to remove the effect of the initial conditions, the first jumps were not considered in the statistical analysis.

The collision algorithm was modified to be consistent with the smooth condition of the bed in the HR3D simulation. In Chapter 4, the moving particle and the particles that composed the bed had the same size. Thus, the values of r_{max_1}/d_p and r_{max_2}/d_p were obtained by replacing the incidence angle (θ_{in} and α_{in} , respectively) in Eq. (4-14) and assuming that the bed contact angle (θ_b and α_b , respectively) were equal to $\theta_{crit} = 30^\circ$, defining θ_{crit} as the maximum angle where the moving particle can hit the bed.

In this chapter the diameter of the particles in the bed (d_{pbed}) is 0.3 mm to maintain the smooth bed characteristic of the bed. Therefore, the value of θ_{crit} changes. Using geometrical properties between the moving particle and the particles composing the bed, θ_{crit} is obtained from the following expression:

$$\theta_{crit} = \arcsin\left(\frac{d_{pbed}}{d_{pbed} + d_p}\right) \quad (5-7)$$

Figs. 5-5 and 5-6 present the numerical results obtained for a particle of $R_p=73$ and $R_p=250$ respectively. The figures depict the dimensionless particle jump height and length, the mean stream-wise velocity and the mean particle spinning, which are compared with the experimental data available. For both particle sizes selected, good agreement was found between the numerical simulation and the experimental data. A slight over prediction of the particle jump length, height and mean stream-wise velocity is observed, and it could be related to the absence of particle-particle collision, feature that is always present in nature/experiments and it is not modeled in this simulation.

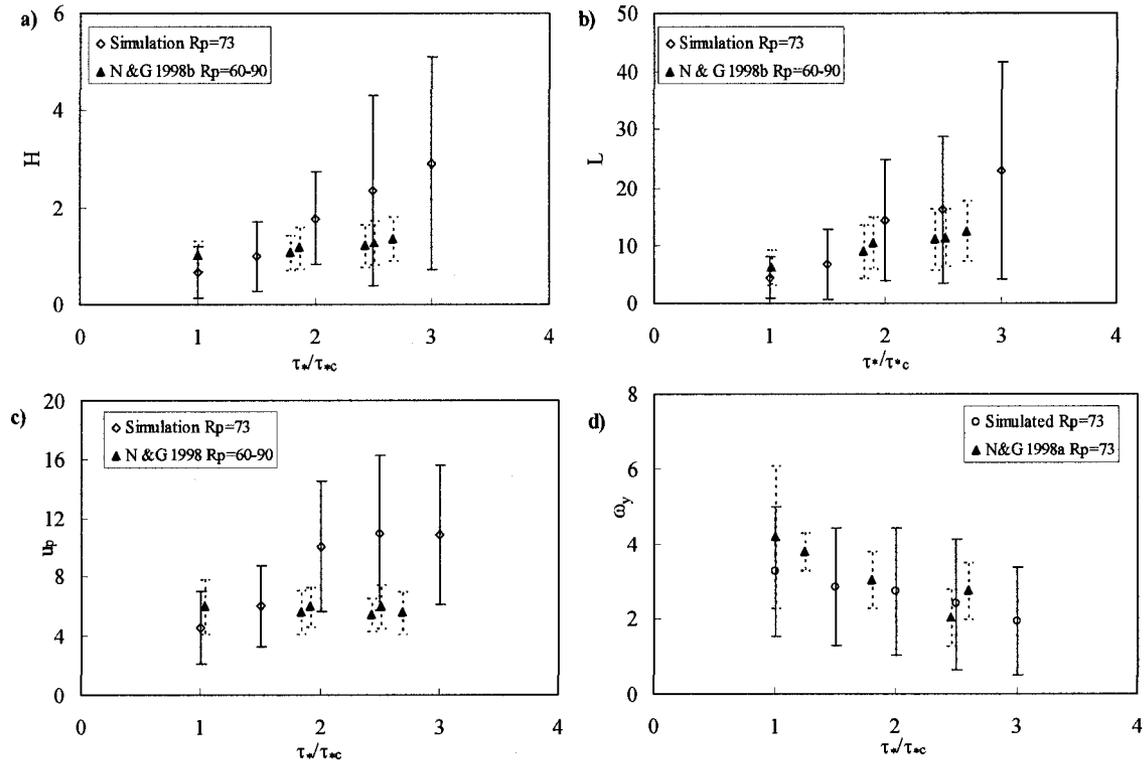


Figure 5-5: Comparison of simulations with experimental data for the case of a particle moving in a flume and rebounding with the wall. The figure shows results associated with: a) the particle jump height (H). b) the particle jump length (L). c) the particle stream-wise mean velocity. d) the mean particle spinning. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. Single-particle simulation $R_p=73$.

Results obtained with the HR3D velocity field show a slight increase in the dimensionless particle jump length, height and velocity in the case of $R_p=73$ with respect to the values obtained in a non-turbulent velocity field (see Fig. 4-4). On the other hand, particle rotation is slightly smaller than the values obtained in the non-turbulent case (see Fig. 4-7).

For a larger particle ($R_p=250$) the results obtained with the HR3D velocity field show a better agreement with the experimental data as compared with the non-turbulent case (see Fig. 4-6). In this case, particle jumps are smaller in length and height and the particle velocity is slightly smaller than the case presented in Chapter 4. This can be

explained by the reduction of the physically possible angles under which the particle hits the bed. In this case the difference between the diameter of moving particles and the particles composing the bed is larger, and therefore the range of variation of the collision angles is smaller.

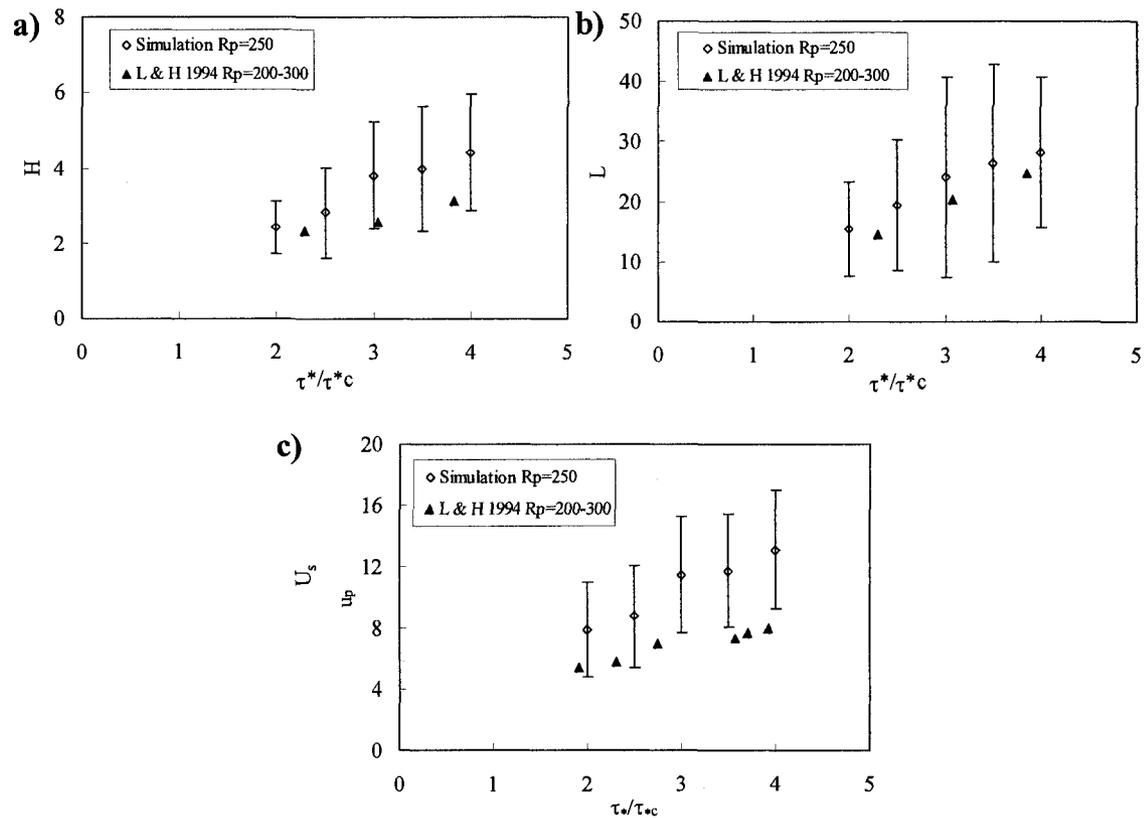


Figure 5-6: Comparison of simulations with experimental data for the case of a particle moving in a flume and rebounding with the wall. The figure shows results associated with: a) the particle jump height (H), b) the particle jump length (L) and c) the particle stream-wise mean velocity. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. Single particle simulation $R_p=250$.

It is also observed that the standard deviation for both particle sizes is larger than in the non-turbulent case, precisely due to the flow fluctuations. Differences between the non-turbulent and turbulent simulations show the importance of the temporal and spatial variation of the velocity field.

The height of the jumps for a particle of $R_p=73$ varies between 0.5 to 2 particle diameters, which indicates that the particle moves in a zone equivalent to 15 to 50 wall units from the bed. For particles of $R_p=250$, the particle moves mainly in a zone defined between 100 to 300 wall units from the bed (defined by using ν/u_* as a length scale).

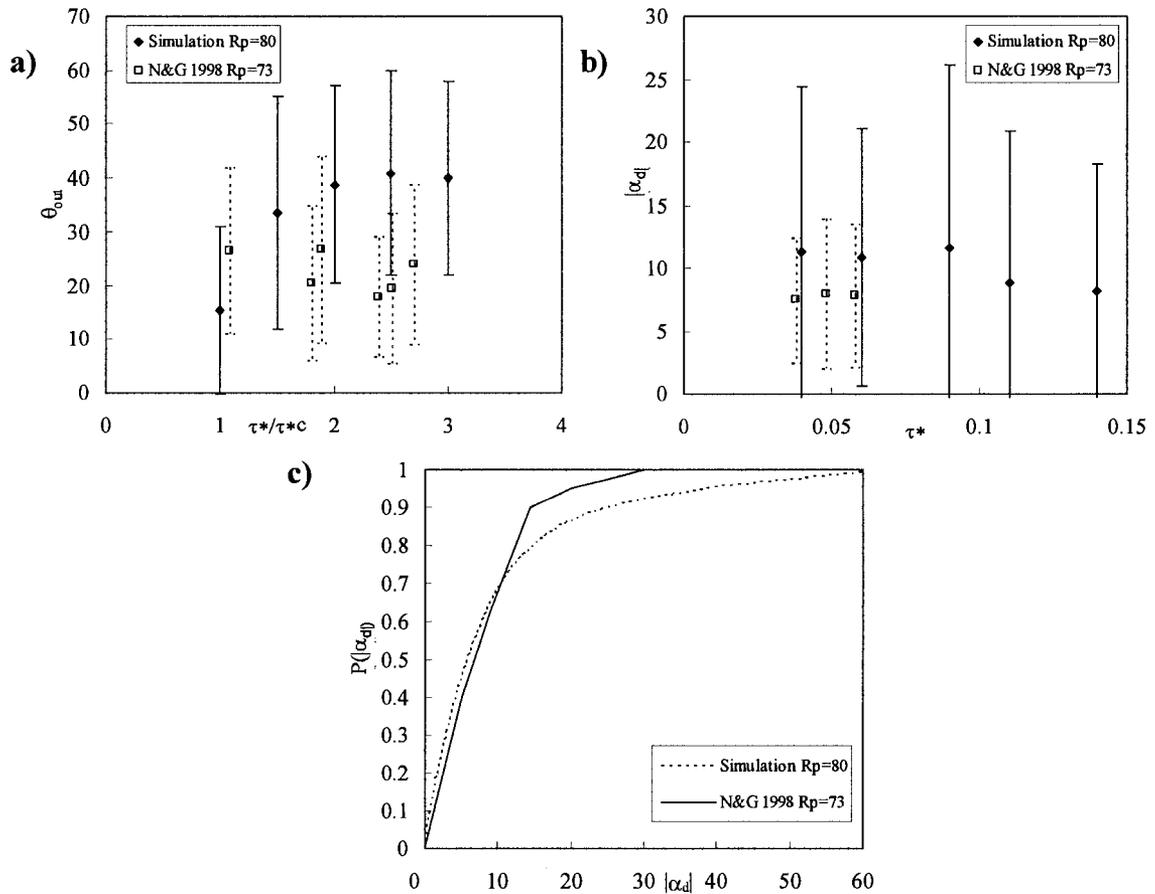


Figure 5-7: Comparison of simulations with experimental data for the case of multiple particles moving in a flume, rebounding with the wall and colliding among themselves. The figure shows results associated with: a) the take-off angle after a collision (θ_{out}), b) the absolute value of the lateral dispersion angle (α_d) and c) the cumulative probability distribution of the absolute value of the deviation angle. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. $R_p=80$.

Further validation was obtained comparing numerical simulation with the experimental data provided by Niño and García (1998a). Those data contain detailed information regarding the characteristics of the collision in terms of take-off angles and

lateral dispersion. Fig. 5-7 presents the comparison between the numerical and experimental results, showing a good agreement in each of the parameters of study. Comparing similar results obtained for the non-turbulent case, these results show larger scatter for all variables.

5.3.2 A New Turbulence Filter

Fig. 5-8 shows the time series of a particle velocity in each direction. It can be seen that a random pattern is superimposed to the "mean" particle velocity in the stream-wise and wall-normal direction. The "mean" particle velocities describe a quasi-periodic behavior.

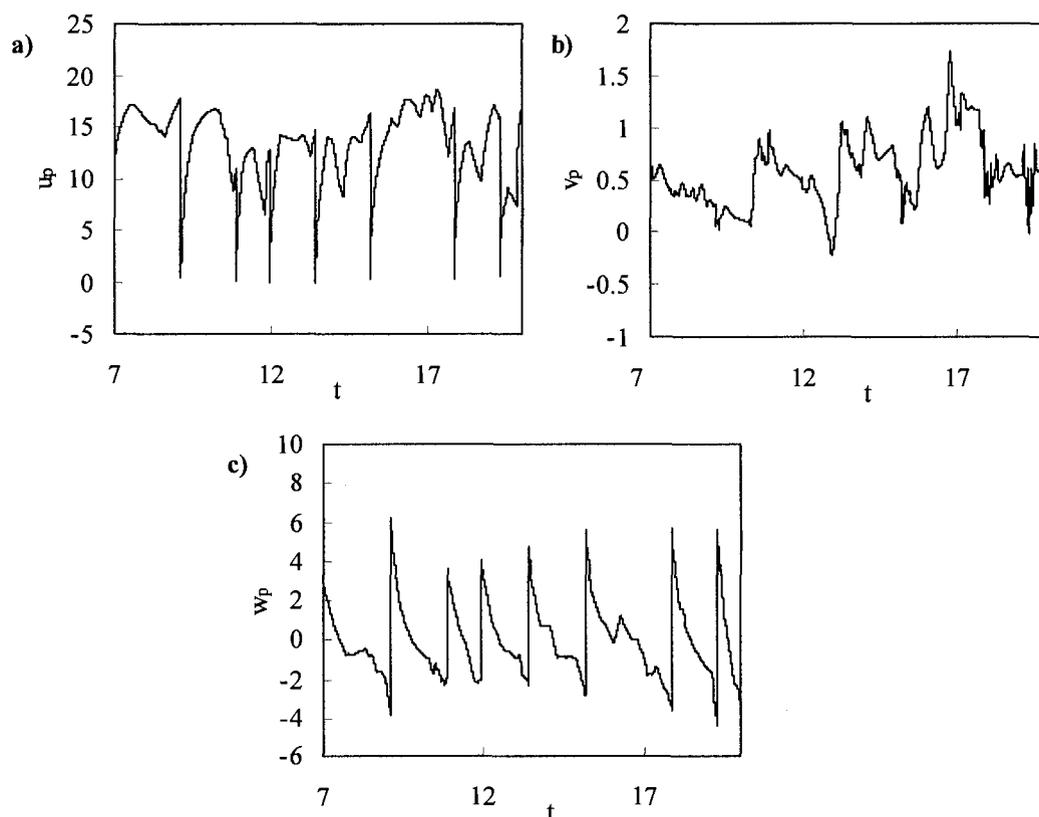


Figure 5-8: Single particle velocity time series using a high resolution three dimensional velocity field. $R_p=73$, $\tau_*/\tau_{*c}=1.5$. a) stream-wise component. b) span-wise component. c) wall-normal component.

The previous facts indicate that using a standard temporal average of the particle velocity (which has been generally used in the case of particles moving in an unbounded flow, see Sommerfeld (2003) and Sommerfeld and Kussin (2003)) is not adequate in this case. For instance, when averaging on the time the wall-normal component of the particle velocity, the mean value is close to zero (because the value of the wall-normal velocity is positive when the particle is going up, and negative when the particle is going down). Therefore, the particle turbulence intensity in this direction would be calculated as the actual velocity value, instead of considering only the fluctuations due to the flow turbulence. Thus, it is necessary to design a new filter

The new filter needs to be capable of separating the turbulence effect from the mean particle velocity conditions, considering both the non-stationary characteristic of the particle velocity and the existence of the particle-wall collision event.

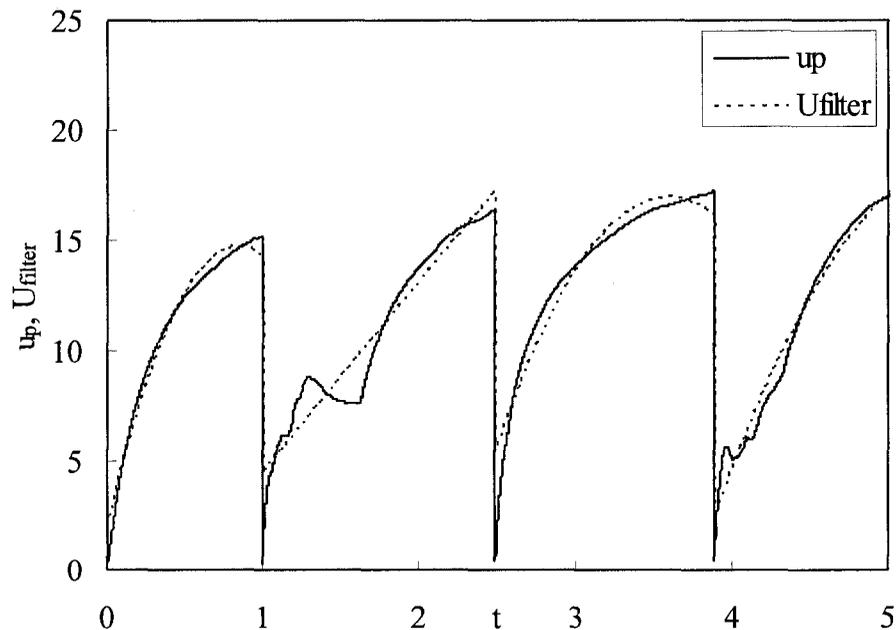


Figure 5-9: Single particle velocity and the new turbulence filter proposed. $R_p = 73$, $\tau_p / \tau_c = 1.5$.

In Figs. 4-10 and Fig. 5-8 c) it is observed that every time the particle collides with the wall, each velocity component describe a “jump-like” pattern, which in principle can be numerically modeled using a second order polynomial. Depending on the time duration of the jump, different coefficients of the second order polynomial function of the time define the best fit for the particle velocity along an individual jump. Therefore, by using a specific second order polynomial that depends only on the jump duration a filter is defined to separate the influence of the flow turbulence on the particle velocity.

The value of the coefficients of the second order polynomial are defined through the least squares fit.

In between two consecutive collisions of the particle with the bed, it is possible to define the particle velocity fluctuation at each time step as the difference between the instantaneous particle velocity and the value of the filter at that time. Fig. 5-9 shows the definition of the filter for the stream-wise component of the velocity. It is worth noticing that after the particle the particle hits the wall, the proposed filter does not accurately reproduce the mean particle velocity after the collision, where the particle acceleration is the highest (see Fig. 5-9). It will produce a peak in the values of the particle turbulence intensity close to the bed that needs to be disregarded from the analysis.

5.3.3 Particle Turbulence intensity and Turbulent Kinetic Energy Computations

Analogously to the turbulence intensity defined for a turbulent flow, the particle turbulence intensity u'_p in the stream-wise direction can be defined as:

$$u'_p = \sqrt{(u_p - U_{filter})^2} \quad (5-8)$$

where u_p is the instantaneous particle velocity, U_{filter} represents the value of the turbulence filter at the same time and the over bar is the ensemble average over the jumps. The same definition is applicable for the transverse (v'_p) and wall-normal (w'_p) turbulent intensities. The particle turbulent kinetic energy (TKE_p) is then defined as:

$$TKE_p = \frac{1}{2} (u'_p{}^2 + v'_p{}^2 + w'_p{}^2) \quad (5-9)$$

Assuming that the greatest variation of the parameter of study is in the wall-normal direction, Fig. 5-10 presents the ensemble average over the jumps of particle turbulence intensity and turbulent kinetic energy at different distances from the wall. The plot on the left shows the values of the particle turbulence intensity in each direction and the turbulent kinetic energy of the particle as a function of the particle position in the vertical. The plot on the right shows the variation of the particle turbulent kinetic energy as a function of the flow shear stress.

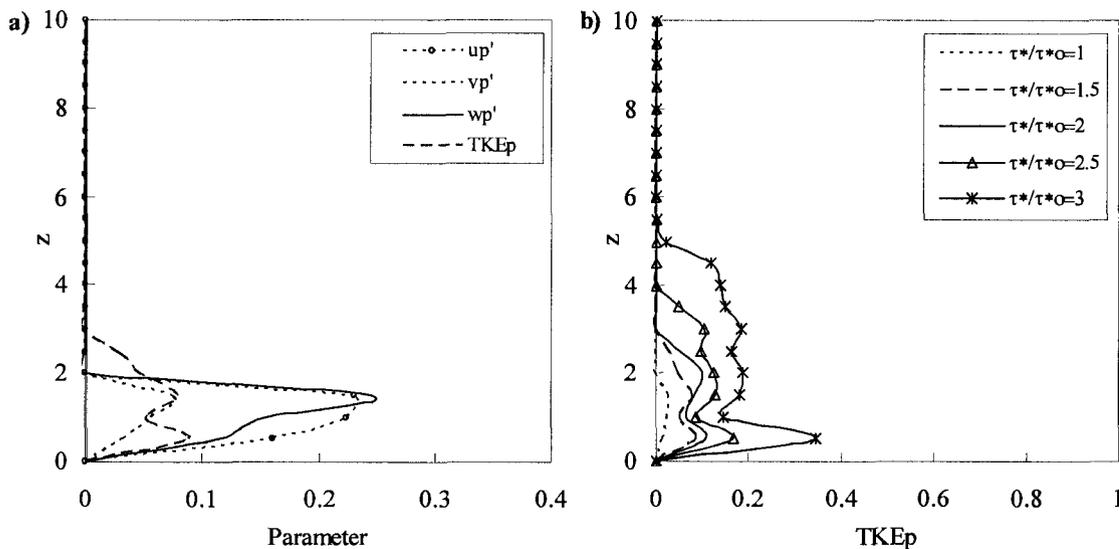


Figure 5-10: Distribution of the particle turbulence intensity and particle turbulent kinetic energy for a single particle moving in a simulated velocity field. $R_p = 120$. a) Wall-normal variation of the parameters for $\tau_* / \tau_{*c} = 1.5$. b) Variation of the turbulent kinetic energy as the flow condition changes.

It is observed that a particle saltating close to the bottom has a maximum value of the particle turbulence intensity and turbulent kinetic energy at a distance of approximately two particle diameters from the bottom. Fig.5.10a shows that the particle turbulence intensity is larger for the wall-normal and stream-wise components as compared with that in the span-wise direction.

Increasing the flow shear velocity (Fig.5-10b), increases the particle turbulent kinetic energy and the zone of influence of the TKE . As the flow velocity increases, the filter produces a new peak value close to the bottom of the bead. This new peak has a mathematical explanation, rather than a physical one, and it is disregarded in the remaining analysis (as explain above).

5.4 Three-Dimensional Tracking of Multiple Particles Under Simulated Turbulent Velocity Field

The next step was to study a more realistic scenario, where multiple particles of the same size move close to the bottom of a channel. The effects of collision between moving particles may or may not considered in this section.

5.4.1 Validation of the Model for Multiple Particles

A series of simulations for two different particle sizes with $R_p=80$ and 120 (equivalent to 0.7 mm and 1 mm, respectively) under different flow conditions (shear velocity ranging from 0.023 to 0.0417 m/s) were undertaken to evaluate the influence on the flow velocity and the particle size in the TKE_p (see below). The results of this simulation are also used to validate the model, by comparing them with the experimental

data presented by Niño and Garcia (1998b) for particles of $R_p=73$. The computations consider a constant particle concentration (0.1%) and binary collisions between particles. The results are presented in terms of ensemble averages over all moving particles.

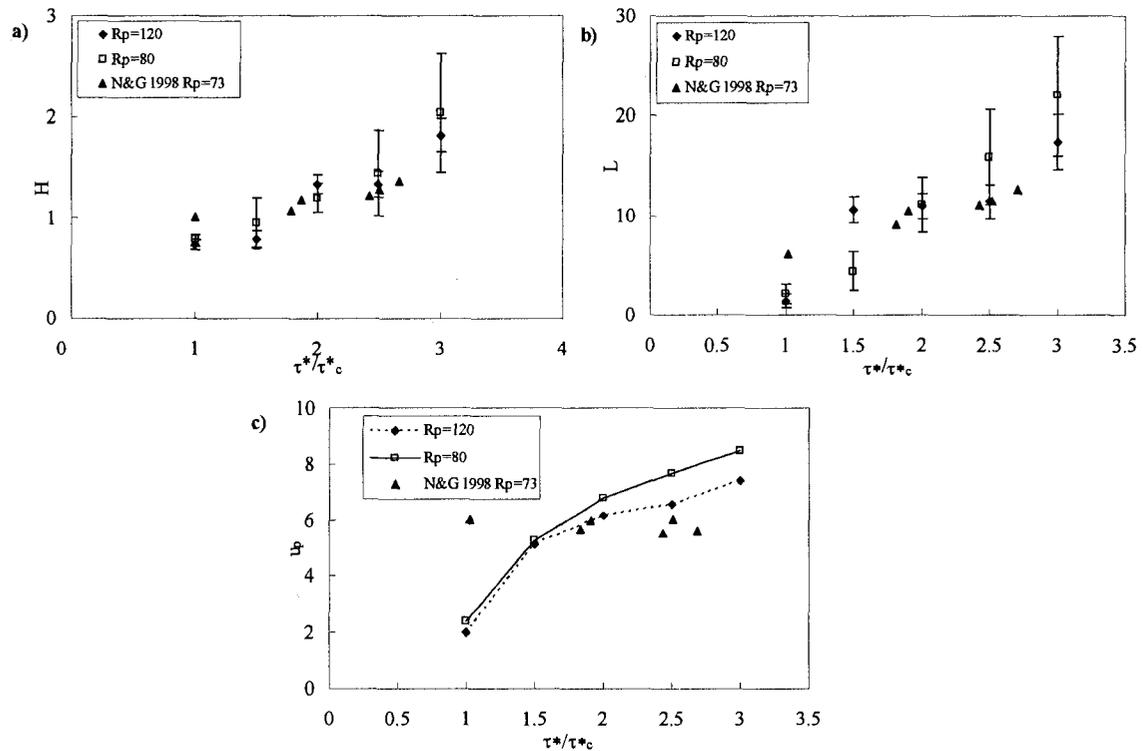


Figure 5-11: Comparison of simulations with experimental data for the case of multiple particles moving in a flume, rebounding with the wall and colliding among them. The figure shows results associated with: a) the particle jump height (H), b) the particle jump length (L) and c) the particle stream-wise mean velocity. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. $R_p=80$ and $R_p=120$.

Fig. 5-11 shows a very good agreement of the average value of particle jump height and length, as well as the mean stream-wise component of the particle velocity. The mean values of L and u_p obtained for the larger particle ($R_p=120$) are slightly smaller than the mean values obtained for the smaller particle ($R_p=80$). As compared with Fig. 5-6, these results show that the inclusion of inter-particle collision in the model

produces better predictions in terms of particle jump length, height and mean stream-wise particle velocity.

When considering multiple particles, it is possible to calculate the bed load rate and to compare the results with widely used expressions of bed load transport (Julien, 1998). The volumetric sediment transport rate q is calculated directly by counting the number of particles that move through a specific location of the simulated channel, in a given period, and multiplying this result by the particle volume. The dimensionless

volumetric bed load rate q^* is then calculated as $q^* = \frac{q}{\sqrt{g R d_p^3}}$. Fig. 5-12 presents the comparison of results obtained with the model and known expressions of bed load transport. Good agreement between the model and the analytical expressions is found.

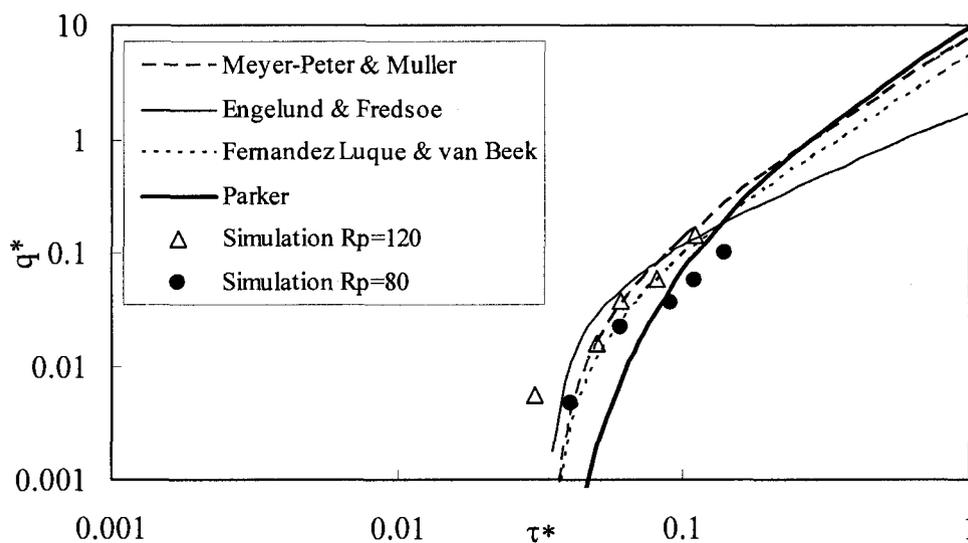


Figure 5-12: Simulated dimensionless bed load transport rates. Comparison with formulae by Meyer-Peter and Muller (1948), Engelund and Fredsoe (1976), Fernandez Luque and van Beek (1976) and Parker (1978).

The validation of the model considers both the wall-collision and the inter-particle collision algorithms. In the case of the collision with the bed, the results obtained

of the simulation for multiple particles with $R_p=80$ were compared with the experimental data provided by Niño and García (1998a), as presented in Fig. 5-13. The take-off and lateral dispersion angles are slightly larger than the ones obtained experimentally, but are within the range of the experimental errors. Overall, the simulations provide expected results.

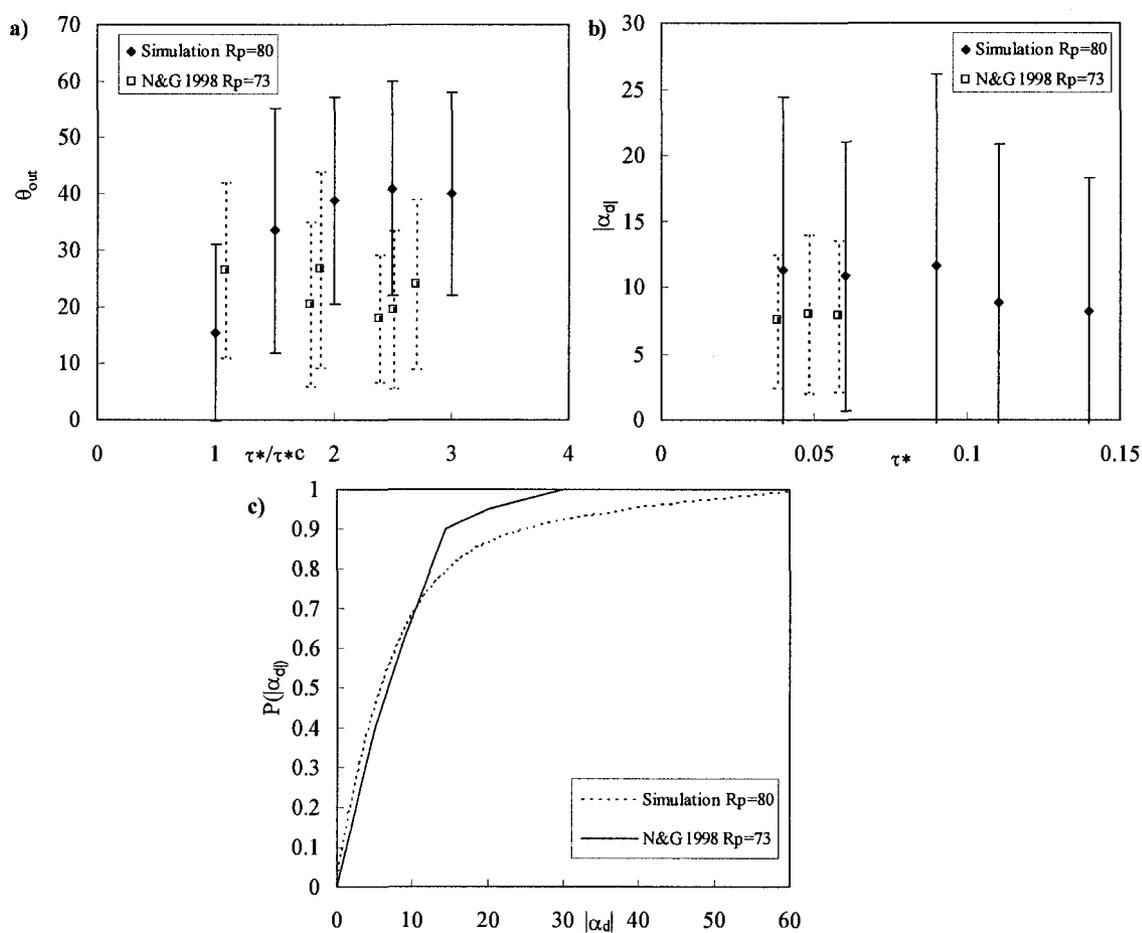


Figure 5-13: Comparison of simulations with experimental data for the case of multiple particles moving in a flume, rebounding with the wall and colliding among themselves. The figure shows results associated with: a) the take-off angle after a collision (θ_{out}), b) absolute value of the lateral dispersion angle (α_d) and c) cumulative probability distribution of the absolute value of the deviation angle. Symbols represent mean values and vertical lines indicate two corresponding standard deviations. $R_p=80$.

Fig. 5-14 presents the change in velocity when a moving particle collides with another moving particle. The collision event is highlighted with a dotted circle denoting how each component of the particle velocity changes after the event. To the best of the writer's knowledge, there are no experimental/numerical data available to validate the change of velocity of a particle moving in bed load after a collision.

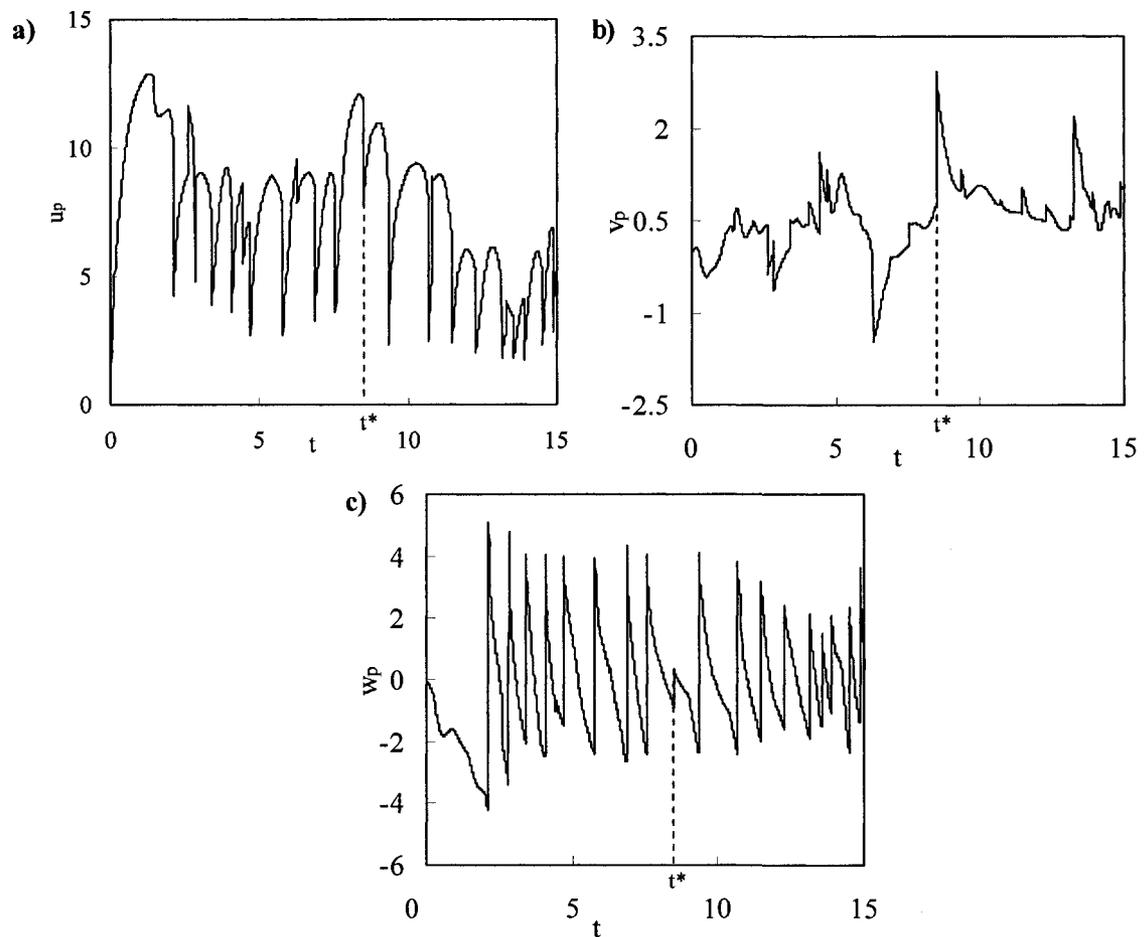


Figure 5-14: Time series of the velocity components of a single moving particle using a high resolution three dimensional velocity field, when multiple particles are simulated. t^* shows the time when a collision with other particle occurs. $R_p=73$, $\tau_p/\tau_c=1.5$. a) Stream-wise component. b) Span-wise component. c) Wall-normal component.

Visual observations of animations of the particle trajectories were used in this section to check the plausibility of the particle motion (see videos at http://mpfg.engr.ucdavis.edu/Student/Andrea/two_phase.html) and a realistic bed load motion was observed in those videos.

The frequency of inter-particle collisions is another parameter we can use to check the correctness of the model, and it is presented in Tables 5-1 and 5-2. As the flow velocity increases (Table 5-1), the number of collisions per particle decreases as a result of particles moving higher and therefore in a larger zone, thus decreasing the probability of encountering another particle (assuming the volume concentration is constant). An increase of concentration (Table 5-2) under the same flow conditions, produces an increase in the number of inter-particle collisions. In this case, a larger amount of particles moving in the same zone increases the probability of encounters with other particles.

Table 5-1: Frequency of inter-particle collisions. $R_p = 120$, $C = 0.1\%$.

τ_* / τ_{*c}	1	1.5	2	2.5	3
Number of inter-particle collisions per particle	6123	1487	1115	1125	1129
Frequency of inter-particle collision	20.4	4.9	3.7	3.8	3.8

Table 5-2: Frequency of inter-particle collisions. $R_p = 120$, $\tau_* / \tau_{*c} = 2.5$.

Concentration	0.05%	0.1%	0.2%	0.3%
Number of inter-particle collisions per particle	509	1275	1870	3113
Frequency of inter-particle collision	2.5	4.3	9.4	13.0

5.4.2 Effect of Inter-Particle Collisions on the Particle Turbulence Intensity and Kinetic Energy

The effect of inter-particle collision on the particle turbulence intensity and kinetic energy is studied using two numerical simulations, with and without particle collisions.

An equal number of particles were considered in both simulations, equivalent to a volume concentration of 0.05%, with an equivalent size of $R_p=73$ and a flow rate equivalent to $\tau_s/\tau_{*c}=1.5$. The simulations were run for a sufficiently long period of time, and the initial jumps were not considered in the statistical analysis.

The particle intensity in each direction and the particle turbulent kinetic energy were calculated for both cases, and the results are presented in Fig. 5-15. As expected, the particle turbulence intensity and turbulent kinetic energy are slightly larger in the case in which inter-particle collisions are included, indicating that collisions between particles provide a new source of energy and momentum independent of the particle-turbulence interactions. In spite of the difference in magnitude of the particle turbulence intensity in each component, the maximum value is located roughly in the same position for both simulations (approximately at two particle diameters from the bed), which lies slightly below the maximum value for the flow turbulence intensity.

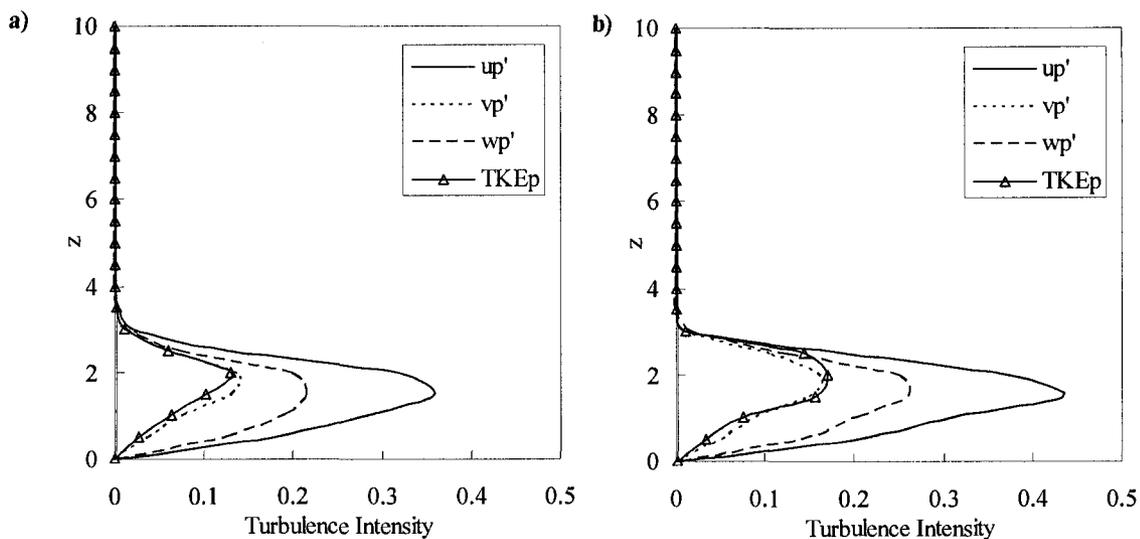


Figure 5-15: Multiple particle simulation with no inter-particle collision. Comparison between flow and particle turbulence intensity and turbulent kinetic energy and their variation in the vertical. $R_p=73$, $\tau_p/\tau_{*c}=1.5$. a) No inter-particle collision. b) Including inter-particle collision.

5.4.3 Effect of Particle Size and Flow Velocity on the Particle Turbulence intensity and Kinetic Energy

Simulations for two different particle sizes ($R_p=80$ and 120) under different flow conditions ($\tau_p/\tau_{*c}=1$ to 3) were run to evaluate the effect of the flow velocity and the particle size in the TKE_p . The results are presented in Fig. 5-16 and 5-17.

In both figures, the vertical profile of the particle turbulence intensity and turbulent kinetic energy present a maximum value around two particle diameters. For both particle sizes analyzed, increased flow velocity produces an increase of area under the curve of the particle turbulence intensity and turbulent kinetic energy. This effect can be explained as follows: as the flow velocity increases the mean, particles describe longer and higher jumps (as shown in Fig.5-11) allowing particles to move to higher elevations, increasing the vertical zone of influence of the turbulent parameters.

It is also observed, as the flow velocity is increased, the magnitude of the maximum value of the particle turbulence intensity in every direction and the TKE_p increase. As the flow velocity becomes larger, the flow turbulent energy becomes larger and the velocity fluctuation of the fluid becomes more important relatively to the particle inertia, providing the particle more energy, which is expressed as TKE_p .

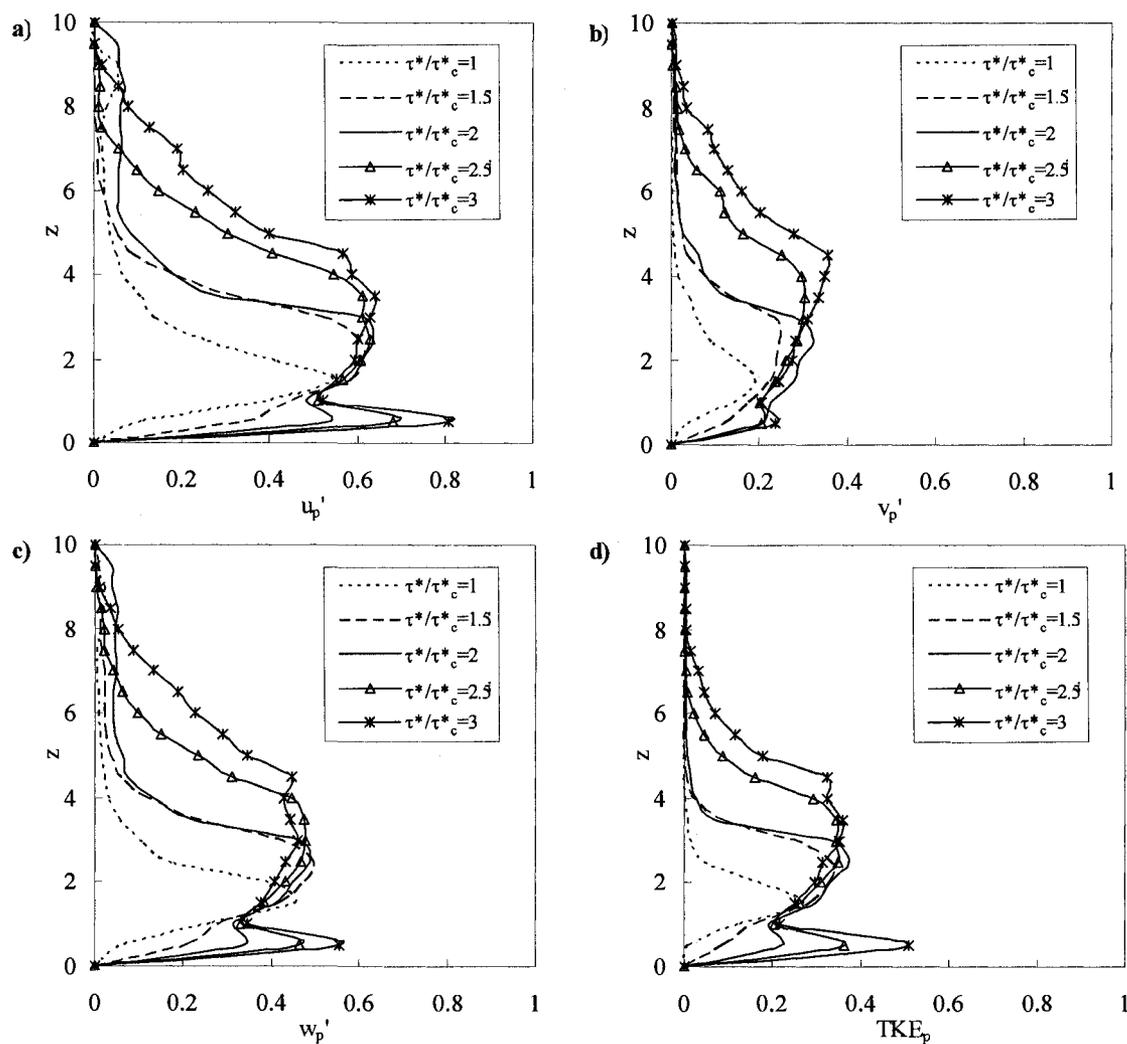


Figure 5-16: Particle turbulence intensity and particle turbulent kinetic energy vertical profile for multiple particle moving in a simulated velocity field. $R_p=80$. a) Stream-wise component of the turbulence intensity. b) Span-wise component of the turbulence intensity. c) Wall-normal component of the turbulence intensity. d) Particle turbulent kinetic energy.

The opposite effect is observed when the particle size increases: as the particle becomes larger, particle inertia becomes more important than the turbulent effects and therefore, the value of the particle turbulence intensity and turbulent kinetic energy tends to decrease.

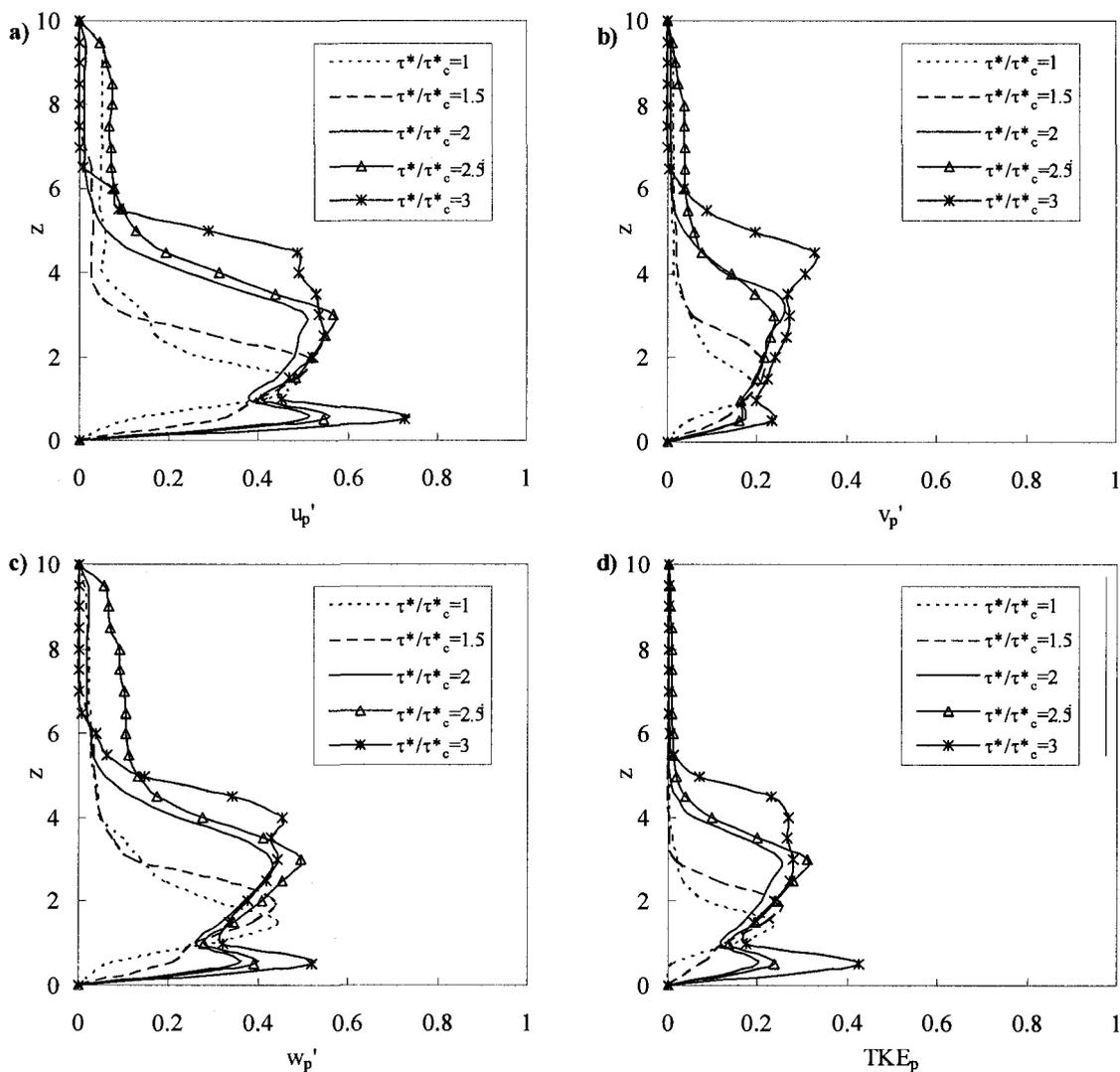


Figure 5-17: Particle turbulence intensity and particle turbulent kinetic energy vertical profile for multiple particle moving in a simulated velocity field. $R_p = 120$. a) Stream-wise component of the turbulence intensity. b) Span-wise component of the turbulence intensity. c) Wall-normal component of the turbulence intensity. d) Particle turbulent kinetic energy.

5.4.4 Effect of Volumetric Concentration on the Particle Intensity and Turbulent Kinetic Energy

The effect of the volumetric concentration of particles (C_v) on the TKE_p is presented in Fig. 5-18, considering the results of the simulation of multiple particles of size $R_p=120$, with a flow condition given by $\tau_p/\tau_{*c}=2$ for a concentration range varying between 0.05% to 0.3%. An increase of C_v increases the particle turbulence intensity and TKE_p . As the particle volumetric concentration increases, the frequency of particle-particle collision events is higher.

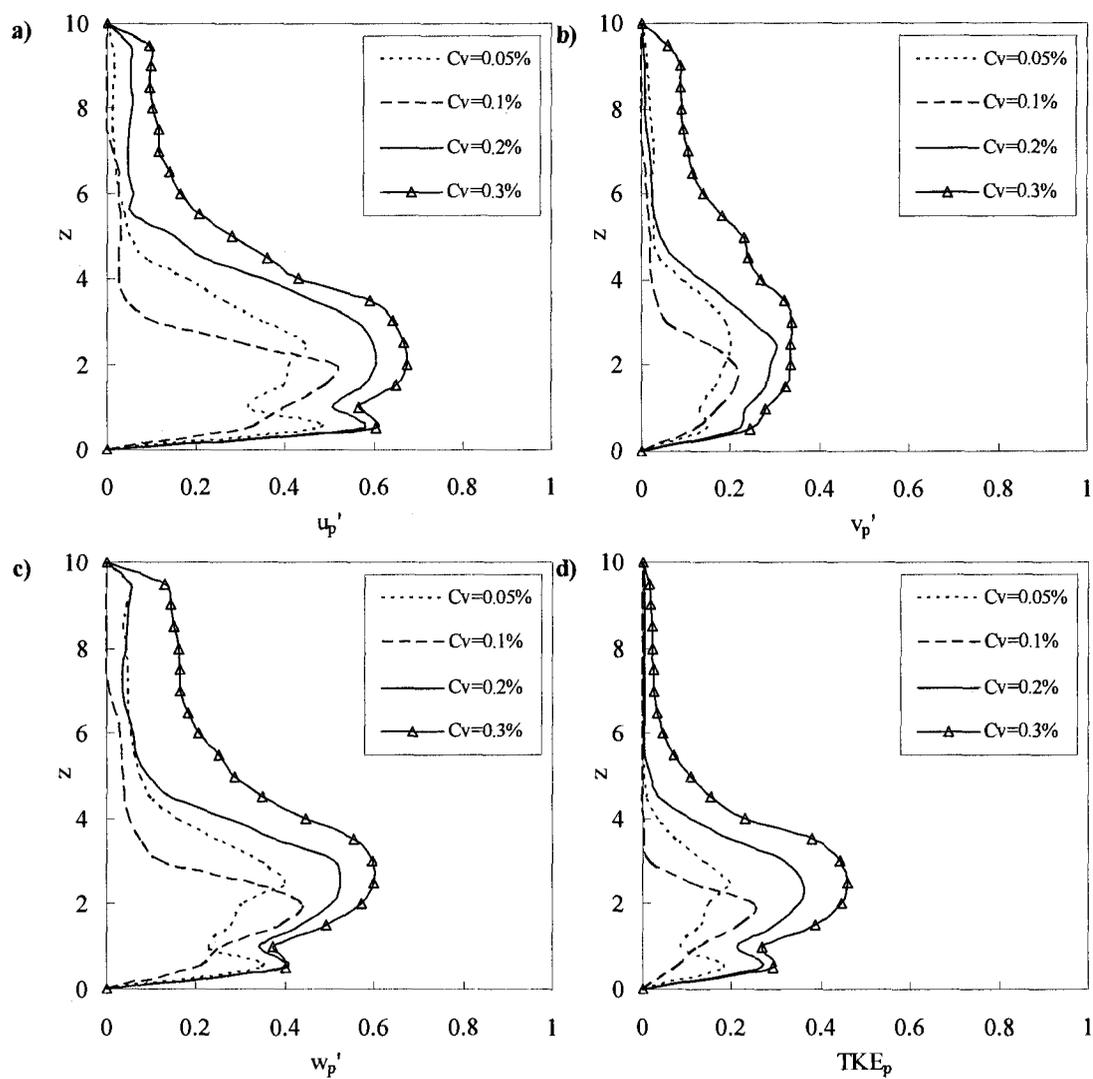


Figure 5-18: Particle turbulence intensity and particle turbulent kinetic energy vertical profile for multiple particle moving in a simulated velocity field. $R_p = 120$, $\tau_p / \tau_c = 2.5$. Concentration variation. a) Stream-wise component of the turbulence intensity. b) Span-wise component of the turbulence intensity. c) Wall-normal component of the turbulence intensity. d) Particle turbulent kinetic energy.

CHAPTER 6

SUMMARY AND CONCLUSIONS

The motion of sediment particles close to river beds is notably complex. Several experimental and numerical works have provided insight into the nature of the saltating motion in the last decades, concerning the magnitude of the jump length and heights, and the total mass put into motion (Niño et al., 1994; Niño and García, 1998a, b; Lee and Hsu, 1994, 1996). Researchers have addressed the conditions for initiation and cessation of saltation of the particles when they hit the bed, and have made tremendous progress in our understanding of the bed load phenomena. However, there are still significant questions without a definitive answer.

This thesis endeavored to answer some of those questions, making important assumptions required to make such complicated phenomena tractable. Obviously, some of the results obtained in this research should be interpreted in light of those assumptions.

Bed load transport in rivers was numerically simulated by combining a Lagrangian model describing the particle motion away from the bed (i.e., a flight model), a rebound model linking the particle velocity before and after the particle collision with the river bed, and a realistic bed roughness (surface) representation. These three components were analyzed in this thesis in order to provide a robust and efficient simulation tool for particles moving in saltation close to river beds. The model considers a Lagrangian-Eulerian approach, and it addresses multi-particle scenarios, including the effect of inter-particle collisions.

A new three-dimensional sub-model for particle free flight was developed and validated. By using available experimental data, it confirmed the necessity of including force terms that have been disregarded in previous simulations, such as the Basset force. The inclusion of all known forces in all directions is unprecedented for simulations of the motion of natural particles close to river beds.

It was corroborated that the Basset force must be included in Lagrangian models of bed load transport for particle Reynolds numbers smaller than about 4,000. This means that the force needs to be considered for sand particles, but can be disregarded in gravel particles. The use of a semi-derivative approximation to calculate the Basset integral was found to reduce the computational cost by 20% as compared with alternative techniques usually employed to compute improper integrals. This thesis introduces a novel concept called the memory time period (T_{back}), i.e., the interval of time during which the history of the particle affects the current particle motion. The memory time period was estimated by using a fixed number of past jumps of the saltating particle and it was also associated with more physical time scales. It was concluded that the minimum number of past jumps to be considered in the Basset force term varies between 25 and 50, depending on flow conditions. The larger the value of τ_* (or u_*), the smaller the number of jumps required to approximate the Basset integral accurately.

The approximation of the Basset force using the semi-derivative approach and the T_{back} concept reduces the computational time of the computations to about 70 to 90 percent of the original simulation time (time reductions of 30 to 10%). These time reductions are significant in large simulations with multiple particles.

The free flight sub-model was paired with the rebound sub-models proposed by Tsuji et al. (1985) and Niño and García (1994) (in two-dimensions). The beauty of the algorithm proposed by Tsuji et al. relies in its simplicity, inclusion of the particle angular and translation velocity after the collision, and the virtue of being easily extended to collisions between particles.

Three surface roughness sub-models were proposed (named sub-model *UDA*, *IBA* and *DBA*, respectively). These sub-models provided simple approaches to simulate the random trajectory pattern that is observed in particles moving in bed load motion. All the sub-models assumed that the bed is composed by an array of uniformly-packed spheres and that the point of contact between the flying particle and the bed is the random parameter to simulate. The point of contact defines an inclination plane, which is specified by using two angles, θ_b and α_b . The *UDA* sub-model assumes that both angles θ_b and α_b are uniformly distributed between -30° and 30° , and that these angles are determined through random number generators. The *IBA* and *DBA* sub-models are based on a geometrical relationship which connects θ_b , α_b , the incidence angles θ_m and α_m and two random variables r_1 and r_2 . In the *IBA* sub-model the values of r_1/d_p and r_2/d_p are obtained as random numbers uniformly distributed between 0 and 0.5, independently of the incidence angles. In the *DBA* sub-model, the values of r_1/d_p and r_2/d_p are obtained from a reduced range of values, assuming that they are uniformly distributed between a minimum and a maximum value depending on the incident angle. By constraining the range of variation of the random parameters, the range of variability

of the angles θ_b and α_b is reduced, obtaining only physically feasible points of contact between the flying particle and the bed.

The proposed sub-models were compared with three existing bed roughness approximations. Four sets of friction and restitution coefficients were selected from the literature to evaluate the sensitivity of those values in the simulations. The results obtained showed that the *DBA* sub-model reproduces the experimental results with very good agreement for a large range of particle sizes. They also show the great variability in the results due to changes in the values of the friction and restitution coefficients for the same model.

It is concluded that the representation of the bed roughness is crucial to model the bed load transport accurately. Values of the friction and restitution coefficients are also very important in defining the jump patterns. It is surprising that very little is discussed in the literature in that regard.

A highly resolved three dimensional (HR3D) turbulent velocity field was coupled with the particle tracking code to study the effect of the flow velocity fluctuations on the particle characteristics. The velocity field was obtained by simulating flow through a flat plate. In spite of the intrinsic differences between the boundary layer on flat plates and in open channels (as pointing out by Nieuwstadt and Bradshaw (1997) and Ashrafian (2004)) the semi-logarithmic expression accurately described the mean stream-wise velocity for both flows. In this work it was shown that the HR3D turbulent flow provided a satisfactory velocity field for the purpose of the study.

The one-way coupled code was validated through comparison with experimental data. The particle turbulent intensity in each direction (stream-wise, span-wise, and wall-

normal) was defined as a measure of the particle velocity fluctuation due to the turbulence with respect to the mean value, and the particle turbulent kinetic energy, was defined as the sum of the particle intensity in all three components; these variables were calculated to characterize the particle response to the flow variability. The detailed study of these parameters for particles moving in bed load motion has not been considered in previous studies and it constitutes one of the main contributions of this dissertation. In order to formally define the mean value for the particle velocity, a new filter was proposed. The discontinuity pattern observed in the particle velocity due to the effect of wall-particle collision defines the basic characteristics of this filter, which constitutes another contribution of this work. The variations of particle intensity and turbulent kinetic energy in the vertical direction were analyzed, showing that the zone of influence of both variables is located within a distance of around a few particle diameters from the bottom of the channel.

The effect of inter-particle collisions was also analyzed: two simulations were developed with the same flow conditions, with and without considering collisions. The results showed that collisions increased particle intensity and the turbulent kinetic energy, which indicates that collisions provide an additional source of turbulent energy to the saltating particle, as expected.

The effects of the particle size, flow velocity and particle concentration on the particle turbulent intensity and turbulent kinetic energy were also studied. Increasing the particle size diminishes the particle turbulence intensity and the turbulent kinetic energy, as the particle inertia becomes bigger, and the particle is less influenced by the turbulent fluctuations of the surrounding flow. Increasing the flow velocity produces an expansion

of the zone of influence of the particle intensity and turbulent kinetic energy close to the wall, as the particles are capable of moving to higher elevations. A slight increase of the value of both jump height and length is also obtained as the flow velocity is increased, since the particle moves to zones where the flow intensity and turbulent kinetic energy are bigger. As the particle concentration increases, the values of turbulent intensity and turbulent kinetic energy increase, due to the increase of the number of events of inter-particle collision. All this highlights the importance of including the detailed simulation of the turbulence in bed load, not only because the total transport of sediments changes but also because this sheds light in the interaction of particles with the flow.

The particle tracking model developed in this work presents a level of detail without precedent. However, all models constitute approximations to the problem under analysis and are, therefore, imperfect. The aim of this work was to represent the key physical processes of bed load transport as close as possible so that the main characteristics of the system of interest were replicated. In this case, the main features of the bed load transport (particle jump height, length, and mean stream-wise and angular velocity) were simulated with good accuracy. In the most realistic scenario simulated, i.e., multiple particles in a highly resolved turbulent velocity field, the prediction of the volumetric bed load rate showed a very good agreement with highly-used expressions.

Summarizing, the model presented in this work provides the following contributions:

- All the forces acting on a particle are considered.
- The computation of the Basset force was optimized.
- A new 3-D bed roughness representation was introduced.

- Inter-particle collisions was considered.
- A new filter to calculate the mean particle velocity for the turbulent flow in bed load transport was presented. With this filter, the particle turbulent intensity and kinetic energy were defined and calculated.
- The effect of particle size, flow condition and particle concentration in the particle turbulent parameters was addressed.

Future work will involve computational activities to learn more about the accuracy of the models developed here. In the first area, the particle tracking code was developed in such a way that two-way coupling can be easily incorporated. By simulating a turbulent open-channel flow with a two-way coupling, the effect of particles on the fluid flow turbulence can be studied. Conclusions related to the exchange of energy and momentum between both phases (fluid and particle) can be achieved by comparing the results obtained in this thesis with simulations of two-way coupling. The influence of the particle size, volumetric concentration, flow rate, and inter-particle collisions can be studied in a two-way coupling scenario to elucidate how those parameters modify the particle turbulent characteristics. This will lead to a more complete understanding of the particle-laden phenomenon in rivers.

REFERENCES

- Abramowitz, M. and Stegun, I. (1970). *Handbook of mathematical functions*. Dover Publications Inc., New York.
- Ahmadi, G. and Ma, A. (1990). "A thermodynamical formulation for dispersed multi phase turbulent flow. I. Basic Theory." *Int. J. Multiphase Flow*. 16. 323-340.
- Arbulieva, K. M., Olevinskaya, S. K. and Pivovarov, A. A. (1987). "Some findings of experimental studies conducted on the intermittent movement of gravel in a turbulent current". *Dynamic and thermal regimes of rivers*, Russian Translations Series 52. K. I. Rossinskii, ed.
- Armenio, V. and Fiorotto, V. (2001). "The importance of the forces acting on particles in turbulent flows." *Phys. Fluids*. 13 (8). 2437-2440.
- Ashrafian, A. (2004). "Numerical investigation of turbulent flow in a channel with rough walls". *PhD Thesis*. Norwegian University of Science and Technology. Trondheim, Norway.
- Bagchi, P. and Balachandar, S. (2003). "Effect of turbulence on the drag and lift of a particle". *Phys. Fluids*. 15 (11). 3496-3513.
- Best, J. (1992). "On the entrainment of sediment and initiation of bed defects: insights from recent developments within turbulent boundary layer research". *Sedimentology*. 39. 797-811.
- Boivin, M., Simonin, O. and Squires, K. (1998). "Direct numerical simulation of turbulence modulation by particles in isotropic turbulence". *J. Fluid Mech*. 375. 235-263.

- Bombardelli, F. A., González, A. E. and Niño, Y. I. In press. "Non-linear, Lagrangian theoretical models for particle motion close to solid boundaries." *ASCE J. of Hyd. Eng.*
- Bonzogo, J., Heim K., Warwick J. and Lyons, W. (1996). "Mercury levels in surface waters of the Carson River Lahontan Reservoir system, Nevada: Influence of historic mining activities". *Env. Pollution*. 92 (2). 193-201.
- Brush, J. S., Ho, H. W. and Yen, B. C. (1964). "Acceleration motion of sphere in a viscous fluid." *ASCE J. Hydr. Div.* 90. 149-160.
- Calo, V.M. (2004) "Residual-based multiscale turbulence modeling: finite volume simulations of bypass transition." *PhD. Dissertation*. Stanford University.
- Chen, X. Q. and Pereira, J.C.F. (2000). "Computational modeling of a dilute turbulent liquid-solid flow using a Eulerian-Lagrangian approach." *Int. J. of Numerical Methods for Heat & Fluid Flow*. 10(4). 409-431.
- Crowe C., Sommerfeld, M. and Tsuji, M. (1998). *Multiphase flow with droplets and particles*. CRC Press. Boca Raton FL.
- Crowe C.(2000). "On models for turbulence modulation in fluid-particle flows". *Int. J. Multiphase Flow*. 26 (5). 719-727.
- Deen, D.G., Van Sint Annaland, M., Van der Hoef, M.A. and Kuipers, J.A.M. (2007). "Review of discrete particle modeling of fluidized beds". *Chem. Eng. Society*. 62. 28-44.
- Drew, D.A. and Lahey, R.T. (1987). "The virtual mass and the lift force in a sphere in rotating and straining inviscid flow". *Int. J. of Multiphase Flow*. 13 (1). 113-121.

- Dennis, S., Singh, S. and Ingham, D. (1980). "The steady flow due to a rotating sphere at low and moderate Reynolds number". *J. Fluid Mech.* 101. 257-279.
- Dorgan, A. and Loth, E. (2004). "Simulation of particles released near the wall in a turbulent boundary layer." *Int. J. Multiphase Flow.* 30. 649-673.
- Dorgan, A. and Loth, E. (2007). "Efficient calculation of the history force at finite Reynolds number." *Int. J. Multiphase Flow.* 33. 833-848.
- Drew, D. (1976). "Production and dissipation of energy in the turbulent flow of a particle-fluid mixture, with some results on drag reduction." *J. Appl. Mech.* 43. 543-547.
- Duan, J., Barkdoll, B., Fan, S., Klumpp, C., McAnnally, B., Papanicolaou, T., Scott, S., Wang, S. , Wu, W. and Ying, X. (2004). "Computational modeling of sediment transport processes." *ASCE J. Hyd. Eng.* 130(7). 3-5.
- Eaton, J. and Fessler, J. (1994). "Preferential concentration of particles by turbulence." *Int. J. Multiphase Flow.* 20. 169-209.
- Einstein, H.A. (1950). "The bedload function for sediment transportation in open channel flow". US Department of Agriculture Technical Bulletin N° 1026, Washington D.C.
- Elgobashi, S.E. and Abou-Arab, T.W. (1983). "A two-equation turbulence closure for two-phase flows". *Phys. Fluids.* 26. 931–938.
- Elghobashi, S. and Truesdell, G. (1993). "On the 2-way interaction between turbulence and dispersed solid particles. 1. Turbulence modification". *Phys. Fluids A-Fluid Dynamics.* 5 (7). 1790-1801.
- Engelund, F. and Hansen, E. (1967). "A monograph on sediment transport in alluvial streams ". *Monograph Denmark Technical University. Hydraulic Lab. Denmark.*

- Fernandez-Luque, R. and van Beek, R. (1976). "Erosion and transport of bed sediments". *J. of Hyd. Res.* 14(2). 127-144.
- Ferrante, A. and Elgobashi, S. (2003). "On the physical mechanisms of two-way coupling in particle-laden isotropic turbulence". *Phys. Fluids* 15 (2), 315-329.
- Gad-El-Hak, M. (2000). *Flow Control*. Cambridge University Press.
- García, M. and Niño, Y. (1992). "Lagrangian description of bedload transport by saltating particles." *Proc. of the Sixth IAHR Int. Symposium on Stochastic Hydr.* Taipei, Taiwan.
- García, M. (1999). "Sedimentation and erosion hydraulics." Ch. 6 in *Hydraulic design handbook*, Larry W. Mays, Ed., McGraw-Hill.
- Gardner, K. and Theis, T. (1996). "A unified kinetic model for particle aggregation." *J. of Colloid and Interface Sci.* 180(1). 162-173.
- Gonzalez, A., Bombardelli, F. and Niño, Y. (2006). "Improving the prediction capability of numerical models for particle motion in water bodies". *Proceedings of the 7th Int. Conf. on Hydroscience and Engineering (ICHE-2006)*, Philadelphia. PA.
- Gordon, R., Carmichael, J.B. and Isackson, F.J. (1972). "Saltation of plastic balls in a 'one-dimensional' flume". *Water Res. Research.* 8(2). 444-459.
- Gore R. and Crowe C. (1989). "Effect of particle-size on modulating turbulence intensity". *Int. J. Multiphase Flow.* 15. 279-285.
- Gotoh, H. and Sakai, T. (1997). "Numerical simulation of sheetflow as granular material". *ASCE J.I of Waterway, Port, Coastal and Ocean Eng.* 123 (6). 329-336.
- Grass, A. J. (1970). "Initial instability of fine bed sand." *ASCE J. Hyd. Eng.* 96. 619-632.

- Greimann, B.P., Muste, M. and Holly, F. (1999). "Two phase formulation of suspended sediment transport." *J. Hyd. Res.* 37 (4). 479-500.
- Greimann, B. P. and Holly, F. (2001). "Two-phase flow analysis of concentration profile." *J. Hyd. Eng.* 127 (9).753-762.
- Groszmann, D.E. and Rogers, C.B. (2004). "Turbulent scales of dilute particle-laden flows in microgravity". *Phys. Fluids.* 16. 4671-4684.
- Gupta, P. K. and Pagalthivarthi, K.V. (2006). "A comparative study of the effect of model lift coefficients on particle trajectory". *Indian J. of Eng. and Material Sci.* 13. 293-306.
- Hanratty, T.J., Theofanous, T., Delhaye, J., Eaton, J., McLaughlin, J., Prosperetti, A., Sundaresan, S., and Tryggvason, G. (2003). " Workshop Findings". *Int. J. of Multiphase Flow.* 29. 1047-1059
- Harada, E. and Gotoh, H. (2006). "Influence of sand shape to vertical sorting under uniform flow condition". *Proceedings of River Flow 2006.* Lisbon, Portugal.
- Heald, J., McEwan, I. and Tait, S. (2004). "Sediment transport over a flat bed in a unidirectional flow: simulations and validation". *Phil. Transactions of the Royal Soc. London A.* 362. 1973-1986.
- Heathershaw, A. and Thorne, P. (1985). "Sea-bed noises reveal role of turbulent bursting phenomenon in sediment transport by tidal currents". *Nature.* 316. 339-342.
- Hu, C. and Hui, Y. (1996). "Bed-Load transport. II. Stochastics characteristics." *ASCE J. Hyd. Eng.* 122. 255-261.
- Hsu, T.J., Jenkins, J. T. and Liu, P. L. (2003). "On two-phase sediment transport: Dilute flow." *J. Geophysical Res.* 108 (C3). 3057.

- Jacobs, R. and Durbin, P. (2001). "Simulation of bypass transition." *J. Fluid Mech.* 428. 185-212.
- Jiang, J., Law, A. and Cheng, N. (2004). "Two-phase modeling of suspended sediment distribution in open channel flows". *J. Hyd. Res.* 42 (3). 273-281.
- Jimenez, J. (2004). "Turbulent flows over rough walls". *Annual Rev. of Fluid Mech.* 36. 173–196.
- Johnson, R.W. (1998). *The handbook of fluid dynamics*. Boca Raton FA. CRC Press.
- Julien, P. (1998). *Erosion and sedimentation*. Cambridge University Press.
- Kaftori, D., Hetsroni, G. and Banerjee, S. (1995). "Particle behavior in the turbulent boundary-layer. 1. Motion, deposition and entrainment". *Phys. Fluids.* 7. 1095-1106.
- Karamanev, D.G. (2001). "The study of free rise of buoyant spheres in gas reveals the universal behaviour of free rising rigid spheres in fluid in general". *Int. J. of Multiphase Flow.* 27. 1479-1489.
- Kartushinsky, A. and Michaelides, E. (2004). "An analytical approach for the closure equations of gas-solid flows with inter-particle collision." *Int. J. Multiphase Flow.* 30. 159-180.
- Kendoush, A. A. (2005). "The virtual mass of a rotating sphere in fluids". *J. of Applied Mech.* 72. 801-802.
- Kenning V. and Crowe C. (1997). "On the effect of particles on carrier phase turbulence in gas-particle flows". *Int. J. Multiphase Flow.* 23. 403-408.
- Keylock, C., Hardy, R., Parsons, D., Ferguson, R., Lane, S. and Richards, L. (2005). "The theoretical foundations and potential for large-eddy simulation (LES) in fluvial geomorphic and sedimentological research". *Earth Sci. Reviews* 71, 271-304.

- Kundu, P.K. and Cohen, I.M. (2004). *Fluid Mechanics*. Elsevier Academic Press.
- Lee, S.Y. (1993). "Investigations of the particle saltating process near the channel bed".
Unpublished *MS Thesis*, National Taiwan University, Taiwan.
- Lee, H. Y. and Hsu, I. S. (1994). "Investigation of saltating particle motions". *ASCE J. of Hyd. Eng.* 120 (7). 831-845.
- Lee, H.Y. and Hsu, I.S. (1996). "Particle spinning motion during saltating process".
ASCE J. of Hyd. Eng. 122(10). 587-590.
- Lee, H., Chen Y., You, J. and Lin, Y. (2000). "Investigations of continuous bed load saltating process." *ASCE J. of Hyd. Eng.* 126. 691-700.
- Lee, H.Y., Lin, Y.T., You, J.Y. and Wang, H.W. (2006). "On three-dimensional continuous saltating process of sediment particles near the channel bed". *J. of Hyd. Research.* 44 (3). 374-389.
- Loth, E. (2005). *Simulating the dynamics of dispersed multi-phase flow*. In press.
- Lun, C. and Liu, H. (1997). "Numerical simulation of dilute turbulent gas-solid flows in horizontal channels." *Int. J. Multiphase Flow.* 23. 575-605.
- Lukerchenko, N., Chara, Z. and Vlasak, P. (2006). "2D Numerical model of particle-bed collision in fluid-particle flows over bed". *J. of Hyd. Research.* 44 (1). 70-78.
- Massoudieh, A., Bombardelli, F. and Ginn, T. (2006). " Mathematical modeling of fate and transport of dissolved and particulate mercury in riverine systems". *Proceedings of the World Environmental and Water Resources Congress*. Omaha, Nebraska, USA.
- Massoudieh, A., Ju, D., Young, T. and Ginn, T. (2008). " Approximation of a radial diffusion model with a multiple-rate model for hetero-disperse particle mixtures". *J. of Contaminant Hydrology.* 97, 55-66 .

- Matsumoto, S. and Saito, S. (1970a). "On the mechanism of suspension of particles in horizontal pneumatic conveying: Monte Carlo simulation based on the irregular bouncing model." *J. of Chem. Eng. of Japan* . 3. 83-92.
- Matsumoto, S. and Saito, S. (1970b). "Monte Carlo simulation of horizontal pneumatic conveying based on the rough wall model." *J. of Chem. Eng. of Japan*. 3. 223-230.
- Mazumder, R. (2000). "Turbulence-particle interactions and their implications for sediment transport and bedform mechanics under unidirectional current: some recent developments". *Earth-Science Reviews*. 50.113-124.
- Maxey, M. and Riley, J. (1983). "Equation of motion for a small rigid sphere in a nonuniform flow". *Phys. Fluids*. 26 (4): 883-889.
- Mei, R., Adrian, J. and Hanratty, T. (1991). "Particle dispersion in isotropic turbulence under Stokes drag and Basset force with gravitational settling". *J. Fluid Mech.* 225. 481-495.
- Michaelides, E. E. (1992). "A novel way of computing the Basset term in unsteady multiphase flow computations." *Phys. Fluids*, 4(7), 1579-1582.
- Mordant, N. and Pinton, J. F. (2000). "Velocity measurement of a settling sphere." *The European Physical Journal B*, 18, 343-352.
- Nelson, J., Shreve, R., McLean, S. and Drake, T. (1995). "Role of near-bed turbulence in bed load transport and bed load mechanics". *Water Res. Research*. 31. 2031-2086.
- Nezu, I. and Nakagawa, H. (1993). *Turbulence in Open-Channel Flows*. IAHR Monograph Series. Rotterdam. A.A. Balkema.
- Nishimoto, K. (1991). *Fractional calculus*. Descartes Press Co., Koriyama, Japan.

- Niño, Y., García, M. and Ayala, L. (1994). "Gravel saltation. 1. Experiments." *Water Res. Research.* 30. 1907-1914.
- Niño, Y. and García, M. (1994). "Gravel saltation. 2. Modelling." *Water Res. Research.* 30. 1915-1924.
- Niño, Y. and García, M. (1996). "Experiments on particle-turbulence interactions in the near-wall region of an open channel flow: implications for sediment transport." *J. Fluid Mech.* 326. 285-319.
- Niño, Y. and García, M. (1998a). "Experiments on saltation of sand in water." *ASCE J. Hyd. Eng.* 124. 1014-1025.
- Niño, Y. and García, M. (1998b). "Using Lagrangian particle saltation observations for bedload sediment transport modeling." *Hydrological Processes.* 12. 1197-1218.
- Nieuwstadt, F. T. M. and Bradshaw, P. (1997). "Similarities and differences of turbulent boundary-layer, pipe and channel flow". In *Boundary -Layer separation in aircraft aerodynamics*. Ed. by Henkes, R.A.W.M., Bakker, P.G. Delft University Press.
- Oldham, Y. and Spanier, M. (1974). *The fractional calculus*. New York: Academic Press.
- Olsen, N. (2003). "Three-dimensional CFD modeling of self-forming meandering channel". *ASCE J. Hyd. Eng.* 129 (5). 366-372 .
- Oseen, C. W. (1927). *Hydrodynamik*. Leipzig : Akademische Verlagsgesellschaft.
- Panton, R. L.(1996). *Incompressible Flow*. Wiley and Sons. New York
- Pierce, C. and Moin, P. (2001). "Progress-variable approach for large eddy simulation of turbulent combustion". Technical Report TF-80, Flow Physics and Computation Division, Department of Mechanical Engineering, Stanford University. Available at <http://ctr.stanford.edu/Pierce/thesis.pdf>

- Portela, L. and Oliemans, R. (2003). "Eulerian-Lagrangian DNS-LES of particle-turbulence interactions in wall bounded flows." *Int. J. for Numerical Methods in Fluids*. 43. 1045-1065.
- Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. (1992). *Numerical Recipes in Fortran 77*. Cambridge University Press.
- Redman, J., Grant, S., Olson, T., Hardy M. and Estes M. (1997). "Filtration of recombinant Norwalk virus particles and bacteriophage MS2 in quartz sand: Importance of electrostatic interactions". *Env. Science & Technology*. 31 (12). 3378-3383.
- Roach, P.E. and Brierley, D.H. (1992). "The influence of a turbulent free-stream on a zero pressure gradient transitional boundary layer development. Part 1: Test case T3A and T3B." In *Numerical Simulation of Unsteady Flows and Transition to Turbulence, ERCOFTAC*. Pironneau, O., Rodi, W., Rhyming, I.L., Savill, A.M, Truong, T.V. editors, Cambridge University Press. 319-347.
- Rouse, H., Siao, T.T. and Nagaratnam, S. (1958). "Turbulence characteristics of the hydraulic jump". *J. of the Hyd. Div. ASCE*. Proc. Paper 1528.1-30.
- Rubey, W.W. (1933). "Settling velocities of gravel, sand and silt particles". *Am. J. Sci.* 25. 325-338.
- Rubinow, S.I. and Keller, J.B. (1961). "The transverse force on a spinning sphere moving in a viscous fluid". *J. Fluid Mech.* 11. 447-459.
- Sankaranarayanan, K., Shan, X., Kevrekidis, I. G. and Sundaresan, S. (2002). "Analysis of the drag and virtual mass forces in bubbly suspensions using an implicit formulation of the lattice Boltzmann method". *J. of Fluid Mech.* 452. 61-96.

- Schmeeckle, M., Nelson, J., Pittlick, J. and Bennett, J. (2001). "Interparticle collision of natural sediment grains in water." *Water Res. Research*. 37. 2377-2391.
- Schmeeckle, M. and Nelson, J. (2003). "Direct numerical simulation of bedload transport using local boundary condition." *Sedimentology*. 50. 279-301.
- Scotti, A. (2006). "Direct numerical simulation of turbulent channel flows with boundary roughened with virtual sand paper". *Phys. of Fluids*. 18. 031701.
- Sekine, M. and Kikkawa, H. (1992). "Mechanics of saltating grains II." *ASCE J. Hyd. Eng.* 118. 536-557.
- Shabani, B., Yeganeh, A. and Wang, S.S.Y. (2005). "Two-phase flow simulation of bedload transport in three dimension with MBS-3D model." *Proceedings of the XXXI IAHR Congress*. Seoul, South Korea.
- Shen, N., Tsuji, T. and Morikawa, Y. (1989). "Numerical simulation of gas-solid two-phase flow in horizontal pipe". *Japan Soc. of Mech. Eng.* 55. 2294.
- Shresta, P. and Orlob, G. (1996). "Multiphase distribution of cohesive sediments and heavy metals in estuarine systems." *ASCE J. Env. Eng.* 122(8). 730-740.
- Sommerfeld, M. (1992). "Modelling of particle-wall collisions in confined gas-particle flows." *Int. J. Multiphase Flow*. 18. 905-926.
- Sommerfeld, M. and Huber, N. (1999). "Experimental analysis and modeling of particle-wall collisions." *Int. J. Multiphase Flow*. 25. 1457-1489.
- Sommerfeld, M. (2003). "Analysis of collision effects for turbulent gas-particle flow in horizontal channel: Part I: Particle transport." *Int. J. Multiphase Flow*. 29. 675-699.

- Sommerfeld, M. and Kussin, J. (2003). "Analysis of collision effects for turbulent gas-particle flow in horizontal channel: Part II: Integral property and validation." *Int. J. Multiphase Flow*. 29. 701-718.
- Squires, K. and Eaton, J. (1990). "Particle response and turbulence modification in isotropic turbulence". *Phys. of Fluids A-Fluid Dynamics*. 2 (7). 1191-1203.
- Squires, K. and Eaton, J. (1994). "Effect of selective modification of turbulence on two-equation models for particle-laden turbulent flows". *ASME J. of Fluids Eng.* 116 (4). 778-784.
- Storti, M. and D'Elia, J. (2001). "A semi-numerical computation for the added mass coefficient of an oscillating hemi-sphere at very low and very high frequencies". arXiv:math.NA/0110302 v1.
- Sundaram, S. and Collins, L. (1999). "A numerical study of the modulation of isotropic turbulence by suspended particles". *J. Fluid Mech.* 379: 105-143.
- Sutsepın, V.A. (1987). "Vertical distribution of bedload material with the current flowing over a rough bed is stable and weak". *Dynamic and thermal regimes of rivers*, Russian Translations Series 52. K. I. Rossinskii, ed.
- Takagi, H. (1977). "Viscous flow induced by slow rotation of a sphere." *J. Phys. Soc. Japan*. 42. 319-325.
- Tanaka, T. and Tsuji, Y. (1991). "Numerical simulation of gas-solid two-phase flow in a vertical pipe: on the effect of interparticle collision". In: Stock, D., Tsuji, Y., Jurewicz, J., Reeks, M., Gautam, M. (Eds). *Gas-Solid Flows*. In: FED- 121. ASME. 123-128.

- Taniere, A. Khalij, M. and Oesterle. B. (2004). "Focus on the disperse phase boundary conditions at the wall for irregular particle bouncing." *Int. J. Multiphase Flow*. 30. 327-345.
- Tatom, F. B. (1988). "The Basset term as a semiderivative." *Applied Scientific Research*. 45. 283-285.
- Ten Cate, A., Dersksen, J., Portela, L. and Van Den Akker, H. (2004). "Fully resolved simulations of colliding monodisperse spheres in forced isotropic turbulence." *J. Fluid Mech*. 519. 233-271.
- Tsuji, Y., Oshima, T. and Morikawa, Y. (1985). "Numerical simulation of pneumatic conveying in a horizontal pipe". *KONA* 3. 38-51.
- Tsuji, Y, Morikawa, T., Tanaka, T., Nakatsukasa, N. and Nakatani, M. (1987). "Numerical simulation of gas-solid two-phase flow in a two-dimensional horizontal channel". *Int. J. of Multiphase Flow*. 13 (5). 671-684.
- Tsujimoto, T. and Nakagawa, H. (1983). "Stochastic study on successive saltation by flowing water." *Proc. 2nd Int. Symp. On River Sedimentation*. Beijing, China.
- Turton, R. and Levenspiel, O. (1986). "A short note on the drag correlation for spheres". *Powder Technology*. 47. 83-86.
- Van Haarlem, B., Boersma, B. and Nieuwstadt, F. (1998). "Direct simulation of particle deposition onto a free-slip and non-slip surface." *Phys. Fluids*. 10. 2608-2620.
- Vojir, D.J. and Michaelides, E.E. (1994). "Effect of the history term on the motion of rigid spheres in a viscous fluid". *Int. J. Multiphase Flow*. 20 (3). 547-556.

- Wang, H., Fan, W. and Yun, C. (2004). "Indicators and impact analysis of sediment from the Changjiang Estuary and East China Sea to the Hangzhou Bay". *Acta Oceanologica Sinica*. 23 (2). 297-308.
- Wei, T., Fife, P., Klewicki, J. and McMurtry, P. 2005. "Properties of the mean momentum balance in turbulent boundary layer, pipe and channel flows". *J. of Fluid Mech.* 552. 303-327.
- White, B.R. and Schulz, J.C. (1977). " Magnus effect in saltation". *J. Fluid Mechanics*. 81 (3). 497-512.
- White, F. M. (1974). *Viscous fluid flow*. McGraw-Hill Book Company, USA.
- Wiberg, P., and Smith, J. D. (1985). "A theoretical model for saltating grains in water." *J. Geophys. Res.*, 90(C4), 7341-7354.
- Wood, I. R. and Jenkins, B. S. (1973). "A numerical study of the suspension of a non-buoyant particle in a turbulent stream." *Proc., IAHR Int. Symp. on River Mech.* Bangkok. Thailand.
- Yamamoto, Y., Potthoff, M., Tanaka, T., Kajishima, T. and Tsuji, Y. (2001). "Large-eddy simulation of turbulent gas-particle flow in a vertical channel: effect of considering interparticle collisions." *J. Fluid Mech.* 442. 303-334.
- Yeganeh, A., Gotoh, H. and Sakai, T. (2000). "Applicability of Euler-Lagrange coupling multiphase-flow model to bed-load transport under high bottom shear". *J. of Hyd. Research*, 38 (5), 389-398.
- Yen, B.C. (1992). "Sediment fall velocity in oscillating flow". *Water resources Environmental Engineering Resources*. Report 11. Dept. of Civil Eng. University of Virginia.

- Yost, S. and Katopodes, N. (1996). "3D numerical modeling of cohesive sediments".
Estuarine and coastal modeling, M. L. Spaulding and R. T. Cheng, eds., ASCE, New York.
- Young, J. and Leemin, A. (1997). "A theory of particle deposition in turbulent pipe flow." *J. Fluid Mech.* 134. 129-159.
- Zedler, E. and Street R. (2001). "Large-eddy simulation of sediment transport: Currents over ripples". *ASCE J. Hyd. Eng.* 127 (6). 444-452.

APPENDIX A

COMPUTATIONAL CODE

```

PROGRAM SALTATE15
USE VELOCITY
OPEN(999,STATUS='UNKNOWN',FILE='MATRIX_INDEX.DAT')
JJ=0
TIME=0.0D0
MIN_TBEGIN=0.0D0
MIN_TBEGIN_O=0.0D0
NODE_NUMBER=0.0D0
CONC_COUNTER=0
ALLOCATE(X(NX))
ALLOCATE(Y(NY))
ALLOCATE(Z(NZ))
ALLOCATE(U(NX,NZ,NY))
ALLOCATE(V(NX,NZ,NY))
ALLOCATE(W(NX,NZ,NY))
ALLOCATE(U_F(NX,NZ,NY))
ALLOCATE(V_F(NX,NZ,NY))
ALLOCATE(W_F(NX,NZ,NY))
! ALLOCATE MEMORY FOR VELOCITY FIELD (U,V,W)
CALL LOAD_GRID !GRID INFORMATION IN ARRAY X,Y,Z
! VELOCITY INFORMATION IN 3D MATRIX U,V,W
CALL LOAD_VELOCITY(TIME,TIME_V,TIME_VP1)
IUNIT=100
CALL CREATE_FILES(IUNIT)
! INITIAL PARTICLE CONDITIONS
OPEN(UNIT=50,STATUS='UNKNOWN',FILE='INITIAL_POSITION.TXT')
OPEN(UNIT=60,STATUS='UNKNOWN',FILE='INITIAL_VELOCITY.TXT')
OPEN(UNIT=70,STATUS='UNKNOWN',FILE='INITIAL_ROTATION.TXT')
NULLIFY(BEGIN_LIST)
DO 100 II=1,N_PARTICLES
  READ(50,*) XPM_N(II),YPM_N(II),ZPM_N(II)
  READ(60,*) UPM_N(II),VPM_N(II),WPM_N(II)
  READ(70,*) RPX_N(II),RPY_N(II),RPZ_N(II)
  XPM_N(II)=XPM_N(II)+X_MIN !ABSOLUTE SYSTEM OF REFERENCE
  YPM_N(II)=YPM_N(II)+Y_MIN
  MATRIX_INDEX(II,1)=0
  MATRIX_INDEX(II,2)=0
  CALL UPDATE(II)
  TBEGIN(II)=0.0D0 !BEGGINING TIME FOR BASSET COMPUTATION
! DEFINITION OF STATISTICAL PARAMETERS
!
  JUMPL(II)=0.0D0
  JUMPH(II)=0.0D0
  JUMPINI(II)=XPM_N(II)
  N_REB(II)=0
100 END DO
  CALL ACCESORIES(TIME)
! PROGRAM CONDITIONS. TIME INITIAL CONDITIONS

```

```

!
EXT=0
TIME=DT
JJ=1
!
IDENTIFICATION OF RUN CHARACTERISTICS
!
WRITE (*,*) 'NBACK=', N_BACK
WRITE (*,*) 'TFIN=', TFIN
WRITE (*,*) 'RP=', RP
WRITE (*,*) 'TAU *=', TAUS
!
LOOP TIME-ADVANCE CALCULATION
1001 DO 500 WHILE (TIME.LT.TFIN.AND.EXT.EQ.0)
!
LOOP PARTICLE VELOCITIES
DO 550 II=1,N_PARTICLES
!
DEFINITION OF DIMENSIONLESS PARAMETERS PER PARTICLE
URA=URM(II)
CD=DRAG (URA, TAUS, RP)
ALFA=1.0D0 / (R+1.0D0+CM*VM)
BETA=-3.0D0/4.0D0*ALFA*CD*DR
GU=ALFA/TAUS*SIN(TITA)*SW
GW=ALFA/TAUS*COS(TITA)*SW
DELTA=3.0D0/4.0D0*ALFA*CL*LF
EPS=ALFA*CM*VM
KAPA=9.0D0*ALFA/(SQRT(RP*PI)*TAUS**0.25D0)*BS
XPM_N(II)=XPM_O(II)+UPM_O(II)*DT
IF (XPM_N(II).LT.X_MIN) XPM_N(II)=XPM_N(II)+X_MIN
IF (XPM_N(II).GT.X_MAX) XPM_N(II)=XPM_N(II)-X_MAX+X_MIN
YPM_N(II)=YPM_O(II)+VPM_O(II)*DT
IF (YPM_N(II).LT.Y_MIN) YPM_N(II)=YPM_N(II)+Y_MIN
IF (YPM_N(II).GT.Y_MAX) YPM_N(II)=YPM_N(II)-Y_MAX+Y_MIN
ZPM_N(II)=ZPM_O(II)+WPM_O(II)*DT
RPX_N(II)=RPX_O(II)+ARX(II)*DT
RPY_N(II)=RPY_O(II)+ARY(II)*DT
RPZ_N(II)=RPZ_O(II)+ARZ(II)*DT
!
II-TH PARTICLE NEW VELOCITY CALCULATION
!
CALCULATED USING EULER (METHOD=1) OR RUNGE KUTTA (METHOD=2)
UU(1)=UPM_O(II)
UU(2)=VPM_O(II)
UU(3)=WPM_O(II)
CALL DERIV(TIME-DT,UU,DUDT,II)
IF (METHOD.EQ.1) THEN
UPM_N(II)=UPM_O(II)+DT*DUDT(1)
VPM_N(II)=VPM_O(II)+DT*DUDT(2)
WPM_N(II)=WPM_O(II)+DT*DUDT(3)
ELSE
CALL RUNGE(UU,DUDT,UOUT,II)
UPM_N(II)=UOUT(1)
VPM_N(II)=UOUT(2)
WPM_N(II)=UOUT(3)
END IF
!
CALCULATE H & L OF JUMP
ZPC=ZPM_N(II)
IF (ZPC.GT.JUMPH(II)) JUMPH(II)=ZPC
IF (XPM_N(II)-JUMPINI(II).GT.JUMPL(II)) THEN
JUMPL(II)=XPM_N(II)-JUMPINI(II)
END IF
IF (ZPC.LE.0.5D0) THEN

```

```

        CALL WALL_COLLISION(II)
    ENDIF
550     END DO
!     CHECK FOR PARTICLE-PARTICLE COLLISION
DO 560 LL=1,N_PARTICLES
DO 565 KK=LL,N_PARTICLES
    IF(KK.GT.LL) THEN
        DX=XPM_N(LL)-XPM_N(KK)
        DY=YPM_N(LL)-YPM_N(KK)
        DZ=ZPM_N(LL)-ZPM_N(KK)
        DIST=SQRT(DX**2.0D0+DY**2.0D0+DZ**2.0D0)
        IF (DIST.LE.SAFE*1.0D0) THEN
            CALL COLLISION(LL, KK)
        END IF
    END IF
END IF
565     END DO
560     END DO
CALL ACCESORIES (TIME) !
!     OUTPUT FOR EACH PARTICLE
CALL OUTPUT (TIME, IUNIT)
!     UPDATE INFORMATION AND CHECK FOR MEMORY DEALLOCATION
!
MIN_TBEGIN=TBEGIN(1)
DO 580 II=1,N_PARTICLES
    CALL UPDATE(II) ! SWITCH OLD VALUES FOR NEW VALUES
    T=TBEGIN(II)
    MIN_TBEGIN=MIN(MIN_TBEGIN, T)
580     END DO
!     IF A NEW TIME FOR THE BASSET TERM IS COMPUTED
!     THEN THE SUBROUTINE LIB_MEMORY IS CALLED
!
IF (MIN_TBEGIN.GT.0.0D0.AND.MIN_TBEGIN.GT.MIN_TBEGIN_O) THEN
    CALL LIB_MEMORY(MIN_TBEGIN)
END IF
    MIN_TBEGIN_O=MIN_TBEGIN
1002 TIME=TIME+DT
WRITE (*, *) 'SIMULATION TIME: ', TIME
IF (TIME.GE.TIME_VP1) THEN
    CALL LOAD_VELOCITY (TIME, TIME_V, TIME_VP1)
END IF
WRITE (300, 501) TIME, T_V, T_VP1, IG, JG, KG
501     FORMAT (1( ' ', F36.20, ' '), 5( ' ', I10, ' ' ) )
JJ=JJ+1
500 END DO
DO II=1,N_PARTICLES
    WRITE (999, *) MATRIX_INDEX(II, 1), MATRIX_INDEX(II, 2)
    WRITE (991, *) XPM_O(II), YPM_O(II), ZPM_O(II)
    WRITE (992, *) UPM_O(II), VPM_O(II), WPM_O(II)
    WRITE (993, *) RPX_O(II), RPY_O(II), RPZ_O(II)
END DO
NULLIFY (N_NODE)
N_NODE => BEGIN_LIST ! MAKE CURRENT POINT TO HEAD OF LIST
K=0
!     DEALLOCATE ALL MEMORY ASSOCIATED TO BASSET FORCE
DO 600 K=0, NODE_NUMBER
IF (.NOT. ASSOCIATED(N_NODE)) EXIT ! EXIT IF NULL POINTER
BEGIN_LIST => N_NODE%NEXT ! MAKE LIST POINT TO NEXT NODE OF HEAD

```

```

DEALLOCATE (N_NODE) ! DEALLOCATE CURRENT HEAD NODE
NULLIFY (N_NODE)
N_NODE => BEGIN_LIST ! MAKE CURRENT POINT TO NEW HEAD
600 END DO
CALL CLOSE_FILES (IUNIT)
CLOSE (999)
DEALLOCATE (X)
DEALLOCATE (Y)
DEALLOCATE (Z)
DEALLOCATE (U)
DEALLOCATE (V)
DEALLOCATE (W)
DEALLOCATE (U_F)
DEALLOCATE (V_F)
DEALLOCATE (W_F)
END PROGRAM SALTATE15

```

```

MODULE VELOCITY
INCLUDE 'PARAM.FI'
INCLUDE 'INPUT.FI'
CONTAINS
! FLUID PROPERTIES
! DRAG COEFFICIENT CALCULATION (CD)
!
DOUBLE PRECISION FUNCTION DRAG (UR,T,RIP)
IMPLICIT NONE
DOUBLE PRECISION J,T,RIP,RE,UR
RE = UR*SQRT(T)*RIP
DRAG = 24/RE*(1.0D0+0.15D0*SQRT(RE)+0.017D0*RE) -
0.208D0/(1.0D0+10000.D0*RE**(-0.5D0))
END FUNCTION DRAG
! VELOCITY FIELD AT (IG,JG,KG) OBTAIN AS A LINEAR INTERPOLATION OF
! TIME_V AND TIME_VP1
DOUBLE PRECISION FUNCTION U_VELOCITY_TEMP (IG,KG,JG)
IMPLICIT NONE
INTEGER IG,JG,KG
DOUBLE PRECISION U_V,U_TIME,U_TIMEP1
U_TIME=U(IG,KG,JG)
U_TIMEP1=U_F(IG,KG,JG)
IF (TIME.EQ.TIME_V) THEN
    U_VELOCITY_TEMP=U_TIME
ELSE IF (TIME.EQ.TIME_VP1) THEN
    U_VELOCITY_TEMP=U_TIMEP1
ELSE
    U_VELOCITY_TEMP=U_TIME+(U_TIMEP1-U_TIME)/(TIME_VP1-TIME_V)*(TIME-
TIME_V)
END IF
END FUNCTION U_VELOCITY_TEMP

DOUBLE PRECISION FUNCTION V_VELOCITY_TEMP (IG,KG,JG)
IMPLICIT NONE
INTEGER IG,JG,KG
DOUBLE PRECISION U_V,U_TIME,U_TIMEP1
U_TIME=V(IG,KG,JG)
U_TIMEP1=V_F(IG,KG,JG)
IF (TIME.EQ.TIME_V) THEN
    V_VELOCITY_TEMP=U_TIME

```

```

ELSE IF (TIME.EQ.TIME_VP1) THEN
  V_VELOCITY_TEMP=U_TIMEP1
ELSE
  V_VELOCITY_TEMP=U_TIME+(U_TIMEP1-U_TIME)/(TIME_VP1-TIME_V)*(TIME-
TIME_V)
END IF
END FUNCTION V_VELOCITY_TEMP

```

```

DOUBLE PRECISION FUNCTION W_VELOCITY_TEMP (IG,KG,JG)
IMPLICIT NONE
INTEGER IG,JG,KG
DOUBLE PRECISION U_V,U_TIME,U_TIMEP1
U_TIME=W(IG,KG,JG)
U_TIMEP1=W_F(IG,KG,JG)
IF (TIME.EQ.TIME_V) THEN
W_VELOCITY_TEMP=U_TIME
ELSE IF (TIME.EQ.TIME_VP1) THEN
  W_VELOCITY_TEMP=U_TIMEP1
ELSE
  W_VELOCITY_TEMP=U_TIME+(U_TIMEP1-U_TIME)/(TIME_VP1-TIME_V)*(TIME-
TIME_V)
END IF
END FUNCTION W_VELOCITY_TEMP

```

```

! CALCULATE SPATIAL DERIVATIVES AT LOCATION (IG,JG,KG)

```

```

SUBROUTINE FLU_DERIV(IG,KG,JG)
IMPLICIT NONE
INTEGER IG,JG,KG
REAL UP1,UM1,VP1,VM1,WP1,WM1
REAL XP1,XM1,YP1,YM1,ZP1,ZM1
IF (IG.NE.NX) THEN
  UP1=U_VELOCITY_TEMP(IG+1,KG,JG)
  UM1=U_VELOCITY_TEMP(IG,KG,JG)
  VP1=V_VELOCITY_TEMP(IG+1,KG,JG)
  VM1=V_VELOCITY_TEMP(IG,KG,JG)
  WP1=W_VELOCITY_TEMP(IG+1,KG,JG)
  WM1=W_VELOCITY_TEMP(IG,KG,JG)
  XP1=X(IG+1)
  XM1=X(IG)
ELSE IF (IG.EQ.NX) THEN
  UP1=U_VELOCITY_TEMP(IG,KG,JG)
  UM1=U_VELOCITY_TEMP(IG-1,KG,JG)
  VP1=V_VELOCITY_TEMP(IG,KG,JG)
  VM1=V_VELOCITY_TEMP(IG-1,KG,JG)
  WP1=W_VELOCITY_TEMP(IG,KG,JG)
  WM1=W_VELOCITY_TEMP(IG-1,KG,JG)
  XP1=X(IG)
  XM1=X(IG-1)
END IF
DU_DX=(UP1-UM1)/(XP1-XM1)
DV_DX=(VP1-VM1)/(XP1-XM1)
DW_DX=(WP1-WM1)/(XP1-XM1)
IF (JG.NE.NY) THEN
  UP1=U_VELOCITY_TEMP(IG,KG,JG+1)
  UM1=U_VELOCITY_TEMP(IG,KG,JG)
  VP1=V_VELOCITY_TEMP(IG,KG,JG+1)
  VM1=V_VELOCITY_TEMP(IG,KG,JG)

```

```

      WP1=W_VELOCITY_TEMP (IG, KG, JG+1)
      WM1=W_VELOCITY_TEMP (IG, KG, JG)
      YP1=Y (JG+1)
      YM1=Y (JG)
ELSE IF (JG.EQ.NY) THEN
      UP1=U_VELOCITY_TEMP (IG, KG, JG)
      UM1=U_VELOCITY_TEMP (IG, KG, JG-1)
      VP1=V_VELOCITY_TEMP (IG, KG, JG)
      VM1=V_VELOCITY_TEMP (IG, KG, JG-1)
      WP1=W_VELOCITY_TEMP (IG, KG, JG)
      WM1=W_VELOCITY_TEMP (IG, KG, JG-1)
      YP1=Y (JG)
      YM1=Y (JG-1)
END IF
DU_DY=(UP1-UM1)/(YP1-YM1)
DV_DY=(VP1-VM1)/(YP1-YM1)
DW_DY=(WP1-WM1)/(YP1-YM1)
IF (KG.NE.NZ) THEN
      UP1=U_VELOCITY_TEMP (IG, KG+1, JG)
      UM1=U_VELOCITY_TEMP (IG, KG, JG)
      VP1=V_VELOCITY_TEMP (IG, KG+1, JG)
      VM1=V_VELOCITY_TEMP (IG, KG, JG)
      WP1=W_VELOCITY_TEMP (IG, KG+1, JG)
      WM1=W_VELOCITY_TEMP (IG, KG, JG)
      ZP1=Z (KG+1)
      ZM1=Z (KG)
ELSE IF (KG.EQ.NZ) THEN
      UP1=U_VELOCITY_TEMP (IG, KG, JG)
      UM1=U_VELOCITY_TEMP (IG, KG-1, JG)
      VP1=V_VELOCITY_TEMP (IG, KG, JG)
      VM1=V_VELOCITY_TEMP (IG, KG-1, JG)
      WP1=W_VELOCITY_TEMP (IG, KG, JG)
      WM1=W_VELOCITY_TEMP (IG, KG-1, JG)
      ZP1=Z (KG)
      ZM1=Z (KG-1)
END IF
DU_DZ=(UP1-UM1)/(ZP1-ZM1)
DV_DZ=(VP1-VM1)/(ZP1-ZM1)
DW_DZ=(WP1-WM1)/(ZP1-ZM1)
AUF=(U_F (IG, KG, JG) -U (IG, KG, JG))/(TIME_VP1-TIME_V)
AVF=(V_F (IG, KG, JG) -V (IG, KG, JG))/(TIME_VP1-TIME_V)
AWF=(W_F (IG, KG, JG) -W (IG, KG, JG))/(TIME_VP1-TIME_V)
END SUBROUTINE FLU_DERIV

!      FINAL CALCULATION OF THE VELOCITY FIELD AT LOCATION (IG, JG, KG)
!      INCLUDES BOTH EFFECT: SPATIAL AND TEMPORAL VARIATION
DOUBLE PRECISION FUNCTION U_VELOCITY (IG, KG, JG)
IMPLICIT NONE
INTEGER IG, JG, KG
DOUBLE PRECISION XP, YP, ZP, XI, YJ, ZK
XP=XPM_O (II)
YP=YPM_O (II)
ZP=ZPM_O (II)
XI=X (IG)
YJ=Y (JG)
ZK=Z (KG)

```

```

U_VELOCITY=U_VELOCITY_TEMP(IG,KG,JG)+DU_DX*(XP-XI)+DU_DY*(YP-
YJ)+DU_DZ*(ZP-ZK)
END FUNCTION U_VELOCITY

```

```

DOUBLE PRECISION FUNCTION V_VELOCITY(IG,KG,JG)
IMPLICIT NONE
INTEGER IG,KG,JG
DOUBLE PRECISION XP,YP,ZP,XI,YJ,ZK
XP=XPM_O(II)
YP=YPM_O(II)
ZP=ZPM_O(II)
XI=X(IG)
YJ=Y(JG)
ZK=Z(KG)
V_VELOCITY=V_VELOCITY_TEMP(IG,KG,JG)+DV_DX*(XP-XI)+DV_DY*(YP-
YJ)+DV_DZ*(ZP-ZK)
END FUNCTION V_VELOCITY

```

```

DOUBLE PRECISION FUNCTION W_VELOCITY(IG,KG,JG)
IMPLICIT NONE
INTEGER IG,KG,JG
DOUBLE PRECISION XP,YP,ZP,XI,YJ,ZK
XP=XPM_O(II)
YP=YPM_O(II)
ZP=ZPM_O(II)
XI=X(IG)
YJ=Y(JG)
ZK=Z(KG)
W_VELOCITY=W_VELOCITY_TEMP(IG,KG,JG)+DW_DX*(XP-XI)+DW_DY*(YP-YJ)
+DW_DZ*(ZP-ZK)
END FUNCTION W_VELOCITY

```

```

! VORTICITY CALCULATION AT (IG,JG,KG) USING SPATIAL AND TEMPORAL
! DERIVATIVES AT THE SPECIFIC LOCATION
DOUBLE PRECISION FUNCTION VORTICITY_X (IG,KG,JG)
IMPLICIT NONE
INTEGER IG,KG,JG
DOUBLE PRECISION DWDY,DVDZ
VORTICITY_X=DW_DY-DV_DZ
END FUNCTION VORTICITY_X

```

```

DOUBLE PRECISION FUNCTION VORTICITY_Y (IG,KG,JG)
IMPLICIT NONE
INTEGER IG,KG,JG
DOUBLE PRECISION DUDZ,DWDX
VORTICITY_Y=DU_DZ-DW_DX
END FUNCTION VORTICITY_Y

```

```

DOUBLE PRECISION FUNCTION VORTICITY_Z (IG,KG,JG)
IMPLICIT NONE
INTEGER IG,KG,JG
DOUBLE PRECISION DVDX,DUDY
VORTICITY_Z=DV_DX-DU_DY
END FUNCTION VORTICITY_Z

```

```

! ROTATION-DRAG COEFFICIENT CALCULATION (CT)
!
```

```

DOUBLE PRECISION FUNCTION DRAG_ROT(RR,T,RIP)
IMPLICIT NONE
DOUBLE PRECISION J,T,RIP,RER,RR
DOUBLE PRECISION, DIMENSION(3) :: C
RER =MIN(RR*SQRT(T)*RIP/4.0D0,300.0D0)
IF(RER.LE.1.0D0) THEN
    C = (/ 0.0D0, 50.27D0, 0.0D0 /)
ELSE IF (RER.GT.1.0D0.AND.RER.LE.10.0D0) THEN
    C = (/ 0.0D0, 50.27D0, 0.0418D0 /)
ELSE IF (RER.GT.10.0D0.AND.RER.LE.20.0D0) THEN
    C = (/ 5.32D0, 37.2D0, 5.32D0 /)
ELSE IF (RER.GT.20.0D0.AND.RER.LE.50.0D0) THEN
    C = (/ 6.44D0, 32.2D0, 6.44D0 /)
ELSE
    C = (/ 6.45D0, 32.1D0, 6.45D0/)
END IF
DRAG_ROT = C(1)/SQRT(RER)+C(2)/RER+C(3)*RER
END FUNCTION DRAG_ROT

SUBROUTINE BASSET(T,II)
IMPLICIT NONE
DOUBLE PRECISION SUMU,T,X,URU,URD,UR,T0,I_UP,I_DW
DOUBLE PRECISION SUMW,WRU,WRD,WR,GAMMA,TBACK_A
DOUBLE PRECISION SUMV,VRU,VRD,VR,T_NODE
INTEGER II,K,JU,JD,N_MAX
IF(T.LT.DT) THEN
    BUINTE=0.0D0
    BVINTE=0.0D0
    BWINTE=0.0D0
ELSE
    SUMU=0.0D0
    SUMV=0.0D0
    SUMW=0.0D0
    K=0
    TBACK_A=TBEGIN(II)
    IF(T.GT.TBACK_A) THEN
        T0=TBACK_A
    ELSE
        T0=0.0D0
    END IF
    N_MAX=MAX(1000,CEILING((T-T0)/DT))
    NULLIFY(U_NODE)
    U_NODE=>BEGIN_LIST
    T_NODE=U_NODE*T
    DO 125 WHILE (K.LE.N_MAX-1)
        X=(T-K*(T-T0)/N_MAX)
        DO 110 WHILE (T_NODE.GT.X.AND.ASSOCIATED(U_NODE%NEXT))
            U_NODE=>U_NODE%NEXT
            T_NODE=U_NODE*T
            IF(.NOT.ASSOCIATED(U_NODE%NEXT)) T_NODE=-1.0D0
110        END DO
        UR=U_NODE%BASSET_VALUE(1,II)
        VR=U_NODE%BASSET_VALUE(2,II)
        WR=U_NODE%BASSET_VALUE(3,II)
        I_UP=K+0.5D0
        I_DW=K+1.0D0
        GAMMA=EXP(DGAMMALN(I_UP)-DGAMMALN(I_DW))

```

```

SUMU=SUMU+GAMMA*UR
SUMV=SUMV+GAMMA*VR
SUMW=SUMW+GAMMA*WR
K=K+1
125     END DO
BUINTE=SUMU*SQRT((T-T0)/N_MAX)
BVINTE=SUMV*SQRT((T-T0)/N_MAX)
BWINTE=SUMW*SQRT((T-T0)/N_MAX)
END IF
END SUBROUTINE BASSET
! PARTICLE VELOCITY DERIVATIVE FOR PARTICLE II
!
SUBROUTINE DERIV(T,UU,DUdT,SS,II)
IMPLICIT NONE
DOUBLE PRECISION UPOLD,WPOLD,UROLD,UFOLD,T
DOUBLE PRECISION VPOLD,VFOLD,SS,UFLUID
DOUBLE PRECISION DUFO,WFOLD,URTOP,URBOT,Z_AUX
DOUBLE PRECISION, DIMENSION (3)::DUdT,UU
INTEGER II,I,J,K
UPOLD=UU(1)
VPOLD=UU(2)
WPOLD=UU(3)
CALL LOCATE(II,IG,KG,JG)
UFOLD=U_VELOCITY(IG,KG,JG)
VFOLD=V_VELOCITY(IG,KG,JG)
WFOLD=W_VELOCITY(IG,KG,JG)
CALL FLU_DERIV(IG,KG,JG)
UROLD=SQRT((UPOLD-UFOLD)**2.0D0+(WPOLD-WFOLD)**2.0D0+(VPOLD-
VFOLD)**2.0D0)
CD=DRAG(UROLD,TAUS,RP)
BETA=-0.75D0*ALFA*CD
Z_AUX=ZPM_O(II)           !KEEP OLD VALUE OF ZPM
ZPM_O(II)=ZPM_O(II)+0.5D0 !RECALCULATE ZPM FOR LIFT FORCE
CALL LOCATE(II,I,K,J)
UFLUID=U_VELOCITY(I,K,J)
URTOP=(UPOLD-UFLUID)**2.0D0+(VPOLD-VFOLD)**2.0D0+(WPOLD-WFOLD)**2.0D0
ZPM_O(II)=Z_AUX-0.5D0    !RECALCULATE ZPM FOR LIFT FORCE
IF(ZPM_O(II).GT.0.5D0) THEN
CALL LOCATE(II,I,K,J)
UFLUID=U_VELOCITY(I,K,J)
URBOT=(UPOLD-UFLUID)**2.0D0+(VPOLD-VFOLD)**2.0D0+(WPOLD-WFOLD)**2.0D0
ELSE
URBOT=UPOLD**2.0D0+(WPOLD-WFOLD)**2.0D0+(VPOLD-VFOLD)**2.0D0
END IF
URMT2(II)=URTOP
URMB2(II)=URBOT
ZPM_O(II)=Z_AUX          !RESTORE OLD VALUE OF ZPM_O
IF(KAPA.NE.0.0D0) CALL BASSET(T,II)
DUdT(1)=BETA*UROLD*(UPOLD-UFOLD)+
EPS*(UPOLD*DU_DX+VPOLD*DU_DY+WPOLD*DU_DZ+AUF)+GU+KAPA*BUINTE
DUdT(2)=BETA*UROLD*(VPOLD-VFOLD)+KAPA*BVINTE
+EPS*(UPOLD*DV_DX+VPOLD*DV_DY+WPOLD*DV_DZ+AVF)
DUdT(3)=BETA*UROLD*(WPOLD-WFOLD)+DELTA*(URTOP-URBOT)-GW
+KAPA*BWINTE+0.75D0*ALFA*UROLD*(SS-0.5D0*DU_DZ)*MG
+EPS*(UPOLD*DW_DX+VPOLD*DW_DY+WPOLD*DW_DZ+AWF)
END SUBROUTINE DERIV
! RUNGE KUTTA U VELOCITY

```

```

!
SUBROUTINE RUNGE(Y, DYDX, YOUT, SS, II)
IMPLICIT NONE
INTEGER II, JJ, I
DOUBLE PRECISION HH, H6, XH, SS
DOUBLE PRECISION, DIMENSION (3) :: DYM, DYT, YT, Y, DYDX, YOUT
HH=DT*0.5D0
H6=DT/6.0D0
! FIRST STEP
DO 11 I=1, 3
    YT(I)=Y(I)+HH*DYDX(I)
11 END DO
! SECOND STEP
CALL DERIV(TIME-0.5D0*DT, YT, DYT, SS, II)
! SECOND STEP
DO 21 I=1, 3
    YT(I)=Y(I)+HH*DYT(I)
21 END DO
CALL DERIV(TIME-0.5D0*DT, YT, DYM, SS, II)
! THIRD STEP
DO 31 I=1, 3
    YT(I)=Y(I)+DT*DYM(I)
    DYM(I)=DYT(I)+DYM(I)
31 END DO
CALL DERIV(TIME, YT, DYT, SS, II)
! FOURTH STEP
DO 41 I=1, 3
    YOUT(I)=Y(I)+(DYDX(I)+DYT(I)+DYM(I)*2.0D0)*H6
41 END DO
END SUBROUTINE RUNGE

! CALCULATE VALUES OF ACCELERATION AND BASSET THE VALUE OF U, V AFTER
! RUNGE KUTTA APPROX.
SUBROUTINE ACCESORIES(TIME)
IMPLICIT NONE
INTEGER II
DOUBLE PRECISION VFY, VFX, VFZ, TIME
DOUBLE PRECISION UPOLD, VPOLD, WPOLD
ALLOCATE(N_NODE) ! CREATE NEW NODE
NODE_NUMBER=NODE_NUMBER+1
DO II=1, N_PARTICLES, 1
    AUP(II)=(UPM_N(II)-UPM_O(II))/DT
    AVP(II)=(VPM_N(II)-VPM_O(II))/DT
    AWP(II)=(WPM_N(II)-WPM_O(II))/DT
    UPOLD=UPM_O(II)
    VPOLD=VPM_O(II)
    WPOLD=WPM_O(II)
    CALL LOCATE(II, IG, KG, JG)
    UFM(II)=U_VELOCITY(IG, KG, JG)
    VFM(II)=V_VELOCITY(IG, KG, JG)
    WFM(II)=W_VELOCITY(IG, KG, JG)
    CALL FLU_DERIV(IG, KG, JG)
    URM(II)=SQRT((UPM_N(II)-UFM(II))**2.0D0
+ (VPM_N(II)-VFM(II))**2.0D0+(WPM_N(II)-WFM(II))**2.0D0)

    N_NODE%T = TIME ! GIVING THE TIME VALUE
    N_NODE%BASSET_VALUE(1, II) = AUF-AUP(II)+UPOLD*DU_DX+VPOLD*DU_DY

```

```

+WPOLD*DU_DZ
      N_NODE%BASSET_VALUE(2,II) = AVF-AVP(II)+UPOLD*DV_DX+VPOLD*DV_DY
+WPOLD*DV_DZ
      N_NODE%BASSET_VALUE(3,II) = AWF-AWP(II)+UPOLD*DW_DX+VPOLD*DW_DY
+WPOLD*DW_DZ
!     PARTICLE ROTATION MOTION
      VFX=VORTICITY_X(IG,KG,JG)
      VFY=VORTICITY_Y(IG,KG,JG)
      VFZ=VORTICITY_Z(IG,KG,JG)
      RREL(II)=SQRT((RPX_O(II))**2.0D0+(RPY_O(II)-
VFY)**2.0D0+RPZ_O(II))**2.0D0)
      RRA=RREL(II)
      CT=DRAG_ROT(RRA,TAUS,RP)
      ETA=-15.0D0*CT/16.0D0/PI/(R+1.0D0)
      ARX(II)=MIN(10.0D0,MAX(-10.0D0,ETA*RREL(II)*(RPX_O(II))))
      ARY(II)=MIN(10.0D0,MAX(-10.0D0,ETA*RREL(II)*(RPY_O(II)-VFY))
      ARZ(II)=MIN(10.0D0,MAX(-10.0D0,ETA*RREL(II)*(RPZ_O(II))))
END DO
N_NODE%NEXT => BEGIN_LIST ! POINT TO PREVIOUS ONE
BEGIN_LIST => N_NODE ! UPDATE HEAD OF LIST
END SUBROUTINE ACCESORIES
!     THIS SUBROUTINES ALLOWS THE CALCULATION OF THE NEW TIME STEP
!     SETTING THE OLD VALUES FOR THE NEW VALUES
!
SUBROUTINE UPDATE (II)
IMPLICIT NONE
INTEGER II
UPM_O(II)=UPM_N(II)
VPM_O(II)=VPM_N(II)
WPM_O(II)=WPM_N(II)
XPM_O(II)=XPM_N(II)
YPM_O(II)=YPM_N(II)
ZPM_O(II)=ZPM_N(II)
RPX_O(II)=RPX_N(II)
RPY_O(II)=RPY_N(II)
RPZ_O(II)=RPZ_N(II)
END SUBROUTINE UPDATE
!     RANDOM NUMBERS GENERATOR
!     MINIMAL RANDOM NUMBER RENERATOR OF PARKER & MILLER. RETURNS
!     UNIFORM NUMBER BETWEEN 0-1. CALL WITH IDUM A NEGATIVE INTEGER TO
INITIALIZA
!     DO NOT ALTER IDUM BETWEEN SUCCESSIVE DEVIATES IN A SECUENCE.
FUNCTION RAN1(IDUM)
INTEGER IDUM,IA,IM,IQ,IR,NTAB,NDIV
REAL RAN1,AM,EPS,RNMX
PARAMETER(IA=16807,IM=2147483647,AM=1./IM,IQ=127773,IR=2836,NTAB=32,NDI
V=1+(IM-1)/NTAB,EPS=1.2E-7,RNMX=1.-EPS)
INTEGER J,K,IV(NTAB),IY
SAVE IV,IY
DATA IV /NTAB*0/, IY/0/
IF(IDUM.LE.0.OR.IY.EQ.0) THEN
IDUM=MAX(-IDUM,1)
DO J=NTAB+8,1,-1
      K=IDUM/IQ
      IDUM=IA*(IDUM-K*IQ)-IR*K
      IF(IDUM.LT.0) IDUM=IDUM+IM
      IF(J.LE.NTAB) IV(J)=IDUM

```

```

END DO
IY=IV(1)
END IF
K=IDUM/IQ
IDUM=IA*(IDUM-K*IQ)-IR*K
IF(IDUM.LT.0) IDUM=IDUM+IM
J=1+IY/NDIV
IY=IV(J)
IV(J)=IDUM
RAN1=MIN(AM*IY,RNMX)
RETURN
END FUNCTION RAN1
!     RANDOM GENERATOR NUMBER USING A NORMAL (0,1) DISTRIBUTION
!
SUBROUTINE RANDOM(IDUM,GASDEV)
INTEGER IDUM, ISET
REAL GASDEV, FAC, GSET, RSQ, V1, V2
DATA ISET/0/
SAVE ISET, GSET
IF (ISET.EQ.0) THEN
1     V1=2.0D0*RAN1(IDUM)-1.0D0
      V2=2.0D0*RAN1(IDUM)-1.0D0
      RSQ=V1**2+V2**2
      IF(RSQ.GE.1.0D0.OR.RSQ.EQ.0.0D0) GOTO 1
      FAC=SQRT(-2.0D0*LOG(RSQ)/RSQ)
      GSET=V1*FAC
      GASDEV=V2*FAC
      ISET=1
ELSE
      GASDEV=GSET
      ISET=0
END IF
RETURN
END SUBROUTINE RANDOM
!     USING THE RANDOM VALUE GENERATED BY RAN1, THIS FUNCTION
!     RECALCULATE THE LIMITS OF THE UNIFORM DISTRIBUTION
!     FORM (0,1) TO (RRMIN,RRMAX)
!
DOUBLE PRECISION FUNCTION LIMITATE(RRMIN,RRMAX,X)
DOUBLE PRECISION RRMIN,RRMAX,X
      LIMITATE=RRMIN+(RRMAX-RRMIN)*X
END FUNCTION LIMITATE
!
!     WALL COLLISION ALGORITHM
!     COMPUTE VALUE OF RCRIT GIVEN A VALUE OF TITC AND TITB
!     IT WILL BE USED TO COMPUTE RANDOM REBOUND ANGLE
!
DOUBLE PRECISION FUNCTION CRIT_R(TITC,TITB)
DOUBLE PRECISION TITC,TITB
IF (TITB.GT.0.0D0) THEN
      CRIT_R=0.5D0*(COS(TITB)-TAN(TITC)*SIN(TITB))
ELSE
      CRIT_R=0.5D0*(COS(ABS(TITB))+TAN(TITC)*SIN(ABS(TITB)))
END IF
END FUNCTION CRIT_R
!     TITAB CALCULATION USING R VALUE
!
DOUBLE PRECISION FUNCTION CAL_TITAB(TET,RN)

```

```

DOUBLE PRECISION TET,XK,DIFF,EXPR,NEW_XK
DOUBLE PRECISION RN
XK=PI/6.0D0
DIFF=1000.0D0
DO 100 WHILE (DIFF.GT.0.001D0)
NEW_XK=XK-(RN-0.5D0*SIN(XK)+0.5D0*COS(XK)*TAN(TET))/(-0.5D0*COS(XK)-
0.5D0*SIN(XK)*TAN(TET))
DIFF=ABS(NEW_XK-XK)
XK=NEW_XK
100      END DO
CAL_TITAB=PI*(0.5D0)-XK
END FUNCTION CAL_TITAB
!
!      CREATE MATRIX AND INVERSE MATRIX
SUBROUTINE CREATE_M(M,INV_M,B,A)
DOUBLE PRECISION,DIMENSION (3,3)::M,INV_M
DOUBLE PRECISION A,B
M(1,1)=COS(A)*COS(B)
M(1,2)=-COS(B)*SIN(A)
M(1,3)=-SIN(B)
M(2,1)=SIN(A)
M(2,2)=COS(A)
M(2,3)=0.0D0
M(3,1)=COS(A)*SIN(B)
M(3,2)=-SIN(A)*SIN(B)
M(3,3)=COS(B)
INV_M(1,1)=COS(A)*COS(B)
INV_M(1,2)=SIN(A)
INV_M(1,3)=COS(A)*SIN(B)
INV_M(2,1)=-COS(B)*SIN(A)
INV_M(2,2)=COS(A)
INV_M(2,3)=-SIN(A)*SIN(B)
INV_M(3,1)=-SIN(B)
INV_M(3,2)=0.0D0
INV_M(3,3)=COS(B)
END SUBROUTINE CREATE_M
!
!      THIS SUBROUTINE COMPUTE X=AB (A:MATRIX, X,B:VECTOR)
SUBROUTINE MULTIPLY(A,B,X)
DOUBLE PRECISION,DIMENSION (3,3)::A
DOUBLE PRECISION,DIMENSION (3)::X,B
X(1)=A(1,1)*B(1)+A(1,2)*B(2)+A(1,3)*B(3)
X(2)=A(2,1)*B(1)+A(2,2)*B(2)+A(2,3)*B(3)
X(3)=A(3,1)*B(1)+A(3,2)*B(2)+A(3,3)*B(3)
END SUBROUTINE MULTIPLY
!
!      REBOUND SUBROUTINE (USING CROWE ET AL 1997) ALGORITHM
SUBROUTINE REBOUND (VIN,WIN,VOUT,WOUT)
DOUBLE PRECISION,DIMENSION (3)::VIN,VOUT,WIN,WOUT
DOUBLE PRECISION VMOD,VALUE,A
A=0.5D0      !A=PARTICLE RADIOUS
VMOD=SQRT((VIN(1)+A*WIN(2))**2.0D0+(VIN(2)-A*WIN(1))**2.0D0)
EX=(VIN(1)+A*WIN(2))/VMOD
EY=(VIN(2)-A*WIN(1))/VMOD
VALUE=-2.0D0/7.0D0/MU/(ECOL+1.0D0)
IF(VIN(3)/VMOD.LT.VALUE) THEN
VOUT(1)=5.0D0/7.0D0*(VIN(1)-2.0D0*A/5.0D0*WIN(2))

```

```

VOUT(2)=5.0D0/7.0D0*(VIN(2)+2.0D0*A/5.0D0*WIN(1))
VOUT(3)=-ECOL*VIN(3)
WOUT(1)=VOUT(2)/A
WOUT(2)=-VOUT(1)/A
WOUT(3)=WIN(3)
ELSE
VOUT(1)=VIN(1)+EX*MU*(ECOL+1.0D0)*VIN(3)
VOUT(2)=VIN(2)+EY*MU*(ECOL+1.0D0)*VIN(3)
VOUT(3)=-ECOL*VIN(3)
WOUT(1)=WIN(1)-2.0D0/5.0D0/A*EY*MU*(ECOL+1.0D0)*VIN(3)
WOUT(2)=WIN(2)+2.0D0/5.0D0/A*EX*MU*(ECOL+1.0D0)*VIN(3)
WOUT(3)=WIN(3)
END IF
END SUBROUTINE REBOUND
! WALL COLLISION SUBROUTINE
!
SUBROUTINE WALL_COLLISION(II)
IMPLICIT NONE
DOUBLE PRECISION UIN,UPPROM,DIFFER,X,CRIT_ANGLE
DOUBLE PRECISION RMIN,RMAX,ALFAC,ALFAB
DOUBLE PRECISION,DIMENSION (3,3)::MATRIX,INV_MATRIX
DOUBLE PRECISION,DIMENSION (3)::V,W,VIN,VOUT,WIN,WOUT
INTEGER KK,JJ,II,R
CRIT_ANGLE=PI/2-ACOS(1.0D0/3.0D0)
TITAC=ATAN(ABS(WPM_O(II))/UPM_O(II))
RMIN=0.0D0
RMAX=CRIT_R(TITAC,CRIT_ANGLE)
X=RAN1(-JJ-II-1000)
RR=LIMITATE(RMIN,RMAX,X)
TITAB=CAL_TITAB(TITAC,RR)
ALFAC=ATAN(VPM_O(II)/UPM_O(II))
X=RAN1(-JJ-II-900)
RR=LIMITATE(RMIN,RMAX,X)
ALFAB=CAL_TITAB(ALFAC,RR)
! CREATE MATRIX AND INV_MATRIX
CALL CREATE_M(MATRIX,INV_MATRIX,TITAB,ALFAB)
V(1)=UPM_O(II)
V(2)=VPM_O(II)
V(3)=WPM_O(II)
W(1)=RPX_O(II)
W(2)=RPY_O(II)
W(3)=RPZ_O(II)
!C CHANGE OF COORDINATE (ALIGN WITH PLANE OF COLISION)
CALL MULTIPLY(INV_MATRIX,V,VIN)
CALL MULTIPLY(INV_MATRIX,W,WIN)
!C COMPUTE VOUT AND WOUT AFTER REBOUND
CALL REBOUND(VIN,WIN,VOUT,WOUT)
!C CHANGE OF COORFINATE (ALIGN WITH ORIGINAL SISTEM OF REFERENCE)
CALL MULTIPLY(MATRIX,VOUT,V)
CALL MULTIPLY(MATRIX,WOUT,W)
!CC NEW VELOCITIES AFTER REBOUND
UPM_N(II)=V(1)
VPM_N(II)=V(2)
WPM_N(II)=V(3)
RPX_N(II)=W(1)
RPY_N(II)=W(2)
RPZ_N(II)=W(3)

```

```

ZPM_N(II)=0.5D0
CALL SAVE_WALL(II)
END SUBROUTINE WALL_COLLISION

SUBROUTINE SAVE_WALL(II)
N_REB(II)=N_REB(II)+1
MATRIX_INDEX(II,1)=MATRIX_INDEX(II,1)+1
N_REB_A=N_REB(II)
T_HIT(N_REB_A,II)=TIME
IF (N_REB_A.GE.N_BACK+1) THEN
    TBEGIN(II)=T_HIT(N_REB_A-N_BACK,II) !TIME FOR BASSET COMP
END IF
WRITE(IUNIT+II+2000,101) II, TIME,JUMPH(II)-0.5D0, JUMPL(II)
JUMPH(II)=0.0D0
JUMPL(II)=0.0D0
JUMPINI(II)=XPM_N(II)
101  FORMAT(' ', I10, ' ', F24.12, 2(' ', F24.20, ' '))
END SUBROUTINE SAVE_WALL
!      COLLISION ALGORITHM GIBEN BY YAMAMOTO ET AL. 2001
!THIS ALG SHOULD GIVE A NEW VALUE OF THE VELOCITIES OF THE PARTICLE AT
!JJ-1 AND A NEW POSITION FOR THE PARTICLES AT JJ
SUBROUTINE COLLISION(I,K)
INTEGER I,K
DOUBLE PRECISION,DIMENSION (1:3)::J,N,NXJ
DOUBLE PRECISION TITA_N
CALL NORMAL_V(I,K,N)
CALL IMPULSIVE_F(I,K,N,J)
CALL CROSS(N,J,NXJ)
UPM_N(I)=UPM_O(I)+J(1)
VPM_N(I)=VPM_O(I)+J(2)
WPM_N(I)=WPM_O(I)+J(3)
UPM_N(K)=UPM_O(K)-J(1)
VPM_N(K)=VPM_O(K)-J(2)
WPM_N(K)=WPM_O(K)-J(3)
RPX_N(I)=RPX_O(I)+1.25D0*NXJ(1)
RPX_N(K)=RPX_O(K)+1.25D0*NXJ(1)
RPY_N(I)=RPY_O(I)+1.25D0*NXJ(2)
RPY_N(K)=RPY_O(K)+1.25D0*NXJ(2)
RPZ_N(I)=RPZ_O(I)+1.25D0*NXJ(3)
RPZ_N(K)=RPZ_O(K)+1.25D0*NXJ(3)
XPM_N(I)=XPM_O(I)+UPM_N(I)*DT
YPM_N(I)=YPM_O(I)+VPM_N(I)*DT
ZPM_N(I)=ZPM_O(I)+WPM_N(I)*DT
XPM_N(K)=XPM_O(K)+UPM_N(K)*DT
YPM_N(K)=YPM_O(K)+VPM_N(K)*DT
ZPM_N(K)=ZPM_O(K)+WPM_N(K)*DT
CALL SAVE_COLLISION(I,K)
END SUBROUTINE COLLISION
!      SUBROUTINE NORMAL CALCULATE THE UNIT VECTOR FROM THE CENTER OF
!      PARTICLE I TO THE CENTER OF PARTICLE K. THIS SUBROUTINE ALSO
!      PROVIDES THE ANGLE WITH RESPECT TO THE HORIZONTAL OF THIS VECTOR
!      NORMAL VECTOR HAS THREE COMPONENTS (X,Y,Z)
!
SUBROUTINE NORMAL_V(I,K,N)
INTEGER I,K
DOUBLE PRECISION,DIMENSION (1:3)::N
DOUBLE PRECISION D_X,D_Z,MOD_N

```

```

D_X=XPM_O(K)-XPM_O(I)
D_Y=YPM_O(K)-YPM_O(I)
D_Z=ZPM_O(K)-ZPM_O(I)
MOD_N=SQRT(D_X**2.0D0+D_Y**2.0D0+D_Z**2.0D0)
IF (MOD_N.NE.0.0D0) THEN
    N(1)=D_X/MOD_N
    N(2)=D_Y/MOD_N
    N(3)=D_Z/MOD_N
ELSE
    N(1)=0.0D0
    N(2)=0.0D0
    N(3)=0.0D0
END IF
END SUBROUTINE NORMAL_V
!SUBROUTINE IMPULSIVE GIVES THE IMPULSIVE FORCE EXCERTED ON PARTICLE I
! PER UNIT MASS. THE VECTOR HAS THREE COMPONENTS.
!
SUBROUTINE IMPULSIVE_F(I,K,N,J)
IMPLICIT NONE
INTEGER I,K,L
DOUBLE PRECISION,DIMENSION (1:3)::J,G,N,GFC,T
DOUBLE PRECISION,DIMENSION (1:3)::ROT_I,ROT_K,CROSS_I,CROSS_K
DOUBLE PRECISION JN,JT,GFC_MOD,TITA_N,G_DOT_N,SLIDE
G(1)=UPM_N(I)-UPM_N(K)
G(2)=VPM_N(I)-VPM_N(K)
G(3)=WPM_N(I)-WPM_N(K)
ROT_I(1)=RPX_N(I)
ROT_K(1)=RPX_N(K)
ROT_I(2)=RPY_N(I)
ROT_K(2)=RPY_N(K)
ROT_I(3)=RPZ_N(I)
ROT_K(3)=RPZ_N(K)
CALL CROSS(ROT_I,N,CROSS_I)
CALL CROSS(ROT_K,N,CROSS_K)
G_DOT_N=DOT_PRODUCT(N,G)
DO 200 L=1,3
GFC(L)=G(L)-G_DOT_N*N(L)+0.5*CROSS_I(L)+0.5D0*CROSS_K(L)
200 END DO
GFC_MOD=SQRT(GFC(1)**2.0D0+GFC(2)**2.0D0+GFC(3)**2.0D0)
! CHECK IF PARTICLE SLIDES (SLIDE > 0)
IF(GFC_MOD.NE.0.0D0) THEN
    SLIDE=2.0D0/7.0D0/MU/(1.0D0+ECOL)-G_DOT_N/GFC_MOD
ELSE
    SLIDE=2.0D0/7.0D0/MU/(1.0D0+ECOL)
END IF
IF (SLIDE.GT.0.0D0) THEN
! PARTICLE SLIDES
JN=- (1.0D0+ECOL)*0.5D0*G_DOT_N
JT=MU*JN
ELSE
! PARTICLE DOESN'T SLIDES
JN=- (1.0D0+ECOL)*0.5D0*G_DOT_N
JT=-1.0D0/7.0D0*GFC_MOD
END IF
! CALCULATE TANGENTIAL VECTOR T & J
DO 300 L=1,3
IF(GFC_MOD.NE.0.0D0) THEN

```

```

          T(L)=GFC(L)/GFC_MOD
      ELSE
          T(L)=0.0D0
      END IF
      J(L)=JN*N(L)+JT*T(L)
300      END DO
END SUBROUTINE IMPULSIVE_F

SUBROUTINE SAVE_COLLISION(I,K)
MATRIX_INDEX(I,2)=MATRIX_INDEX(I,2)+1
MATRIX_INDEX(K,2)=MATRIX_INDEX(K,2)+1
WRITE(IUNIT+I+1000)I,K,TIME,UPM_N(I),UPM_O(I),VPM_N(I),VPM_O(I),WPM_N(I)
,WPM_O(I),RPX_N(I),RPX_O(I),RPY_N(I),RPY_O(I),RPZ_N(I),RPZ_O(I)
WRITE(IUNIT+K+1000)I,K,TIME,UPM_N(K),UPM_O(K),VPM_N(K),VPM_O(K),WPM_N(K)
),WPM_O(K),RPX_N(K),RPX_O(K),RPY_N(K),RPY_O(K),RPZ_N(K),RPZ_O(K)
WRITE(90,302)I,K,TIME
301  FORMAT(' ',I10,' ',I10,' ',F40.24,12(' ',F40.12,' '))
302  FORMAT(' ',I10,' ',I10,' ',F40.12 )
END SUBROUTINE SAVE_COLLISION

!      LOAD GRID FILES IN X,Y,Z ARRAY
SUBROUTINE LOAD_GRID
REAL*4, ALLOCATABLE, DIMENSION(:)::AUX
OPEN (301,FILE='XMCOORD', FORM="UNFORMATTED")
OPEN (302,FILE='ZMCOORD', FORM="UNFORMATTED")
OPEN (303,FILE='YMCOORD', FORM="UNFORMATTED")
ALLOCATE (AUX(NX))
READ (301) AUX !ASSOCIATED TO IG=1..NX
X=AUX/D_D0
DEALLOCATE (AUX)
ALLOCATE (AUX(NY))
READ (302) AUX !ASSOCIATED TO JG=1..NY
Y=AUX/D_D0
DEALLOCATE (AUX)
ALLOCATE (AUX(NZ))
READ (303) AUX !ASSOCIATED TO KG=1..NZ
Z=AUX/D_D0
DEALLOCATE (AUX)
CLOSE(301)
CLOSE(302)
CLOSE(303)
X_MAX=X(NX-1)      ! DEFINE MAXIMUM SIMULATION LENGTH
X_MIN=X(1)
Y_MAX=Y(NY-1)
Y_MIN=Y(1)
Z_MAX=Z(NZ-1)
END SUBROUTINE LOAD_GRID

!      LOAD_VELOCITY FIELD MATRIX U,V,W
SUBROUTINE LOAD_VELOCITY(T,TIME_V,TIME_VP1)
REAL*4, ALLOCATABLE, DIMENSION(:,:,:)::AUX,AUX1
DOUBLE PRECISION T,TIME_V,TIME_VP1
INTEGER N_TIME,TVMAX
JUNIT=10
!      TRANSFORM TIME USED IN SALTATE TO TIME SIMULATED BY V.CALO
TIME_V=T/US_UM*D_D0*2.0D0 ! THE FILE NAME REPRESENTS B1/0.5 DT
TIME_VP1=TIME_V+1 ! THE FILE NAME REPRESENTS B1/0.5 DT
TVMAX=300 !MAX OF V CALO SIMULATION FILES

```

```

T_V=INT(TIME_V)
T_VP1=T_V+1
IF (T_VP1.GE.TVMAX) T_VP1=0
IF (T_V.EQ.TVMAX) THEN
    T_V=0
    T_VP1=1
END IF
IF (T_V.GE.TVMAX) THEN
    N_TIME=FLOOR(TIME_V/TVMAX)
    T_V=T_V-N_TIME*TVMAX+1 ! CONTINUOUS TIME DOMAIN
    T_VP1=T_V+1
END IF
IF(T_VP1.GE.TVMAX) T_VP1=T_VP1-TVMAX
IF(T_V.EQ.TVMAX) T_V=0
IF(T_V.EQ.TVMAX+1) T_V=1
IF(T_V.EQ.TVMAX+2) T_V=2
IF(T_VP1.EQ.TVMAX) T_VP1=0
IF(T_VP1.EQ.TVMAX+1) T_VP1=1
IF(T_VP1.EQ.TVMAX+2) T_VP1=2
IF (INT(T_V).EQ.29.OR.INT(T_V).EQ.198) T_V=T_V+1
IF (INT(T_VP1).EQ.29.OR.INT(T_VP1).EQ.198) T_VP1=T_VP1+1
C1='0'
C2=CHAR(INT(T_V/100)+48)
C3=CHAR(INT(INT((T_V)-INT(T_V/100)*100)/10)+48)
C4=CHAR(INT(T_V)-INT(T_V/100)*100-INT(INT((T_V)-
INT(T_V/100)*100)/10)*10+48)
B1=C1//C2//C3//C4
WRITE(*,*)'PROCESS TIME: ',B1
OPEN (JUNIT+6, CONVERT='BIG_ENDIAN', FILE='DATA.BLOCK.U.'//B1,
FORM="UNFORMATTED")
OPEN (JUNIT+7, CONVERT='BIG_ENDIAN', FILE='DATA.BLOCK.V.'//B1,
FORM="UNFORMATTED")
OPEN (JUNIT+8, CONVERT='BIG_ENDIAN', FILE='DATA.BLOCK.W.'//B1,
FORM="UNFORMATTED")
ALLOCATE (AUX (NX, NZ, NY))
READ (JUNIT+6) AUX
U=AUX/US_UM ! U VELOCITY
READ (JUNIT+7) AUX
V=AUX/US_UM ! V VELOCITY
READ (JUNIT+8) AUX
W=AUX/US_UM ! W VELOCITY
CLOSE (JUNIT+6)
CLOSE (JUNIT+7)
CLOSE (JUNIT+8)
DEALLOCATE (AUX)
! NEW TIME STEP (LINEAR INTERPOLATION)
C2=CHAR(INT(T_VP1/100)+48)
C3=CHAR(INT(INT((T_VP1)-INT(T_VP1/100)*100)/10)+48)
C4=CHAR(INT(T_VP1)-INT(T_VP1/100)*100-INT(INT((T_VP1)-
INT(T_VP1/100)*100)/10)*10+48)
B1=C1//C2//C3//C4
WRITE(*,*)'PROCESS TIME: ',B1
OPEN (JUNIT+26, CONVERT='BIG_ENDIAN', FILE='DATA.BLOCK.U.'//B1,
FORM="UNFORMATTED")
OPEN (JUNIT+27, CONVERT='BIG_ENDIAN', FILE='DATA.BLOCK.V.'//B1,
FORM="UNFORMATTED")

```

```

OPEN (JUNIT+28, CONVERT='BIG_ENDIAN', FILE='DATA.BLOCK.W.'//B1,
FORM="UNFORMATTED")
ALLOCATE (AUX (NX, NZ, NY))
READ (JUNIT+26) AUX
U_F=AUX/US_UM ! U VELOCITY
READ (JUNIT+27) AUX
V_F=AUX/US_UM ! V VELOCITY
READ (JUNIT+28) AUX
W_F=AUX/US_UM ! W VELOCITY
CLOSE (JUNIT+26)
CLOSE (JUNIT+27)
CLOSE (JUNIT+28)
DEALLOCATE (AUX)
TIME_V=TIME_V/(D_D0/US_UM*2.0D0)
TIME_VP1=TIME_VP1/(D_D0/US_UM*2.0D0)
END SUBROUTINE LOAD_VELOCITY

```

```

SUBROUTINE LOCATE (II, IG, KG, JG)
DOUBLE PRECISION XP, YP, ZP, XINI, YINI
INTEGER N_X, N_Y
IG=1 ! PARTICLES LOCATED ONLY IN THE TURBULENT REGION
JG=1
KG=1
XP=XPM_O(II)
YP=YPM_O(II)
ZP=ZPM_O(II)
XINI=X(IG)
YINI=Y(JG)
IF (XP.GT.X_MAX) THEN
    N_X=FLOOR(XP/(X_MAX-XINI))
    XP=XP-N_X*(X_MAX-XINI) ! CONTINUOUS GRID DOMAIN IN X
END IF
IF (YP.GT.Y_MAX) THEN
    N_Y=FLOOR(YP/(Y_MAX-YINI))
    YP=YP-N_Y*(Y_MAX-YINI) ! CONTINUOUS GRID DOMAIN IN Y
END IF
IF (ZP.GT.Z_MAX) ZP=Z_MAX ! LIMITS VERTICAL DIRECTION
DO 151 I=1, NX-1
    IF (XP.GE.X(I).AND.XP.LT.X(I+1)) IG=I
151 END DO
DO 152 J=1, NY-1
    IF (YP.GE.Y(J).AND.YP.LT.Y(J+1)) JG=J
152 END DO
DO 153 K=1, NZ-1
    IF (ZP.GE.Z(K).AND.ZP.LT.Z(K+1)) KG=K
153 END DO
END SUBROUTINE LOCATE
! OUTPUT FILES
!
SUBROUTINE CREATE_FILES (IUNIT)
DOUBLE PRECISION T_V
OPEN (UNIT=90, STATUS='UNKNOWN', FILE='CSTAT.DAT')
OPEN (UNIT=300, STATUS='UNKNOWN', FILE='RECORD_TIME.DAT')
OPEN (999, STATUS='UNKNOWN', FILE='MATRIX_INDEX.DAT')
DO 560 II=1, N_PARTICLES
T_V=II
C1='0'

```

```

C2=CHAR(INT(T_V/100)+48)
C3=CHAR(INT(INT((T_V)-INT(T_V/100)*100)/10)+48)
C4=CHAR(INT(T_V)-INT(T_V/100)*100-INT(INT((T_V)-
INT(T_V/100)*100)/10)*10+48)
B1=C1//C2//C3//C4
D1=C3//C4
OPEN(IUNIT+II+1000,FORM="UNFORMATTED",STATUS='UNKNOWN',FILE='CSTAT'//B1
//'.DAT')
OPEN(IUNIT+II+2000,STATUS='UNKNOWN',FILE='WSTAT'//B1//'.DAT')
OPEN(IUNIT+II+3000,FORM="UNFORMATTED",STATUS='UNKNOWN',FILE='POSIC'//B1
//'.DAT')
OPEN(IUNIT+II+4000,FORM="UNFORMATTED",STATUS='UNKNOWN',FILE='VELOC'//B1
//'.DAT')
OPEN(IUNIT+II+5000,FORM="UNFORMATTED",STATUS='UNKNOWN',FILE='ROTAT'//B1
//'.DAT')
OPEN(IUNIT+II+6000,FORM="UNFORMATTED",STATUS='UNKNOWN',FILE='FORCE'//B1
//'.DAT')
OPEN(IUNIT+II+7000,FORM="UNFORMATTED",STATUS='UNKNOWN',FILE='FLU_V_Q'//
B1//'.DAT')
560   END DO
END SUBROUTINE CREATE_FILES

```

```

SUBROUTINE OUTPUT(T,UNIT)
DOUBLE PRECISION T_V,T
INTEGER UNIT
DO 570 II=1,N_PARTICLES
    CALL LOCATE(II,IG,KG,JG)
    CALL FORCES(II)
    WRITE(IUNIT+II+3000) II, T, XPM_N(II) , YPM_N(II), ZPM_N(II)
    WRITE(IUNIT+II+4000) II, T, UPM_N(II) , VPM_N(II), WPM_N(II)
    WRITE(IUNIT+II+5000) II, T, RPX_N(II), RPY_N(II), RPZ_N(II)
    WRITE(IUNIT+II+6000) II, T, UFM(II) , VFM(II), WFM(II), Q(II)
    WRITE(IUNIT+II+7000) II, T, DRAGF,VMASF,SWG HF,BASSF,LIFTF,MAGNF
570   END DO
END SUBROUTINE OUTPUT

```

```

SUBROUTINE CLOSE_FILES(IUNIT)
DOUBLE PRECISION T_V
DO 580 II=1,N_PARTICLES
    T_V=II
    C1='0'
    C2=CHAR(INT(T_V/100)+48)
    C3=CHAR(INT(INT((T_V)-INT(T_V/100)*100)/10)+48)
    C4=CHAR(INT(T_V)-INT(T_V/100)*100-INT(INT((T_V)-
INT(T_V/100)*100)/10)*10+48)
    B1=C1//C2//C3//C4
    CLOSE(IUNIT+II+2000)
    CLOSE(IUNIT+II+3000)
    CLOSE(IUNIT+II+4000)
    CLOSE(IUNIT+II+5000)
    CLOSE(IUNIT+II+6000)
    CLOSE(IUNIT+II+7000)
580   END DO
END SUBROUTINE CLOSE_FILES

```

```

SUBROUTINE FORCES(II)

```

```

DRAGF=DR*(BETA*URM(II)*SQRT((UPM_N(II)-UFM(II))**2.0D0+(VPM_N(II)-
VFM(II))**2.0D0+(WPM_N(II)-WFM(II))**2.0D0))
VMASF=(EPS*DUF(II)*WPM_N(II))*VM
SWGHF=SQRT(GU**2.0D0+GW**2.0D0)*SW
BASSF=KAPA*SQRT(BUINTE**2.0D0+BVINTE**2.0D0+BWINTE**2.0D0)*BS
LIFTF=(DELTA*(URMT2(II)-URMB2(II)))*LF
MAGNF=(0.75D0*ALFA*URM(II)*(S-0.5D0*DUF(II))*MG)
END SUBROUTINE FORCES
!      THIS ROUTINE CALCULATE THE NATURAL LOGARITM OF THE
!      GAMMA FUNCTION FOR A DOUBLE PRECISION ARGUMENT X
DOUBLE PRECISION FUNCTION DGAMMALN(XX)
INTEGER J
DOUBLE PRECISION X, SER, STP, TMP, XX, Y, COF(6)
SAVE COF STP
DATA COF, STP/76.18009172947146D0, -86.50532032941677D0,
24.01409824083091D0, -1.231739572450155D0, 0.1208650973866179D-2
, -.5395239384953D-5, 2.5066282746310005D0/
X=XX
Y=X
TMP=X+5.5D0
TMP=(X+0.5D0)*LOG(TMP)-TMP
SER=1.000000000190015D0
DO 11 J=1,6
      Y=Y+1.0D0
      SER=SER+COF(J)/Y
11  END DO
DGAMMALN=TMP+LOG(STP*SER/X)
RETURN
END FUNCTION DGAMMALN
!      CROSS CALCULATE THE CROSS PRODUCT BETWEEN VECTOR A AND B
!      THE RESULTANT VECTOR IS GIVEN BY AXB
!
SUBROUTINE CROSS(A,B,AXB)
DOUBLE PRECISION, DIMENSION (1:3)::A,B,AXB
INTEGER L
AXB(1)=A(2)*B(3)-A(3)*B(2)
AXB(2)=- (A(1)*B(3)-A(3)*B(1))
AXB(3)=A(1)*B(2)-A(2)*B(1)
END SUBROUTINE CROSS

SUBROUTINE LIB_MEMORY(MINT)
DOUBLE PRECISION MINT
TYPE (NODE), POINTER ::BEGIN_LIST_AUX,AUX
NULLIFY(N_NODE)
N_NODE => BEGIN_LIST ! MAKE CURRENT POINT TO HEAD OF LIST
K=0
DO 100 WHILE (N_NODE%T.GE.MINT.AND.K.LT.NODE_NUMBER)
IF (.NOT. ASSOCIATED(N_NODE)) EXIT ! EXIT IF NULL POINTER
AUX=>N_NODE
N_NODE => N_NODE%NEXT ! JUMP TO NEXT POINTER
K=K+1
100  END DO
BEGIN_LIST_AUX=>N_NODE  N_NODE => BEGIN_LIST_AUX
NULLIFY(AUX%NEXT)
DO 200
IF (.NOT. ASSOCIATED(N_NODE)) EXIT ! EXIT IF NULL POINTER
BEGIN_LIST_AUX => N_NODE%NEXT ! MAKE LIST POINT TO NEXT NODE OF HEAD

```

```
DEALLOCATE(N_NODE) ! DEALLOCATE CURRENT HEAD NODE
NULLIFY(N_NODE)
NODE_NUMBER=NODE_NUMBER-1 !REDUCE NUMBER OF NODES
N_NODE => BEGIN_LIST_AUX ! MAKE CURRENT POINT TO NEW HEAD
200   END DO
END SUBROUTINE LIB_MEMORY

END MODULE VELOCITY
```

APPENDIX B

ANALYSIS OF THE SEMIDERIVATIVE METHOD TO CALCULATE THE BASSET INTEGRAL

A Lagrangian model to describe the trajectory of a particle surrounded by fluid requires the evaluation of the forces acting around the object. Associated with the unsteadiness of the drag force, the Basset force is regarded as a “history” force, that become increasingly important for relatively small particles (Niño and García, 1998; Mordant and Pinton, 2000; Armenio and Fiorotto, 2001; Lukerchenko, 2006). The i^{th} component of this force is calculated as follows:

$$F_{Basset\ i} = \frac{9}{d_p} \sqrt{\frac{\rho_f \mu}{\pi}} \int_0^T \frac{1}{\sqrt{T-t}} \frac{d}{dt} (u_{fi} - u_{pi}) dt \quad (\text{B-1})$$

where a is the particle radius, ρ_f and μ denotes the fluid density and dynamic viscosity respectively, u_f and u_p represent the fluid and particle velocity, T represents the actual time step, t is an integration dummy variable.

It is easy to see why the Basset term has been usually neglected in the estimation of the forces acting on a particle; it requires to storage the temporal variation of the relative particle velocity with respect to the fluid, requiring a large amount of computer memory depending of the integration time considered. Also, the Basset force presents additional difficulties because the denominator in the integrand vanishes when the upper

integration limit is enforced. To circumvent this problem, Brush *et al.* (1964) suggested a method which consists in dividing this integral in two terms, as follows:

$$\int_0^t \frac{du_{ri}}{\sqrt{t-\tau}} d\tau = \int_0^{t-\delta t} \frac{du_{ri}}{\sqrt{t-\tau}} d\tau + 2\sqrt{\delta t} \left(u_j \frac{du_{fi}}{dx_j} - \frac{du_{pi}}{dt} \right) \quad (\text{B-2})$$

with δt denoting the time step employed in the numerical solution. Summation on j is implied in (A-2). This approximation was used by Niño and García (1998) to calculate the Basset term for sediment particles moving close to the bed of channel.

As it was mentioned before, the Basset force requires a lot of computational resources if the integration time is large, which is common for numerical simulation of several particles moving as bedload. Therefore, any improvement in the amount of time required to compute the Basset force would benefit the performance of any numerical simulation.

A different approach was presented by Tatom (1988) to approximate the Basset term as follows:

$$\int_0^t \frac{du_r}{\sqrt{t-\tau}} d\tau = \Gamma\left(\frac{1}{2}\right) \frac{d^{-0.5}\left(\frac{du_r}{dt}\right)}{dt^{-0.5}} \quad (\text{B-3})$$

where $\Gamma(\cdot)$ represents the gamma function (Abramowitz and Stegun 1970). The right hand side of Eq. (A-3) includes a semi-derivative, which belongs to the so-called fractional mathematics. This theory offers a generalized view of derivatives and integrals for which the orders are not necessarily integers (see Oldham and Spanier 1974, Nishimoto 1991).

For a general function $u_r(t)$, the semi-derivative can be calculated using a series, as proposed by Oldham and Spanier (1974, page 30):

$$\frac{d^q u_r(t)}{dt^q} = \lim_{N \rightarrow \infty} \left\{ \left(\frac{t}{N} \right)^{-q} \frac{1}{\Gamma(-q)} \sum_{k=0}^{N-1} \frac{\Gamma(k-q)}{\Gamma(k+1)} u_r \left(t - \frac{kt}{N} \right) \right\} \quad (\text{B-4})$$

where q is an arbitrary value (equal to -0.5 in the semi-derivative above).

Tatom did not provide further analysis with respect to the accuracy or the convergence of this method. In this appendix, both issues are addressed in order to compare the performance of this method (called from now on the semi-derivative method) with previously used method to solve the Basset integral.

The accuracy of the semi-derivative method was tested using an arbitrary function, namely, $u_r(t) = t^3/3$. Using this function, the Basset integral can be calculated both numerically and analytically. The analytical solution for this expression is given by:

$$\int_0^T \frac{df(t)}{\sqrt{T-t}} dt = \int_0^T \frac{dt^3}{\sqrt{T-t}} dt = \frac{16}{15} T^{2.5} \quad (\text{B-5})$$

For an arbitrary value of T equal to 100, the Basset integral presents a value equal to 106,667. This value was considered as a reference value, for error calculations. The numerical approximation of the Basset integral using the semi-derivative method depends on the number of terms (N) used in the sum in (B-4). I defined the relative error as the relative difference between the numerical integration and the value obtained from the analytical expression. The results are presented in Table B-1.

Table B-1: Error associated with the semi-derivative method versus N, using a test function that has analytical solution.

N	Semi-derivative Method	Error (%)
Analitical Value	106,667	-
5	120,163	13.32
50	108,002	1.86
500	106,800	0.07
1000	106,733	0.07
5000	106,680	0.06
50000	106,668	0.06

The proposed method can achieve a 2% error (as compared with the analytical solution) with less than 50 points. Selecting 500 or 1,000 points reduces the error drastically up to 0.1%, what we considered to be as almost “exact.”

Considering that the Basset integral is an improper integral, it is important to compare the performance of the semi-derivative method with known algorithms used to calculate improper integrals. One of these algorithms, the Euler-MacLaurin summation for improper integrals formula, taken verbatim from Press et al/ (1992) is used to compare with the semi-derivative method. In the comparison, the test function $u_r(t) = t^3/3$ and the same number of terms used in the quadrature was used for both cases. The results are presented in Table B-2.

Table B-2: Comparison of the method with the semi-derivative approach and the Euler-MacLaurin summation formula for an integration which has analytical solution.

N	Semi-derivative Method	Euler_MacLaurin Method	Error Semiderivative (%)	Error Euler-MacLaurin (%)
1	177245	69501	66.17	34.84
3	129337	86059	21.25	19.32
9	114125	94939	6.99	10.99

N	Semi-derivative Method	Euler_MacLaurin Method	Error Semiderivative (%)	Error Euler-MacLaurin (%)
27	109141	99929	2.32	6.32
81	107490	102783	0.77	3.64
243	106941	104426	0.26	2.10
729	106758	105373	0.09	1.21
2187	106697	105920	0.03	0.70
6561	106677	106236	0.01	0.40
19683	106670	106418	0.00	0.23
59049	106668	106523	0.00	0.13
177147	106667	106584	0.00	0.08
531441	106667	106619	0.00	0.04
1594323	106667	106639	0.00	0.03

It is clearly from Table B-2 that the semi-derivative approach provides smaller values of the error for the same number of terms used in both methods, indicating a much faster convergence of the semi-derivative approach than the Euler-MacLaurin summation formula. For instance, the error for the Euler-MacLaurin formula at about 1,000 terms is of the order of 1%, more than 10 times larger than the error in the semi-derivative approach for the same number of terms. Notice also that for a very large number of terms, Euler-MacLaurin still has error, while the semi-derivative approach does not. This performs better than the Euler-MacLaurin summation formula.

Considering that in reality particles moving along the bed of a channel would experience a turbulent velocity field (not as smooth as the test function), a second test was performed using a “more turbulent looking velocity field”. To that end, it was considered the following derivative of the relative velocity: $du_r/dt = \cos(t)$ and performed the integration of the Basset term between 0 and $T=100 \pi/2$. This integral has analytical solution in terms of Fresnel integrals, as follows:

$$\int_0^T \frac{\cos t}{\sqrt{T-t}} dt = -\sqrt{2\pi} \left\{ \cos t C\left(\sqrt{\frac{2}{\pi}} \sqrt{T-t}\right) + \sin t S\left(\sqrt{\frac{2}{\pi}} \sqrt{T-t}\right) \right\} \Big|_0^T \quad (\text{B-6})$$

where $C(t) = \int_0^t \cos\left(\frac{\pi z^2}{2}\right) dz$ and $S(t) = \int_0^t \sin\left(\frac{\pi z^2}{2}\right) dz$. These functions are tabulated

in the book by Abramowitz and Stegun (1970). For the above value of T , the integration gives the value 1.253060201. The semi-derivative approach and the Euler-MacLaurin summation formula give the errors for different numbers of terms depicted in Table B-3. Again, we see that the semi-derivative approach does a better job in converging to the right number than the Euler-MacLaurin summation formula.

Table B-3: Comparison of the method with the semi-derivative approach and the Euler-MacLaurin summation formula for an integration of a "more turbulent looking velocity field" which has analytical solution.

N	Semi-derivative Method	Euler-Mac Laurin Method	Error Semiderivative (%)	Error Euler-Mac Laurin (%)
1	22.21441469079	1.49668211888	1672.813	19.442
3	7.21434279466	-1.79915200491	475.738	243.581
9	5.18811465433	-2.92933905411	314.036	333.775
27	4.66783568853	0.36399574478	272.515	70.951
81	1.75815554296	0.76393412206	40.309	39.035
243	1.42522996020	0.97209483967	13.740	22.422
729	1.31140473958	1.09093574870	4.656	12.938
2187	1.27263337040	1.15946342479	1.562	7.469
6561	1.25959918803	1.19902244847	0.522	4.312
19683	1.25524151190	1.22186151231	0.174	2.490
59049	1.25378748899	1.23504760541	0.058	1.437
177147	1.25330265107	1.24266057556	0.019	0.830
531441	1.25314102020	1.24705587523	0.006	0.479
1594323	1.25308714121	1.24959343185	0.002	0.277

It is also necessary to compare the performance of the semi-derivative method in relation to the Brush et al. method to solve the Basset integral. For the Brush et al. method, a Simpson quadrature was used to calculate the integral. In this case, the test function should represent in a more realistic way the type of motion that affect a particle moving in a boundary layer type of flow. The main interest of this project is with particles moving as bedload, where particles describe a saltating trajectory. Using the computational code described in Chapter 3, the results obtained for a single particle jump considering $R_p=53$ (which corresponds to a small particle size) and $\tau_* = 0.056$, shows that the particle velocity in the wall normal component could be describe as a parabolic function of the time (See Figure B-1).

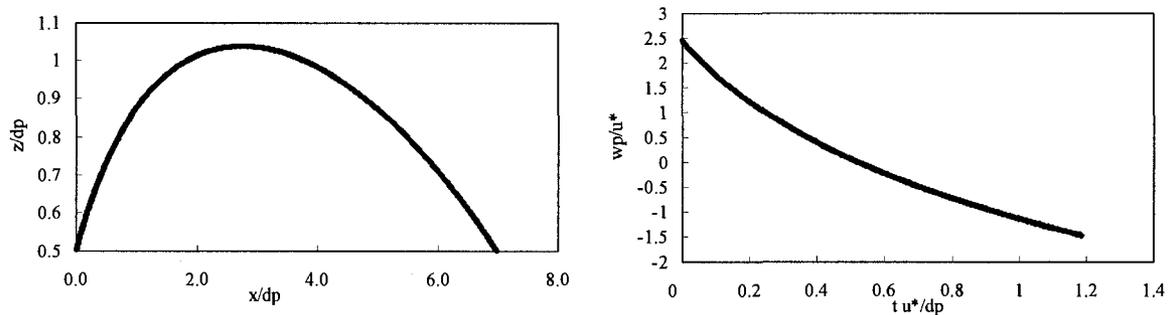


Figure B-1: Left: Single particle trajectory, using the particle diameter as a length scale. Right: Single particle velocity in the wall normal direction, using the shear velocity u^* as a velocity scale. $R_p=53$, $\tau^*=0.056$.

For this specific jump, the particle velocity in the wall normal direction can be estimated using the following second order function of the time (with a correlation function equal to 0.997):

$$w_p(t) = 1.875289056t^2 - 5.281765321t + 2.25364967 \quad (\text{B-7})$$

$$\frac{du_r}{dt} = \frac{dw_p(t)}{dt} = 3.75058112t - 5.281765321 = At - B \quad (\text{B-8})$$

The value of the Basset integral just before the particle hits the wall can be analytical estimated as:

$$\int_0^T \frac{dw_p}{dt} \frac{dt}{\sqrt{T-t}} = \int_0^T (At - B) \frac{dt}{\sqrt{T-t}} = -\frac{2}{3} \sqrt{T-t} [(2T+t)A - 3B]_0^T$$

(B-9)

Considering a collision time equal to 1.188, the value of the Basset integral is -5.038438204. The semi-derivative approach and the Brush et al. method give the errors for different numbers of terms depicted in Table B-4. It is found that the semi-derivative approach does a better job in converging to the right number than the Brush et al formula, with a faster convergence rate. For instance, for an error of 0.015%, the semi-derivative method requires only 5000 points, while the Brush et al. method requires 100,000 points (more than two orders of magnitude) to achieve the same precision. In this case, a reduction of about 20% of the computational time could be achieved by using the semi-derivative method.

Table B-4: Comparison of the method with the semi-derivative approach and the Brush et al. method for a typical particle velocity in a saltating pattern which has analytical solution.

N	Semi-derivative Method	Brush et al. Method	Error Semi-derivative (%)	Error Brush et al. (%)
1000	-5.03457118	-5.03440779	0.077	0.080
2500	-5.03689129	-5.03505008	0.031	0.067
5000	-5.03766473	-5.03573687	0.015	0.054
7500	-5.03792255	-5.03612092	0.010	0.046
10000	-5.03805146	-5.03637346	0.008	0.041
50000	-5.03836085	-5.03742149	0.002	0.020
100000	-5.03839953	-5.03770352	0.001	0.015

Finally, it is necessary to compare the computations of the Basset force in the code developed to track particles, using both the semi-derivative and Brush et al. methods. In this case, the Basset force is calculated in each time step. The results were obtained for a single particle jump considering $R_p=53$ (which corresponds to a small particle size) and $\tau_* = 0.056$. The results of Table B-5 indicate that the percentage of difference between both approximations is less than 3%, which indicates a very good agreement. In addition to providing a close estimate to the Basset force, this methodology is accurate and it converges faster than Euler-Mac Laurin and Brush methods to calculate the Basset integral. It was also found that the semi-derivative method reduces the computational time by 20% as compared with Brush et al.'s method, depending of the desired precision.

Table B-5: Comparison of the method with the semi-derivative approach and the Brush et al. method for a typical particle jump.

N	Basset Integral X Component	Basset Integral Z Component	Error X Component (%)	Error Z Component (%)
100	-4.710	-5.512	12.88	5.03
500	-5.090	-5.683	5.84	2.08
1000	-5.173	-5.721	4.30	1.42
5000	-5.231	-5.740	3.23	1.10
10000	-5.235	-5.743	3.15	1.05
50000	-5.239	-5.745	3.09	1.02
Brush's	-5.406	-5.804	-	-

APPENDIX C

SIMULATION RESULTS FOR $R_p=73$ Table C-1: Dimensionless particle jump height results. $R_p=73$

RUN	τ^*/τ^*c	A		B		C		D	
1	1	0.548 +/-	0.01	0.869 +/-	0.05	0.796 +/-	0.09	0.766 +/-	0.03
	1.5	0.694 +/-	0.03	1.231 +/-	0.13	0.953 +/-	0.12	1.034 +/-	0.09
	2	0.819 +/-	0.05	1.525 +/-	0.19	1.161 +/-	0.15	1.306 +/-	0.14
	2.5	0.899 +/-	0.07	1.720 +/-	0.23	1.345 +/-	0.18	1.481 +/-	0.17
	3	0.910 +/-	0.06	1.786 +/-	0.22	1.468 +/-	0.19	1.546 +/-	0.17
2	1	0.006 +/-	0.01	0.009 +/-	0.01	0.002 +/-	0.00	0.005 +/-	0.01
	1.5	0.008 +/-	0.02	0.017 +/-	0.03	0.003 +/-	0.00	0.007 +/-	0.01
	2	0.005 +/-	0.01	0.014 +/-	0.02	0.003 +/-	0.00	0.006 +/-	0.01
	2.5	0.006 +/-	0.01	0.021 +/-	0.03	0.004 +/-	0.01	0.007 +/-	0.01
	3	0.011 +/-	0.02	0.023 +/-	0.03	0.003 +/-	0.01	0.010 +/-	0.02
3	1	0.004 +/-	0.01	0.008 +/-	0.01	0.003 +/-	0.01	0.007 +/-	0.02
	1.5	0.008 +/-	0.02	0.010 +/-	0.02	0.002 +/-	0.01	0.008 +/-	0.02
	2	0.009 +/-	0.03	0.013 +/-	0.03	0.013 +/-	0.03	0.008 +/-	0.02
	2.5	0.004 +/-	0.02	0.008 +/-	0.02	0.004 +/-	0.01	0.006 +/-	0.01
	3	0.003 +/-	0.01	0.021 +/-	0.04	0.003 +/-	0.01	0.010 +/-	0.03
4	1	0.116 +/-	0.06	0.124 +/-	0.09	0.111 +/-	0.14	0.192 +/-	0.25
	1.5	0.445 +/-	0.26	0.419 +/-	0.21	0.209 +/-	0.29	0.439 +/-	0.53
	2	0.869 +/-	0.49	0.654 +/-	0.34	0.314 +/-	0.49	0.713 +/-	0.82
	2.5	1.361 +/-	0.64	0.919 +/-	0.37	0.536 +/-	0.73	0.779 +/-	1.09
	3	1.834 +/-	0.75	1.060 +/-	0.33	0.726 +/-	0.99	1.104 +/-	1.41
5	1	0.152 +/-	0.09	0.355 +/-	0.19	0.067 +/-	0.05	0.249 +/-	0.14
	1.5	0.419 +/-	0.28	0.929 +/-	0.44	0.406 +/-	0.23	0.915 +/-	0.40
	2	0.771 +/-	0.52	1.553 +/-	0.75	0.666 +/-	0.42	1.413 +/-	0.62
	2.5	1.280 +/-	0.66	2.226 +/-	0.92	1.092 +/-	0.53	1.834 +/-	0.83
	3	1.473 +/-	0.76	2.659 +/-	1.30	1.705 +/-	0.68	2.323 +/-	0.97
6	1	0.348 +/-	0.15	0.265 +/-	0.14	0.045 +/-	0.03	0.166 +/-	0.10
	1.5	0.782 +/-	0.29	0.662 +/-	0.34	0.206 +/-	0.15	0.661 +/-	0.33
	2	1.275 +/-	0.45	1.437 +/-	0.70	0.636 +/-	0.40	1.163 +/-	0.56
	2.5	1.775 +/-	0.51	2.032 +/-	0.88	1.116 +/-	0.53	1.698 +/-	0.74
	3	2.281 +/-	0.62	2.634 +/-	1.18	1.440 +/-	0.72	2.144 +/-	0.92

Table C-2: Dimensionless particle jump length results. $R_p = 73$

RUN	τ^*/τ^*c	A		B		C		D	
1	1	0.414 +/-	0.05	2.814 +/-	0.52	2.757 +/-	0.44	2.216 +/-	0.34
	1.5	2.178 +/-	0.38	7.190 +/-	1.79	5.518 +/-	1.20	5.634 +/-	1.30
	2	4.180 +/-	0.90	11.708 +/-	3.17	8.815 +/-	2.05	9.753 +/-	2.35
	2.5	5.741 +/-	1.32	15.413 +/-	4.44	12.119 +/-	2.99	13.019 +/-	3.29
	3	6.141 +/-	1.36	16.957 +/-	4.59	14.558 +/-	3.61	14.561 +/-	3.69
2	1	0.262 +/-	0.28	0.280 +/-	0.28	0.097 +/-	0.12	0.179 +/-	0.20
	1.5	0.251 +/-	0.44	0.477 +/-	0.54	0.138 +/-	0.17	0.266 +/-	0.32
	2	0.215 +/-	0.33	0.490 +/-	0.50	0.134 +/-	0.19	0.255 +/-	0.33
	2.5	0.263 +/-	0.40	0.685 +/-	0.72	0.188 +/-	0.24	0.317 +/-	0.37
	3	0.487 +/-	0.60	0.749 +/-	0.86	0.179 +/-	0.23	0.389 +/-	0.54
3	1	0.127 +/-	0.22	0.267 +/-	0.26	0.113 +/-	0.18	0.210 +/-	0.28
	1.5	0.251 +/-	0.44	0.292 +/-	0.44	0.098 +/-	0.20	0.255 +/-	0.39
	2	0.302 +/-	0.59	0.404 +/-	0.67	0.404 +/-	0.67	0.265 +/-	0.44
	2.5	0.196 +/-	0.42	0.288 +/-	0.54	0.167 +/-	0.36	0.254 +/-	0.35
	3	0.179 +/-	0.31	0.668 +/-	0.92	0.131 +/-	0.27	0.351 +/-	0.68
4	1	0.598 +/-	0.39	0.878 +/-	0.56	1.133 +/-	1.21	1.643 +/-	1.81
	1.5	3.431 +/-	2.22	3.819 +/-	1.68	2.431 +/-	2.79	4.103 +/-	4.31
	2	7.937 +/-	4.50	6.928 +/-	3.08	3.765 +/-	5.08	7.288 +/-	7.66
	2.5	13.920 +/-	6.07	10.706 +/-	3.75	6.542 +/-	8.02	8.413 +/-	10.61
	3	20.252 +/-	7.40	12.984 +/-	3.54	9.053 +/-	11.20	12.485 +/-	14.65
5	1	0.942 +/-	0.66	1.960 +/-	1.21	0.444 +/-	0.40	1.590 +/-	0.99
	1.5	3.490 +/-	2.33	6.715 +/-	3.54	3.721 +/-	2.09	7.082 +/-	3.01
	2	7.353 +/-	4.62	13.286 +/-	6.03	7.101 +/-	4.17	0.000 +/-	5.47
	2.5	13.555 +/-	6.12	20.395 +/-	8.03	12.351 +/-	5.39	17.911 +/-	7.42
	3	17.278 +/-	7.33	26.186 +/-	11.86	19.894 +/-	6.83	24.139 +/-	9.26
6	1	2.537 +/-	0.80	1.190 +/-	0.85	0.242 +/-	0.23	0.782 +/-	0.63
	1.5	6.898 +/-	1.94	4.323 +/-	2.61	1.674 +/-	1.35	4.734 +/-	2.69
	2	12.345 +/-	3.21	11.574 +/-	5.78	6.331 +/-	3.94	9.865 +/-	4.86
	2.5	18.375 +/-	3.93	18.371 +/-	7.65	12.308 +/-	5.48	16.215 +/-	6.72
	3	25.216 +/-	4.85	26.132 +/-	10.83	17.114 +/-	7.39	22.044 +/-	8.68

Table C-3: Dimensionless mean stream-wise particle velocity results. $R_p = 73$

RUN	τ^*/τ^*c	A		B		C		D	
1	1	1.461 +/-	0.68	3.671 +/-	1.13	4.700 +/-	0.54	3.416 +/-	0.94
	1.5	2.915 +/-	1.08	5.100 +/-	1.39	5.719 +/-	0.66	4.756 +/-	1.18
	2	3.722 +/-	1.26	5.933 +/-	1.52	6.430 +/-	0.75	5.624 +/-	1.32
	2.5	4.134 +/-	1.34	6.396 +/-	1.58	6.931 +/-	0.82	6.098 +/-	1.39
	3	4.187 +/-	1.35	6.548 +/-	1.61	7.203 +/-	0.86	6.268 +/-	1.41
2	1	4.974 +/-	0.18	4.389 +/-	0.20	4.191 +/-	0.15	4.298 +/-	0.17
	1.5	5.201 +/-	0.21	4.391 +/-	0.24	4.054 +/-	0.15	4.222 +/-	0.20
	2	5.341 +/-	0.22	4.567 +/-	0.26	3.976 +/-	0.16	4.246 +/-	0.23
	2.5	5.418 +/-	0.22	4.723 +/-	0.28	3.920 +/-	0.16	4.240 +/-	0.25
	3	5.464 +/-	0.24	4.786 +/-	0.30	3.889 +/-	0.18	4.310 +/-	0.27
3	1	5.004 +/-	0.22	4.511 +/-	0.30	4.315 +/-	0.21	4.409 +/-	0.26
	1.5	5.325 +/-	0.27	4.650 +/-	0.40	4.248 +/-	0.27	4.428 +/-	0.35
	2	5.538 +/-	0.35	4.981 +/-	0.52	4.234 +/-	0.33	4.526 +/-	0.44
	2.5	5.653 +/-	0.39	5.058 +/-	0.52	4.226 +/-	0.38	4.590 +/-	0.52
	3	5.751 +/-	0.00	5.123 +/-	0.53	4.196 +/-	0.45	4.666 +/-	0.54
4	1	1.657 +/-	1.05	4.191 +/-	0.15	4.360 +/-	0.73	4.920 +/-	0.86
	1.5	3.640 +/-	1.84	4.054 +/-	0.15	5.383 +/-	0.93	6.148 +/-	1.13
	2	4.729 +/-	2.15	3.976 +/-	0.16	6.300 +/-	1.19	7.136 +/-	1.33
	2.5	5.764 +/-	2.27	3.920 +/-	0.16	7.084 +/-	1.41	7.996 +/-	1.53
	3	6.710 +/-	2.34	3.889 +/-	0.18	7.773 +/-	1.47	8.655 +/-	1.69
5	1	2.348 +/-	1.25	2.879 +/-	1.67	1.839 +/-	1.05	2.837 +/-	1.46
	1.5	4.008 +/-	1.76	4.786 +/-	2.17	3.849 +/-	1.59	4.922 +/-	1.92
	2	4.883 +/-	2.08	6.081 +/-	2.38	4.950 +/-	1.86	5.795 +/-	2.25
	2.5	6.000 +/-	2.14	6.840 +/-	2.57	5.775 +/-	2.00	6.633 +/-	2.37
	3	6.486 +/-	2.22	7.537 +/-	2.65	6.768 +/-	2.04	7.176 +/-	2.49
6	1	3.862 +/-	1.02	2.127 +/-	1.47	0.161 +/-	0.55	1.612 +/-	1.17
	1.5	5.386 +/-	1.25	3.679 +/-	2.16	2.894 +/-	1.52	3.884 +/-	1.97
	2	6.349 +/-	1.49	5.420 +/-	2.51	4.416 +/-	1.96	5.384 +/-	2.32
	2.5	7.097 +/-	1.60	6.364 +/-	2.69	5.556 +/-	2.09	6.113 +/-	2.48
	3	7.429 +/-	1.62	7.203 +/-	2.72	6.203 +/-	2.22	6.797 +/-	2.54

APPENDIX D

SIMULATION RESULTS FOR $R_p = 250$ Table D-1: Dimensionless particle jump height results. $R_p = 250$

RUN	τ^*/τ^*c	A		B		C		D	
1	2	1.030 +/-	0.47	2.236 +/-	1.06	1.907 +/-	1.10	1.869 +/-	0.95
	2.5	1.077 +/-	0.49	2.302 +/-	1.19	1.982 +/-	1.13	2.121 +/-	1.03
	3	1.153 +/-	0.47	2.648 +/-	0.95	2.287 +/-	1.13	2.275 +/-	0.87
	3.5	0.804 +/-	0.32	2.316 +/-	0.90	1.956 +/-	0.93	1.525 +/-	0.80
	4	0.509 +/-	0.20	1.827 +/-	0.79	1.846 +/-	0.82	1.458 +/-	0.62
2	2	0.008 +/-	0.02	0.017 +/-	0.03	0.003 +/-	0.00	0.009 +/-	0.02
	2.5	0.004 +/-	0.01	0.022 +/-	0.04	0.003 +/-	0.01	0.021 +/-	0.03
	3	0.007 +/-	0.01	0.030 +/-	0.05	0.002 +/-	0.01	0.012 +/-	0.02
	3.5	0.015 +/-	0.03	0.038 +/-	0.07	0.003 +/-	0.00	0.010 +/-	0.02
	4	0.011 +/-	0.02	0.037 +/-	0.07	0.002 +/-	0.00	0.008 +/-	0.01
3	2	0.005 +/-	0.01	0.013 +/-	0.02	0.002 +/-	0.01	0.008 +/-	0.02
	2.5	0.007 +/-	0.02	0.024 +/-	0.05	0.002 +/-	0.01	0.012 +/-	0.04
	3	0.016 +/-	0.08	0.113 +/-	0.21	0.001 +/-	0.01	0.017 +/-	0.04
	3.5	0.007 +/-	0.01	0.034 +/-	0.07	0.002 +/-	0.01	0.018 +/-	0.03
	4	0.008 +/-	0.02	0.022 +/-	0.05	0.003 +/-	0.01	0.013 +/-	0.02
4	2	0.444 +/-	0.58	1.481 +/-	1.60	0.564 +/-	0.72	1.002 +/-	1.18
	2.5	0.533 +/-	0.77	1.116 +/-	1.73	0.943 +/-	1.18	1.145 +/-	1.58
	3	0.827 +/-	1.00	1.993 +/-	2.52	0.968 +/-	1.18	1.508 +/-	1.81
	3.5	0.921 +/-	1.24	4.020 +/-	4.76	1.221 +/-	1.66	3.131 +/-	3.71
	4	0.867 +/-	1.15	3.722 +/-	4.87	1.651 +/-	2.31	3.434 +/-	3.95
5	2	2.214 +/-	0.93	4.790 +/-	2.64	2.596 +/-	1.43	3.518 +/-	1.76
	2.5	3.489 +/-	1.61	5.413 +/-	2.70	4.605 +/-	2.84	4.761 +/-	2.24
	3	3.310 +/-	1.34	6.567 +/-	3.47	4.495 +/-	2.84	5.455 +/-	2.67
	3.5	3.485 +/-	1.63	7.216 +/-	3.47	5.243 +/-	2.57	6.826 +/-	3.28
	4	4.584 +/-	2.08	8.013 +/-	4.41	5.136 +/-	2.43	7.544 +/-	3.92
6	2	2.282 +/-	0.93	3.694 +/-	1.43	1.913 +/-	0.75	3.171 +/-	1.33
	2.5	3.146 +/-	1.11	4.890 +/-	2.11	2.745 +/-	1.06	4.201 +/-	1.64
	3	3.830 +/-	1.54	6.196 +/-	2.50	3.295 +/-	1.06	4.719 +/-	2.01
	3.5	4.209 +/-	1.81	5.772 +/-	2.52	3.961 +/-	1.81	5.331 +/-	2.30
	4	5.063 +/-	2.23	6.202 +/-	2.86	4.471 +/-	1.74	5.806 +/-	2.71

Table D-2: Dimensionless particle jump length results. $R_p=250$

RUN	τ^*/τ^*c	A		B		C		D	
1	2	12.037 +/-	4.70	22.779 +/-	9.14	20.015 +/-	9.40	19.630 +/-	8.39
	2.5	13.932 +/-	5.51	25.614 +/-	11.39	23.041 +/-	10.99	24.334 +/-	10.18
	3	15.238 +/-	5.38	30.501 +/-	9.32	27.391 +/-	10.99	26.976 +/-	8.80
	3.5	10.843 +/-	3.81	27.237 +/-	8.99	24.417 +/-	9.45	19.468 +/-	8.68
	4	6.470 +/-	2.23	21.465 +/-	8.03	22.445 +/-	8.09	17.996 +/-	6.50
2	2	0.338 +/-	0.51	0.534 +/-	0.67	0.148 +/-	0.17	0.320 +/-	0.44
	2.5	0.216 +/-	0.32	0.702 +/-	0.95	0.154 +/-	0.23	0.657 +/-	0.88
	3	0.348 +/-	0.51	0.891 +/-	1.23	0.123 +/-	0.23	0.415 +/-	0.65
	3.5	0.581 +/-	0.84	1.069 +/-	1.52	0.146 +/-	0.21	0.387 +/-	0.52
	4	0.472 +/-	0.65	1.038 +/-	1.52	0.109 +/-	0.16	0.324 +/-	0.43
3	2	0.213 +/-	0.41	0.451 +/-	0.58	0.119 +/-	0.20	0.283 +/-	0.43
	2.5	0.314 +/-	0.53	0.710 +/-	1.08	0.099 +/-	0.20	0.364 +/-	0.84
	3	0.456 +/-	1.38	2.339 +/-	3.53	0.083 +/-	0.20	0.521 +/-	0.99
	3.5	0.321 +/-	0.49	0.982 +/-	1.43	0.112 +/-	0.23	0.594 +/-	0.79
	4	0.321 +/-	0.60	0.664 +/-	1.08	0.122 +/-	0.32	0.441 +/-	0.63
4	2	5.453 +/-	6.26	14.217 +/-	13.68	6.899 +/-	7.72	10.462 +/-	11.12
	2.5	6.899 +/-	9.06	11.910 +/-	16.74	11.173 +/-	13.03	12.589 +/-	15.94
	3	11.011 +/-	12.30	21.082 +/-	25.23	12.193 +/-	13.03	17.956 +/-	19.89
	3.5	12.907 +/-	15.83	39.669 +/-	41.98	16.225 +/-	20.29	34.001 +/-	35.82
	4	13.686 +/-	15.94	39.509 +/-	45.47	21.634 +/-	28.35	36.635 +/-	39.04
5	2	17.537 +/-	6.34	32.782 +/-	16.30	20.696 +/-	13.41	25.560 +/-	12.53
	2.5	32.896 +/-	12.95	40.696 +/-	19.29	36.928 +/-	19.58	36.964 +/-	16.28
	3	31.848 +/-	9.50	53.552 +/-	26.38	41.088 +/-	19.58	43.599 +/-	20.68
	3.5	32.235 +/-	13.16	62.961 +/-	27.82	50.764 +/-	21.23	60.752 +/-	26.33
	4	48.574 +/-	17.07	74.100 +/-	36.28	53.432 +/-	20.91	70.968 +/-	32.77
6	2	17.433 +/-	6.30	24.847 +/-	9.14	15.671 +/-	5.01	22.386 +/-	8.83
	2.5	26.218 +/-	7.90	36.669 +/-	14.63	24.233 +/-	7.76	32.538 +/-	11.38
	3	34.685 +/-	11.74	50.019 +/-	18.26	31.272 +/-	7.76	40.197 +/-	14.89
	3.5	41.245 +/-	14.11	51.486 +/-	19.64	39.754 +/-	14.73	49.081 +/-	18.37
	4	51.680 +/-	18.42	59.305 +/-	23.48	47.801 +/-	14.65	56.672 +/-	22.33

Table D-3: Dimensionless mean stream-wise particle velocity results. $R_p = 250$

RUN	τ^*/τ^*c	A		B		C		D	
1	2	5.182 +/-	1.92	7.127 +/-	2.14	7.134 +/-	1.80	6.824 +/-	1.99
	2.5	5.468 +/-	1.95	7.442 +/-	2.22	7.322 +/-	1.87	7.299 +/-	2.01
	3	5.413 +/-	1.95	7.660 +/-	2.16	7.786 +/-	1.87	7.278 +/-	2.00
	3.5	4.829 +/-	1.77	7.258 +/-	2.11	7.365 +/-	1.79	6.606 +/-	1.86
	4	3.772 +/-	1.53	6.630 +/-	2.03	7.058 +/-	1.76	6.283 +/-	1.78
2	2	4.941 +/-	0.20	4.341 +/-	0.34	3.348 +/-	0.17	3.777 +/-	0.30
	2.5	5.065 +/-	0.22	4.534 +/-	0.39	3.349 +/-	0.19	3.861 +/-	0.33
	3	5.111 +/-	0.23	4.680 +/-	0.44	3.345 +/-	0.19	3.952 +/-	0.36
	3.5	5.148 +/-	0.25	4.666 +/-	0.41	3.307 +/-	0.17	3.905 +/-	0.35
	4	5.170 +/-	0.27	4.656 +/-	0.41	3.272 +/-	0.16	3.823 +/-	0.36
3	2	5.286 +/-	0.43	4.918 +/-	0.71	3.699 +/-	0.47	4.361 +/-	0.70
	2.5	5.552 +/-	0.58	5.205 +/-	0.78	3.735 +/-	0.47	4.394 +/-	0.70
	3	5.616 +/-	0.58	5.421 +/-	0.81	3.751 +/-	0.47	4.769 +/-	0.93
	3.5	5.749 +/-	0.60	5.247 +/-	0.83	3.746 +/-	0.56	4.442 +/-	0.71
	4	5.615 +/-	0.57	5.329 +/-	0.84	3.634 +/-	0.46	4.411 +/-	0.71
4	2	6.748 +/-	1.13	8.510 +/-	1.70	7.058 +/-	1.26	7.960 +/-	1.54
	2.5	7.394 +/-	1.26	8.932 +/-	1.83	8.086 +/-	1.55	8.796 +/-	1.70
	3	7.965 +/-	1.29	9.773 +/-	1.95	8.588 +/-	1.55	9.142 +/-	1.72
	3.5	8.295 +/-	1.42	11.054 +/-	2.16	9.007 +/-	1.70	10.308 +/-	2.15
	4	8.530 +/-	1.40	11.144 +/-	2.19	9.787 +/-	1.82	10.803 +/-	1.98
5	2	6.539 +/-	2.51	8.396 +/-	3.32	7.702 +/-	2.88	7.730 +/-	3.03
	2.5	7.899 +/-	2.55	9.014 +/-	3.34	8.771 +/-	3.01	8.803 +/-	3.11
	3	7.679 +/-	2.55	9.809 +/-	3.28	8.756 +/-	3.01	9.506 +/-	3.07
	3.5	8.390 +/-	2.53	9.993 +/-	3.28	9.338 +/-	2.80	9.881 +/-	3.12
	4	9.065 +/-	2.67	10.443 +/-	3.34	9.241 +/-	2.83	10.267 +/-	3.24
6	2	6.059 +/-	2.64	7.013 +/-	3.15	6.022 +/-	2.36	7.070 +/-	2.95
	2.5	7.244 +/-	2.73	8.442 +/-	3.25	7.026 +/-	2.57	7.912 +/-	3.04
	3	7.915 +/-	2.86	9.057 +/-	3.23	7.026 +/-	2.57	8.499 +/-	3.13
	3.5	8.337 +/-	2.91	8.898 +/-	3.39	7.026 +/-	2.57	8.933 +/-	3.11
	4	9.080 +/-	2.92	9.208 +/-	3.38	8.696 +/-	2.75	9.214 +/-	3.22